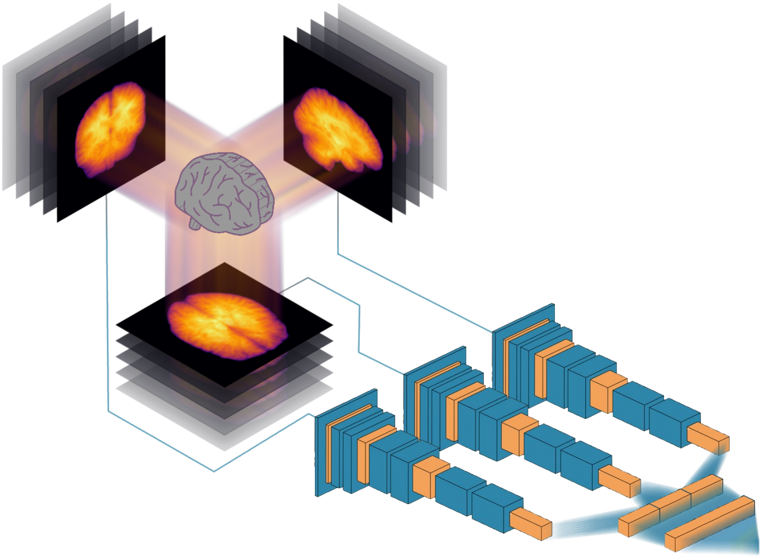
Linköping Studies in Science and Technology
Licentiate Thesis No. 1986

Deep learning on large neuroimaging datasets



Johan Jönemo

LINKÖPING
UNIVERSITY

Linköping Studies in Science and Technology
Licentiate Thesis No. 1986

Deep learning on large neuroimaging datasets

Johan Jönemo

Linköping University
Department of Biomedical Engineering
SE-581 83 Linköping, Sweden

Linköping 2024

This is a Swedish Licentiate's Thesis

Swedish postgraduate education leads to a doctor's degree and/or a licentiate's degree.
A doctor's degree comprises 240 ECTS credits (4 years of full-time studies).
A licentiate's degree comprises 120 ECTS credits.

Edition 1.0

© Johan Jönemo, 2024
ISBN 978-91-8075-569-6 (print)
ISBN 978-91-8075-570-2 (PDF)
ISSN 0280-7971
URL `https://doi.org/10.3384/9789180755702`

Published articles have been reprinted with permission from the respective copyright holder.
Typeset using LaTeX

Printed by LiU-Tryck, Linköping 2024

POPULÄRVETENSKAPLIG SAMMANFATTNING

Magnetresonansavbildningar, ofta kallat MR eller MRI, är en bilddiagnostikmetod som har blivit allt viktigare under de senaste 40 åren. Detta på grund av att man kan erhålla 3D-bilder av kroppsdelar utan att utsätta patienter för joniserande strålning. Dessutom får man typiskt bättre kontraster mellan mjukdelar än man får med motsvarande genomlysningsmetod (CT, eller 3D röntgen). Själva bildinsamlingsförfarandet är också mera flexibelt med MR. Man kan genom att ändra program för utsända och registrerade signaler, inte bara ändra vad som framförallt framträder på bilden (t.ex. vatten, fett, H-densitet, o.s.v.) utan även mäta flöde och diffusion eller till och med hjärnaktivitet över tid.

Maskininlärning har fått ett stort uppsving under 2010-talet, dels på grund av utveckling av teknologin för att träna och konstruera maskininlärningsmodeller dels på grund av tillgängligheten av massivt parallella specialprocessorer – initialt utvecklade för att generera datorgrafik. Detta arbete kombinerar MR med maskininlärning, för att dra nytta av de stora mängder MR data som finns samlad i öppna databaser, för att adressera frågor av kliniskt intresse angående hjärnan.

Avhandlingen innehåller tre studier. I den första av dessa undersöks delproblemet vilken eller vilka metoder för att artificiellt utöka träningsdata som är bra vid klassificering om en person har autism. Det andra arbetet adresserar bedömning av så kallad "hjärn-ålder". Framför allt strävar arbetet efter att hitta lättviktsmodeller som använder en komprimerad form av varje hjärnvolym, och därmed snabbt kan tränas till att bedöma en persons ålder från en MR-volym av hjärnan. Det tredje arbetet utvecklar modellen från det föregående genom att undersöka andra typer av komprimering.

ABSTRACT

Magnetic resonance imaging (MRI) is a medical imaging method that has become increasingly more important during the last 4 decades. This is partly because it allows us to acquire a 3D-representation of a part of the body without exposing patients to ionizing radiation. Furthermore, it also typically gives better contrast between soft tissues than x-ray based techniques such as CT. The image acquisition procedure of MRI is also much more flexible. One can vary the signal sequence, not only to change how different types of tissue map to different intensities, but also to measure flow, diffusion or even brain activity over time.

Machine learning has gained great impetus the last decade and a half. This is probably partly because of the work done on the mathematical foundations of machine learning done at the end of last century in conjunction with the availability of specialized massively parallel processors, originally developed as graphical processing units (GPUs), which are ideal for training or running machine learning models. The work presented in this thesis combines MRI and machine learning in order to leverage the large amounts of MRI-data available in open data sets, to address questions of clinical relevance about the brain.

The thesis comprises three studies. In the first one the subproblem which augmentation methods are useful in the larger context of classifying autism, was investigated. The second study is about predicting brain age. In particular it aims to construct light-weight models using the MRI volumes in a condensed form, so that the model can be trained in a short time and still reach good accuracy. The third study is a development of the previous that investigates other ways of condensing the brain volumes.

Acknowledgments

I acknowledge that this is a thesis. I also acknowledge that I wrote it. Apart from that, I'd also like to acknowledge some things for which I am grateful. I would like to thank my supervisor for giving me an opportunity. He also provided a lot of help and even advice. I would like to thank Muhammad Usman Akbar for his advice, encouragement and for being a friend. I also want to acknowledge the IT-department for their help with this and that and for encouraging[1] me to learn to write my own Kerberos tools. I would also thank any of my fellow graduate students that had helped me.

[1] Indirectly...

Contents

List of Papers

This thesis is based on the following publications, referred to in the text with their roman numeral:

I. Jönemo, Johan and Abramian, David and Eklund, Anders
Evaluation of Augmentation Methods in Classifying Autism Spectrum Disorders from fMRI Data with 3D Convolutional Neural Networks
Diagnostics, Volume 13, Issue 17
DOI: 10.3390/diagnostics13172773, LiU repository number: diva2:1791824

II. Jönemo, Johan and Akbar, Muhammad Usman and Kämpe, Robin and Hamilton, J Paul and Eklund, Anders
Efficient brain age prediction from 3D MRI volumes using 2D projections
Brain Sciences, Volume 13, Issue 9
DOI: 10.3390/brainsci13091329, LiU repository number: diva2:1797861

III. Jönemo, Johan and Eklund, Anders
Brain age prediction using 2D projections based on higher order statistical moments and eigenslices from 3D magnetic resonance imaging volumes
Journal of Imaging, Volume 9, Issue 12
DOI: 10.3390/jimaging9120271, LiU repository number: diva2:1817740

List of Figures

1 Introduction

Brain diseases affect a large number of people worldwide [1, 2], and neuroimaging is often used to study the different diseases [3–6]. In this thesis the focus is on applying deep learning techniques to large open neuroimaging datasets, such as UK biobank [7] and ABIDE [8]. Such techniques can, at least in theory, be used for early detection of different brain diseases.

1.1 Aims

The main aims of this thesis are:

To investigate if deep learning can be used to automatically diagnose brain diseases from neuroimaging data, and what the most appropriate architectures are.

To investigate how to apply deep learning to neuroimaging datasets containing thousands of subjects, using limited hardware.

1.2 Outline of the thesis

Chapter 2 is a short introduction to the brain. It raises the question why, and above all, how, do cells communicate? How could this have come about? The discussion touches upon how cell membranes could be used to propagate a short electrical impuls, as well as how and why some cells might specialise in transferring information by this mechanism. Some aspects of the evolution of the central nervous system, and how clues can be found in our development, is discussed. It gives a little overview of the overall structure of our brain and compares it briefly to some other species. This description is mainly organised after developmental origin, but this also has relevance to the evolutionary history of the different parts. It also explains some conditions of the brain that relate to the work I have done.

Chapter 3 is a short introduction to nuclear magnetic resonance in general and its application on imaging in particular. It touches upon why some nuclei are magnetically active. Some physical underpinnings are discussed. Different relaxation time constants and their relation to different image types are introduced. Some acquisition techniques are discussed. The scanner and its principal parts are also presented.

In chapter 4 I give an introduction to data analysis and machine learning, with a focus on the techniques that I used the most in my investigations. Principal component analysis, support vector machines and decision tree learning are discussed briefly. There is also a section that talks about preprocessing. It focuses partly on the difference between preprocessing needs for artificial neural networks and normal statistical methods, partly on the types and amount of preprocessing that is needed, in general, for functional magnetic resonance imaging data. The following part is about artificial neural networks, and also introduces convolutional neural networks (CNNs) which have been used in the three papers.

1.3 Abbreviations

AD	Alzheimer's disease
ASD	autism spectrum disorder
CNN	convolutional neural network
CNS	central nervous system
CT	computed tomography
DL	deep learning
dMRI	diffusion MRI
EEG	electroencephalogram
fMRI	functional MRI
FSL	FMRIB software library
GPU	graphics processing unit
MEG	magnetoencephalography
ML	machine learning
MRI	magnetic resonance imaging
MR	NMR meaning 1, but when concerned with humans
NMR	1. nuclear magnetic resonance, the physical phenomenon 2. NMR spectroscopy
PCA	principal component analysis
PD	1. Parkinson's disease 2. proton density
SI	système international d'unités

1.4 Reproducibility

All included papers are based on openly available data, and the used code is shared on GitHub [1]. Together, this facilitates reproduction and extension of our results [9, 10].

1.5 Funding

This research was supported by the Swedish research council (2017-04889), the ITEA/VINNOVA project ASSIST (Automation, Surgery Support and Intuitive 3D visualization to optimize workflow in IGT SysTems, 2021-01954), and the Åke Wiberg foundation (M20-0031, M21-0119, M22-0088).

[1] https://github.com/emojjon/

2
The Brain

"The mind is a terrible thing to taste."

Alain Jourgensen

Our evolution is characterised by competition in the presence of inhospitable environments, predation and starvation. As the “competitors” in this race have honed their various skills, collection and analysis of information from the environment have become increasingly important.

2.1 Origins

Even single celled organisms without identifiable nuclei (the domains of Bacteria and Archaea, respectively, according to currently widely accepted classification) gather and react on external information – chiefly chemical in nature, albeit in very predictable ways. Typically this consists of detecting chemical species on the surface of the cell, by means of receptors, selective pores or transporters, all of which are composed of a small number of macromolecules — possibly even one — typically proteins. Chains of electrochemical activity result in hardcoded responses. Even so, these organisms are often capable of e.g. chemotaxis and quorum sensing/signalling [11, 12].

The Excitable Membrane

It has been hypothesised that primitive life evolved in an environment with relatively high concentration of potassium and low concentration of sodium, compared to e.g. the sea of today. This early life was in all likelihood almost completely vegetative. Each unit (seeing as calling them cells might invoke an idea of too complex a system) marshalled what little energy was needed to make a copy of itself and little more.

To be able to survive outside of such a refugium, however, an organism would have to be able to uphold this relationship between potassium and sodium internally while living in a high sodium, low potassium environment such as the sea. This would require a cell membrane not permeable to either of these ions (to any significant degree), special channels for these ions that could be tightly controlled and, above all, a pump to maintain this disequilibrium, including a supply of energy to run it. In fact, all known living extracellular[1] organisms today have all of these traits.

Upholding the homœostasis of such cells (given the extant transmembrane proteins) also has the side effect of introducing a negative voltage as measured from the outside, over the membrane — often somewhat spuriously called the membrane potential. Given a few ubiquitous ion channel types, this makes membranes potentially excitable. This mechanism can somewhat simplistically[2] be seen as consisting of three parts. The first is a sodium channel that is sensitive to the voltage over the membrane. It circles through three states with certain (possibly voltage dependent) probabilities, namely — in order — closed, open and refractory. The second one is a potassium channel that just leaks potassium passively, subject only to the membrane voltage and the difference in concentration on either side. The third is the sodium potassium ion exchanger, which compensates for influx of sodium and outflux of potassium.

In fact, we could — just to test the idea — make a very simple model of the membrane voltage U with discrete intervals, indexed by i, of a "one dimensional" membrane at discrete points in time indexed by t. Each interval has N sodium channels that can be in any of the described states (where appropriate indicated by a superscript of o, c or r). The formulae also contain a number of arbitrary constants C_i, in order to tweak the model to be excitable enough to react to a sufficiently large localised pertubation of the voltage, but not so excitable as to fire spontaneously or self-oscillate.

$$\Delta_t N^o_{i,t} = C_1 \cdot N^c_{i,t} \sum_{k=i-n}^{i+n} (U_{k,t} - \tilde{U})^2 - C_2 \cdot N^o_{i,t} \tag{2.1}$$

$$\Delta_t N^r_{i,t} = C_2 \cdot N^o_{i,t} - C_3 \cdot N^r_{i,t} \tag{2.2}$$

[1] When cells live within other cells there is a continuum of dependency levels with regard to the host cell. In this context, it can become a bit of a grey area what traits such organisms possess.

[2] In reality there are usually several types of channels for each ion as well as other ion types participating in the interaction.

$$\Delta_t N^c_{i,t} = C_3 \cdot N^r_{i,t} - C_1 \cdot N^c_{i,t} \sum_{k=i-n}^{i+n} (U_{k,t} - \tilde{U})^2 \tag{2.3}$$

$$\Delta_t U_{i,t} = C_4 \cdot N^r_{i,t} - C_5(U_{i,t} - \tilde{U}) \tag{2.4}$$

Where the equations describe the increase from one time to the next of (2.1) the number of open Na$^+$-channels, (2.2) the number of refractory Na$^+$-channels, (2.3) the number of closed Na$^+$-channels and (2.4) the trans-membrane voltage, respectively.

In this crude model, closed channels at position i open with a probability[3] determined by the average voltage on a neighbourhood of i (all points at a distance no more than n). All other transitions occur with constant probability[3]. The steady state maintained by the sodium potassium exchanger and the leak channels, keeping concentrations (not accounted for in the above model) and voltages stable over longer periods of time, are here modelled by a simple exponential decay of the voltage to the "resting potential" $\tilde{U}$.

The above model is not very accurate and ignores any and all complications. It does however illustrate how a system[4] of ion channels, some of which are voltage sensitive, can create an electric signal travelling along a membrane, see Figure 2.1.

A more involved and much more accurate model was presented by Alan Hodgkin and Andrew Huxley in 1952, for which they were awarded the Nobel Prize in Physiology or Medicine in 1963 [13]. This model was developed trying to fit experimental measurements on a very large unmyelinated nerve from a squid[5], with the more general (but not describing action potentials, per se) models then recently postulated for membrane potentials [14, 15]. Several other famous mathematical models expanding and refining the one developed by Hodgkin and Huxley have since been suggested [16, 17].

[3] The equations 2.1–2.4 show mean differences and thus somewhat glosses over the stochasticity of the model, in favour of simplicity.

[4] Even though only one type of ion channel is explicitly modelled in this case.

[5] Because molluscs do not have myelinated nerves, their only recourse to increase the conduction speed is to increase the axonal diameter, wherefore certain nerves from a squid were deemed the easiest to perform measurements on.

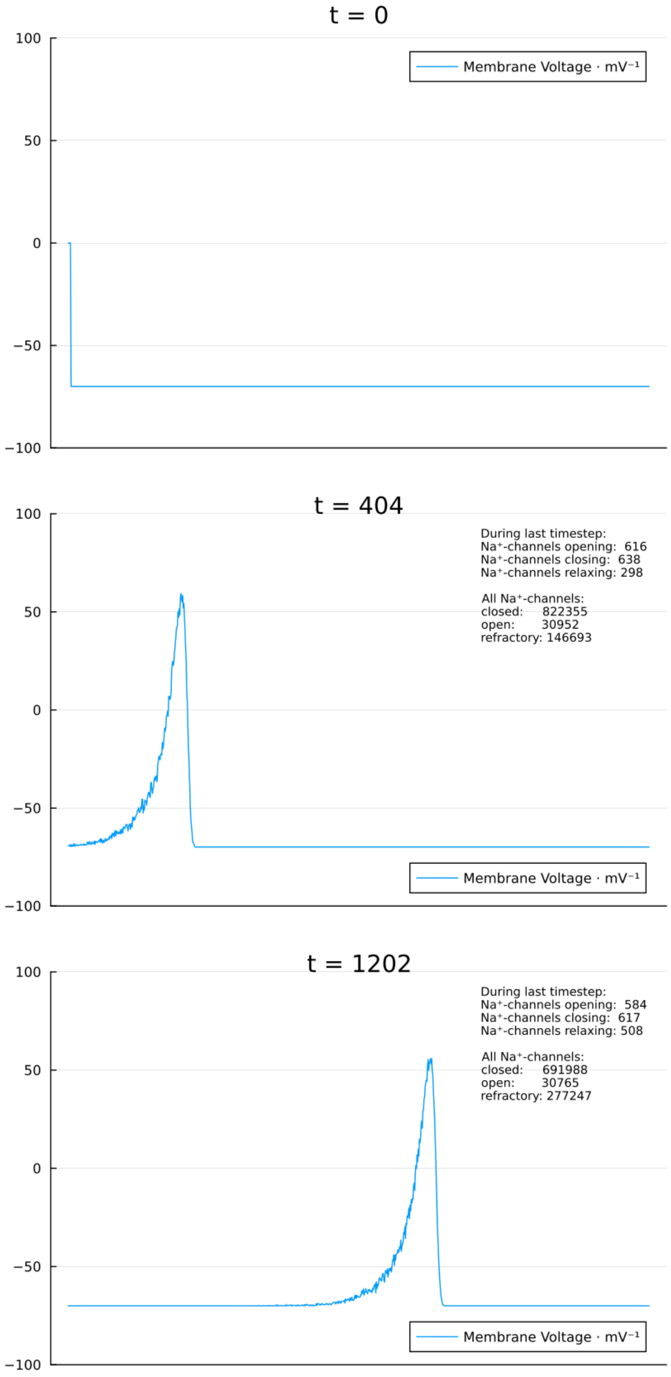


Figure 2.1: Simulation of the model described in equations 2.1–2.4 — with suitable constants — with a given initial condition. The top diagram represent the initial voltage as functions of n, the ones below how the voltage develops over time.

The Nerve Cell

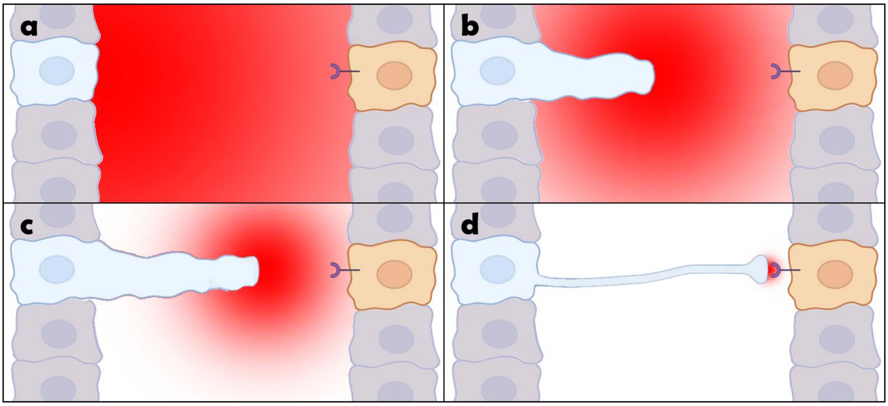


Figure 2.2: This is a conceptual illustration of how specialisation of certain cells into nerve cells, could have evolved. In all frames the blue cell is the one emitting some kind of chemical signal, the concentration of which can be seen by the redness of the background. The target cell is orange and has an (oversized) receptor, the cleft of which is the point where the signal substance concentration is measured. **a** With no special adaptations the signalling cell has to produce quite large amounts of signal substance to guarantee that the receptor on the target cell would be triggered. The time for said substance to diffuse would also be long. **b** & **c** These are intermediary stages where the signalling cell has changed shape in such a way that it can use an ever smaller amount of signal substance. The diffusion time in **b** would here have been only a fourth of that in **a** (assuming that the distance is half). **d** Here one can see this development taken to its logical conclusion. The signalling cell have developed a thin stalk to exactly the place where the signal is received, effectively forming a synapse. The information can quickly travel the membrane of the "stalk" or proto-axon, as the case may be, and trigger release of built up reserves of signal substance at the tip. The distance over which the substance need to diffuse can be made so short that the speed matches that of the signal transmission along the membrane.

One can imagine, as cells became fixed in space in relation to each other, that releasing chemical signals into the surrounding medium and waiting for them to diffuse was both wasteful and slow. Clearly, it would be a great evolutionary advantage for organisms that could provide a better way to transport information.

A possible adaptation could be that cells begun to extend, little by little, towards the cells they needed to signal. At the same time an intracellular signal would have to be sent so that the now remote part of the cell could release its signal substance in this new and more adapted location. The previously described travelling voltage spike would be an excellent candidate for a reasonably fast intracellular signal. If we take into account the energy expenditure of this electric activity, it is also clear that having a few cells specialise on this behaviour and handle the signalling for the rest of the cells would be beneficial. For a conceptual illustration of such an adaptation process, see figure 2.2.

While much of the detail is buried in a distant past, the reasoning above provides at least an intuition about how a nervous system could have evolved. The fact that many other types of cells[6] utilise action potentials, as this type of "travelling voltage spike" is usually called, also supports this aetiology.

Finding information about soft tissues and indeed very small structures of such tissue, from the fossil records, is much harder than gaining information that can be gleaned from fossils of mineralised structures such as bones of apatite, exoskeletons of calcified chitin, trilobite lenses of calcite, structural elements from diatoms and certain sponges of silica, and so on. Never the less, a good argument can be made for nerve cells existing in animals at least from some point in the Ediacaran (635–540 Ma) period [18]. We cannot exclude the possibility that nerve cells existed earlier than that since the fossil records from earlier periods are extremely scant and only contain very small specimens, making it even more difficult to analyse or even find them.

The Central Nervous System

If we were to trace our ancestry back to the Cambrian period (540–485 Ma), it is widely beleived that we would find something closely resembling Pikaia — a Cambrian animal attested in fossils found in the Burgess Shale deposits in British Columbia, Canada [19]. This animal appears to have a tube containing nerve cells just dorsal to its notochord — an elastic rod helping the swimming motion and providing structure. This neural tube is likely what evolved into our central nervous system (CNS), i.e. our brain and spinal cord. Indeed, in an embryonic state humans (and other vertebrates) have a notochord which serves to guide dorsal ectodermal cells to invaginate and form the neural tube, which in later developmental stages give rise to our brain and spinal cord.

As we belong to the taxon Gnathostomata, that is to say the jawed vertebrates, it is very telling to look at the structure of the brain of *Coccocephalus wildi* — a ray-finned fish that lived during the Carboniferous period (360–300 Ma). Because of a remarkably well preserved 320 Ma old fossil, the structure of its brain has been studied [20]. If we take into consideration that different parts of the brain can quite easily — on an evolutionary time scale — change size according to the sizes of the animals and their different living patterns, it is interesting how, in its overall structure and organisation, it is much like ours.

[6] For example muscle cells, certain endocrine cells

It is not known when the last common ancestor of lobe-finned fish — including for example coelacanths, lung fish, cows and humans — and ray-finned fish lived, or what it looked like. We can guess that it lived during the late Silurian period (445–420 Ma). It seems however that the branch containing this ancestor had evolved something that would offer these two orders of bony fish a common advantage, namely myelinisation of nerve fibres. This made it cheaper to evolve a more complicated nervous system, as axons could transmit signals fast without having large diameters, thus using less energy (per axon). It seems likely that much of the structures we have in common with *Coccocephalus wildi* also occurred in this last common ancestor.

2.2 Structure

Below, the primitive partition of our central nervous system in early embryonic development, the primary and secondary brain vesicles, will be used to briefly provide a structured overview of the brain.

Our brain consists of — in order of how tracts diverge from the spinal cord and also in order of evolutionary age — rhombencephalon, mesencephalon and prosencephalon. It should be noted that, whereas these terms and some others used below, in sensu stricto might apply only to certain early developmental phases, they are here used in sensu lato to mean any parts of the brain that develop from the corresponding structure.

Throughout the brain there exists a variety of cell types. The blood vessels are made up of epithelial cells, smooth muscle cells and fibroblasts. Inside of them all types of blood cells pass. In the white matter, oligodendrocytes provide myelin sheets for axons. Other glial cells provide a range of "services" to the neurons. This will be taken as given below and focus will be on the structures containing neurons.

Because of how neural tissue fold while our brain is formed it is often useful to talk about caudal and rostral directions, by which we mean in the direction of neural tissue formed from rear and front parts of the embryonic neural tube, respectively, regardless of the exact physical position in the mature brain. In general the description below follows the order from more caudal parts, such as the hind brain, to more rostral parts such as the cerebrum. By brainstem we mean the stalk-like structure that connect the brain caudally to the spinal cord, that is everything except the forebrain or prosencephalon and the cerebellum.

Another caveat is that the brain has plasticity. Even in a single individual, neural tissue can be repurposed in response to changing needs or damage. In evolution more rostral parts of the brain have typically evolved to take

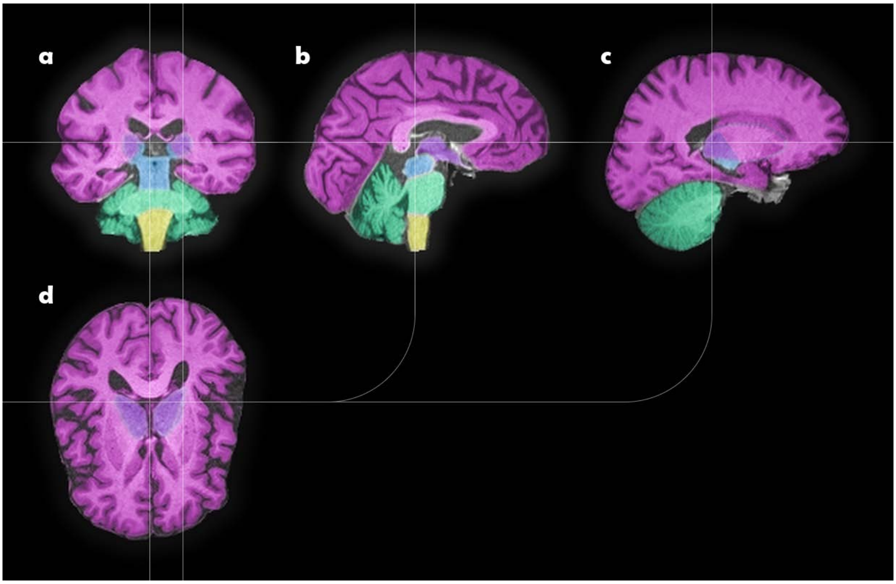


Figure 2.3: An adult brain (from UK Biobank) MRI-volume overlaid with colours representing which part — actually which "secondary brain vesicle" — the nervous tissue belong to. The myelencephalon and metencephalon — together constituting the rhombencephalon — are yellow and green, respectively, the mesencephalon is blue and the diencephalon and the telencephalon — together constituting the prosencephalon — are bluish and reddish purple, respectively. The different views represent (from left to right, top to bottom): **a** A coronal section taken so that it passes through the medulla oblongata. With reference to the exterior of the head it would go approximately through the outer opening or the ear canal (external auditory meatus). **b** A sagittal section taken near the mid-sagittal plane. **c** A sagittal section taken a short distance from the mid-sagittal plane. It would pass through the medial half of the eye. **d** A transversal section at approximately the height of the eyebrows.

over some tasks from more caudal parts. The functions described below are therefore true for humans and almost certainly for all simians, but perhaps not for more distant relatives such as a bird or a frog.

Rhombencephalon

The rhombencephalon or hindbrain is the oldest part of the brain and we share it with remotely related groups such as insects and crustaceans (where it is typically referred to as the supraesophageal ganglion). Considering how remote this kinship is, the first rhombencephalon must have been present in a very old animal indeed, perhaps during the late Ediacaran period (570–540 Ma). In vertebrates we call the parts of the rhombencephalon — from lower to higher — myelencephalon (in humans often called medulla oblongata –

the extended marrow, by which we understand the extended spinal cord) and metencephalon.

Myelencephalon

This is probably the first part of the brain to evolve and has nuclei (collections of nerve cell bodies) that control functions that in humans[7] are not necessarily subject to our own volition or perception, such as heart rate, breathing, vasoconstriction, et.c.

Metencephalon

The rostral part of the hindbrain — the metencephalon — consist of the cerebellum and pons. The cerebellum in a human is a smaller delimited part of the brain, but contains more neurons than the entire rest of the body. This is the center for advanced and learned movement patterns, including integration of sensory information from our eyes, vestibula, proprioception, et.c. The pons contains tracts that connect the forebrain, the cerebellum and the spinal cord (and ultimately the peripheral neural system). The sensory signals to the brain pass here on their way to the thalamus.

Mesencephalon

The midbrain or mesencephalon is the smallest part of the brain stem. Much of it consist of tracts connecting the prosencephalon to the caudal parts of the brain.

The midbrain has two pairs of protuberances on its backside: the inferior and superior culliculi. These are in fact nuclei that monitor the auditory and visual input, respectively, to be able to fast-track information that needs to be handled quickly, such as a loud bang or a big fast-moving object in our field of vision.

In the interior of the midbrain we find the substantia nigra, which is a dopamine producing nucleus that supplies structures in the diencephalon with this signal substance. It is involved with inhibiting conscious voluntary movement to a suitable level. It seems that without this regulation, motor systems get excessive inputs which are counteracted at more distal parts of the nervous system, resulting in difficulty initiating conscious movement and tremor as can be seen in Parkinson's disease which is a degradation of the substantia nigra.

[7] I do not purport to know what — if anything — a slug or even a cow is conscious of.

Even more centrally in this part of the brain, the nuclei rubri can be found. These structures connect through a dedicated tract of the spinal cord (the rubrospinal tract) to muscles in the upper part of the body. These nuclei seems to play a role in arm movement while walking. As voluntary movement is normally initiated in the primary motor cortex of our telencephalon, it is likely that this is a vestigial part of this system whose role has now almost entirely been taken over by more rostral structures.

In the midbrain we also encounter nuclei connected to control of our eyes. The saccadic movements of the eye that we use to let "sweep" over a bigger field of vision than we can take in without moving our eyes is dependent on the substantia nigra and nearby we also find nucleus oculomotorius and the Edinger–Westphal nucleus which control muscles moving the eye and muscles controling the iris and lense, respectively.

Prosencephalon

Rostrally to the midbrain, most structures separate into the two crura (legs) that lead to the left and right hemispheres. The diencephalon is the caudal part of the forebrain and consist of the structures attached to these crura. Between them the third ventricle is located. Rostral to the diencephalon, but physically surrounding it on all sides we have the end brain or telencephalon which is the majority of our brain volume.

Diencephalon

Either side of the diencephalon consists of a large structure called the thalamus and several smaller adjacent structures, i.e. the hypothalamus, the subthalamus and the epithalamus. The two halves are connected by the thalamus through the habenular commissure and by the epithalamus through the posterior commissure.

All sensory information except for smell pass through the thalamus where it is filtered, prioritised and dispatched to different parts of the endbrain. Areas of cortex in the cerebrum — as the endbrain is often called — that receive information from thalamus typically also have plenty of projections into the thalamus. This feedback is probably essential for allowing the thalamus to correctly process and select the outgoing information and is also believed to play an important role in wakefulness.

At the front bottom of the thalamus, the hypothalamus is connected to the pituitary gland – also known as hypophysis. Most of this structure does not originate from any part of the nervous system, but is rather differentiated from the developing ectodermal invagination that cover much of the inside of the mouth. Many of the nuclei in the hypothalamus are concerned with regulating hormone secretion in the pituitary gland alternatively directly projecting the axons that make said hormones into the pituitary, which in turn releases into the blood stream hormones that regulate a large number of systems in the body, such as growth, sexual maturation, body temperature, water reabsorption in the kidneys, arousal, lactation, et.c.

The subthalamus lies under the thalamus, behind the hypothalamus and seems be involved in the complex regulation of selective inhibition of actions as loss of function of this region on either side lead to hemiballismus, i.e. uncontrolled exaggerated movements of half the body, mainly the arm, on the opposite side of the body from the insult.

The epithalamus is the part located behind the thalamus in the rear of the diencephalon. Between the left and right parts the pineal gland is located, roughly on the midline of the brain. While the pineal gland is enervated, it mainly consists of endocrine cells – hence the designation gland. It controls the internal body clock and secretes melatonin as a signal for the nightly part of the cycle. Interestingly, even though this gland is located more or less in the middle of the head, it is thought to have evolved from a light sensor — a third eye — on the top of the head. In some animals, such as amphibians, lizards and tuataras, the pineal gland has a stalk that penetrates the skull in the sagital suture, forming the foramen pinealis, and has a small eye at the end of it.

Telencephalon

The endbrain, cerebrum or telencephalon is the largest part of the human brain and is involved in a wide range of activities. All conscious perception, voluntary motion and cogitation is connected to some part — often several — of the endbrain.

It consists of two, almost completely separated halves or hemispheres, each consisting of a frontal, a parietal, an occipital, a temporal and an insular lobe. The two hemispheres are mainly connected through the two commissures: corpus callosum and the hippocampal commissure.

In some early ancestor it is likely that the telencephalon was a small part of the brain that only processed our sense of smell. In fact, our sense of smell is the only mode of perception that is contained in it's entirety in our endbrain. Cells in our olfactory lobe have neurites that protrude through the lamina cribrosa into our nasal cavity. Humans (and all vertebrates) effectively smell with their brain.

In mammals the telencephalon still houses the olfactory lobes and some other structures, typically located quite centrally, near the diencephalon, such as the fornix with its hippocampi, that have a special organisation, but otherwise it is dominated by the same general architecture: neocortex. Like the cerebellum, but unlike all other parts of the central nervous system, the grey matter is mostly located on the surface of the lobes, which in humans is very large because of folding, and it consists almost everywhere in the endbrain of neocortex – a six-layered heavily interconnected network of large nerve cells. This general architecture is then organised into a large number of areas with different tasks in processing incoming information, deciding what actions to take and getting this executed by the rest of the body, or just organise, analyse and store information. These areas are also dynamic to a certain degree. Over time, areas that are used a lot get allocated more cortex and vice versa, probably helped by the relative structural homogeneity of the neocortex. The white matter underneath the cortex consist of fast myelinised nervefibers that connect areas far apart.

2.3 Pathology

The research described in the papers in this thesis, have been concerned with a few different disorders of the brain. All these disorders can be studied using neuroimaging, especially magnetic resonance imaging (MRI) which is covered in chapter 3. Although the disorders are different, certain biomarkers like brain age, which is the focus of papers 2 and 3, can potentially be used for early detection in several cases. Brain age simply means how old the brain appears, and a large difference compared to the biological age can indicate a disease [21, 22].

Autism Spectrum Disorders

Autism Spectrum Disorders (ASD) is a condition or rather a continuum of conditions characterised by difficulties with communication, especially the more abstract social aspects thereof, and with restricted, stereotypical and repetitive behavioural patterns [2, 23]. It affects approximately 1% of the population. Typically, ignoring or overreacting to sensory stimuli is also a part of the clinical picture. It usually presents itself in early childhood, though high-functioning individuals may be diagnosed later .

The designation of ASD as a spectrum reflects not only the range of the severity of the symptoms, but also a wide qualitative variety in the way it manifests itself in different people.

History

While there is no reason to assume that ASD didn't occur throughout our history, the records are scant. This is most likely due to another understanding or perhaps lack thereof of neuropsychiatric conditions. That is, to the extent people described persons with atypical behaviour attributable to such conditions, it was mainly in the context of demon possession or — in later historical periods — of some kind of generalised lunacy. It is also likely that historical observers would forego unremarkable cases of ASD in silence for lack of appropriate terminology and/or because the tendency to label aberrations from the average mindset, was not as prominent as it is today.

The term autism was coined in the early 1900's by the Swiss psychiatrist Eugen Bleuler, although its denotation was not exactly the same as during the later part of the century.

The actual study of what we now think of as autism and related conditions, was pioneered by Grunya Sukhareva[8] in the Soviet Union, Hans Asperger in Austria and Leo Kanner[9] in the USA, during the first half of the twentieth century.

Pathophysiology

The understanding of the mechanisms behind ASD is not complete, and there are likely different elements that contribute to its aetiology. Several hundred genetic variations that correlate to the condition have been identified but none of them are present in all cases [24]. Nevertheless, the heritability of ASD is high [25, 26].

There is also a wide range of environmental factors that have been proposed, including among others infections in utero, prenatal stress, perinatal complications and teratogenic chemicals [27–30].

Clearly, outside of being caused by a wider syndrome such as for example Fragile X syndrome or Rett syndrome, the exact cause for ASD is hard to pinpoint in the individual case. There are, however, some patterns that are common and strongly correlated to the severity of symptoms in functional brain connectivity studies. In adults on the spectrum, there are fewer long-range connections between areas in the cortex but more short-range connections within each area. There is also a tendency to favour the left hemisphere, meaning that persons with ASD perform some tasks that are usually mainly associated with the right hemisphere using their left hemisphere.

[8] Transcription of Груня Сухарева

[9] Born in the present-day Ukraine.

These differences have little or no impact on structural images but can be observed by electroencephalography (EEG), magnetoencephalography (MEG) and functional magnetic resonance imaging (fMRI). This is also the rationale for investigating augmentation in conjunction with fMRI in paperr I, in this thesis.

Parkinson's disease

Parkinson's disease (PD) [31] is a neurodegenerative disorder which affects $1 - 2$ ‰ of the population at any given time. It usually presents itself after the age of 60 and is not curable, but often manageable with medication for extended periods of time.

The symptoms of PD is primarily parkinsonism, a state where the substantia nigra fails to give sufficient dopaminergic stimulus to the striatum, thereby disrupting the latter's function primarily in planning and initiating movement — thus causing tremor, slowness of movement, rigidity and a slouching posture — although other aspects of the striatum are affected as the deficit is exacerbated. In PD this is caused by the accumulation of insoluble α-synuclein — a protein that normally mediates the release of signal substance from vesicles in the presynaptic axon terminal — into so called Lewy bodies.

In later stages, symptoms can include dementia, depression, dysfagia, impaired olfaction and vision, insomnia and problems with the autonomic nervous system leading to uneven blood pressure with a tendency to faint, problems with micturition and defaecation, among others.

Alzheimer's disease

Alzheimer's disease (AD) [32] is a neurodegenerative disorder and the most common cause of dementia (60–70% of all dementias). As of 2023 it has a prevalence of around 1% in Sweden, though it is likely to rise as the population grows older on average.

The disease presents itself after the age of 65 in 90% of cases. AD comes in two varieties, familial which makes up 1–2% of cases and sporadic which is the rest. Familial AD is inherited in an autosomal dominant pattern with high penetrance (>90%) and tends to strike at a younger age and progress faster. Sporadic AD, while not having such a distinct inheritance pattern, still show a 70% inheritability.

Symptoms mainly consist of different forms of cognitive impairment from short term memory loss to a total apathetic passivity. Death mainly occur through complications of bedriddenness such as infected pressure ulcers and embolisms or opportunistic infections.

The direct cause of the cognitive symptom is the demyelisation and atrophy of neurons in the brain, which in turn is linked with axonal formation of τ-protein tangles and degradation of axonal microtubules with subsequent formation of extracellular amyloid plaques from peptides from the degradation of a certain neuronal membrane protein, mediated by the microtubule break down. The plaques could conceivably cause recruitment of immune cells with the ensuing inflammatory process causing demyelisation of axons and possibly a positive feedback in the modification of the τ-proteins that make them destabilise the microtubules.

3

MRI

"I suggest the Clark Nova Portable. It has Mythic Resonance."

Mugwump to Bill Lee

Magnetic resonance imaging (MRI) has provided clinicians and scientists the possibility to make high resolution, 3D images of living humans and other objects, without having to use ionising radiation. It works by the same mechanism as nuclear magnetic resonance spectroscopy (NMR), utilising the quantised intrinsic magnetism of atomic nuclei, and it incorporates much of the technical advances made in that field since the introduction of the first commercial NMR spectrometer in 1952 [33].

Because MRI scanners can send very accurate signals, while the time frame of the decay is typically such that it can be followed with good precision and the excited nuclei have time to interact with their environment in different ways, there are many ways to modify the procedure. For example, during the time the nuclei are excited they interact with haemoglobin differently depending on if it's oxygenated or not, which form the basis for functional MRI (fMRI), with which one can map what parts of the brain are active over time or how the parts work together [34–36]. Nuclei also have time to travel during the decay time, either in a moving medium like blood or by diffusion. This can be used to measure flows or diffusion in different ways. Diffusion MRI (dMRI) is for example one method employed to study brain connectivity [37]. Both fMRI and dMRI could be considered to generate 4D datasets; for fMRI the fourth dimension is time and for dMRI the fourth dimension represents the different directions in which the diffusion is being measured.

3.1 Physical Basis

The basis for MRI is the magnetism of particles. Particles have spin states, that is to say a finite number of states they can be in. This gives them intrinsic magnetism. The number of spin states m is in turn determined by a particles' spin quantum number I.

$$m = 2I + 1 \tag{3.1}$$

Apart from spin state, the magnetic moment is also influenced by magnetic moment from its electric charge in conjunction with its spatial wavefunction, which also has a finite number of states, controlled by other quantum numbers. The particle with the strongest spin is the electron which has $I = \frac{1}{2}$ and therefore $m = 2$ different states it can be in. Ferromagnetism is caused by the magnetism of electrons.

Quantum States

All particles that make up matter are fermions. That means that in any system no two particles can have exactly the same quantum numbers, i.e. quantum state. This is called the Pauli exclusion principle. Particles that don't follow this rule are called bosons and don't make up matter. They typically mediate forces like the photon, gluon or Higg's boson does.

That means that an electron will pair up with another electron with the same quantum state except for opposite spin if at all possible, because any other unoccupied state would have a higher energy. The exceptions would occur if we had unpaired electrons, either because a whole system only have an odd number of electrons or because several quantum states (disregarding the spin) have the same energy in which case a pair of electrons can have the same spin because they differ in some other quantum number. If we then reintroduce the spin, the electrons could either point in the same direction or in the opposite direction and these two states would differ in energy. Typically the opposite spin a.k.a. the singlet state, would be the most favourable one, but there are exceptions such as the dioxygen molecule, which naturally occurs in its so called triplet state, where both unpaired electrons have the same spin. Atoms, molecules or ions with an odd number of electrons and not very delocalised wavefunctions, tend to react in some way with each other forming species with an even number of electrons. Electron resonance is therefore not something that is commonly used in medical imaging.

How about other fermions? Out of the particles we have lying around, only up and down quarks occur in any larger amounts. However, they do have $I = {}^1\!/\!_2$ like the electron, giving them 2 possible spin states. Here we are in luck because the aggregation of three quarks, either two ups and one down to form a proton or two downs and one up to form a neutron is energetically favoured. Do not undertake that reaction at home! In both cases we end up with an unpaired quark. That means that also protons and neutrons have $I = {}^1\!/_2$.

In many cases it is — on the whole — tidier to organise one's protons and neutrons into atomic nuclei[1]. Here is another chance for spins to pair up. A nucleus with an even number of protons and an even number of neutrons would not have any spin and we would not be able to magnetically separate spin states, because it would only have one. This is also often the case. For example the most common isotope of carbon has six protons and six neutrons. If it had an odd number of either protons or neutrons we would again end up with $I = n + \frac{1}{2}$ but $n \in \mathbb{N}_0$ would not necessarily be 0. If both these numbers were odd we would get $I = n, n \in \mathbb{N}_0$[2], i.e. an integer, and thus an odd number of spin states.

Though the field felt by each type of nucleon isn't obvious, it is still possible to separate the Schrödinger equation and therefore its solution in a spherical and a radial part, wherefore we get more states to separate as we populate higher energy levels. The unpaired nucleon of either type will simply occupy the lowest energy state that's left. Because the spherical part being the same as for electron orbitals, the same levels of degeneracy could be expected. This is not exactly true for reasons given below.

Because of differences in the radial components of the field the relative sizes of energy separations associated with the different quantum numbers are different. Also a spin–orbit interaction term is added to the hamiltonian in the nucleon case, which split up all but the states with a constant spherical part, that is with an angular momentum quantum number $l = 0$ – often called s- orbitals. This is not a fundamental difference from the electron case but because of the relative magnitude of the interaction term for electrons, it can be ignored with little consequence. There is also the caveat that all wavefunctions here mentioned are solutions for one particle. That the order of the levels remain the same as we fill them up is an assumption. All interactions of n particles should ideally be described by a field in $3n$ dimensions[3]. Obviously, that would make calculations very difficult. For a comparison of some states for nucleons vs electrons see Figure 3.1.

[1] Also these reactions can be exothermic so take measures to dissipate the extra energy.

[2] While no isotopes with odd both neutron and proton count with spin 0 are known, it is not theoretically impossible. Several isotopes that have not been synthesised are predicted to have these properties.

[3] Unless it is time dependent, in which case it would need a time dimension as well.

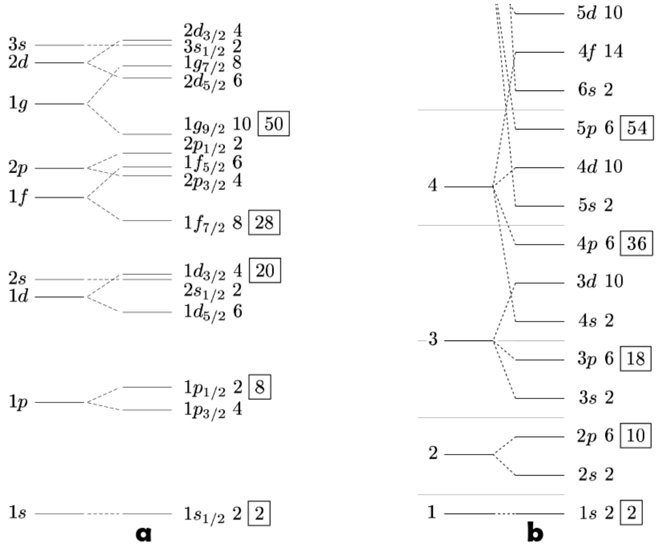


Figure 3.1: Schematic representations of some energy states in (**a**) nucleons and (**b**) electrons. Note that we are here mainly interested in the order of orbitals. The distances are not proportional. Especially, the energies in (**a**) and (**b**) should not be understood as equal at the same height. The nucleon's energies are many orders of magnitude higher.

From the diagram (**b**) we would for example see that the element with atomic number 54 should be quite chemically inert because it would have a full outer shell of electrons. This element is xenon, which is a noble gas and is inert in almost all conditions. Correspondingly, we could look in the diagram (**a**) and say that the element with atomic number 20 would have a full shell of protons. That would make this element very stable. What would that mean? It has nothing to do with oxidation states and so on, it's the nucleus that is stable. We should expect this element to be more abundant than its neighbours and have more stable isotopes. The element is calcium and makes up around 5% of the earths crust. The same number is 1.5% and 0.0026% for its left neighbour potassium and its right neighbour scandium, respectively. Calcium have 5 stable isotopes and one that decay with a half-life of $6.4 \cdot 10^{19}$ years, all of these occur in nature. Potassium have 3 naturally occurring isotopes out of which two are stable and one has a half-life of $1.248 \cdot 10^{9}$ years and scandium only has one stable, naturally occurring isotope.

What spin would we expect $^{45}_{21}$Sc to have? It contains 24 neutrons so they will not give any contribution because of pairing. It has 21 protons so the 20 first will fill up the three innermost shells and the last one should end up in the $1f_{7/2}$ orbital and its spin is in fact $7/2$. What about $^{73}_{32}$Ge? It has 32 protons so they are all paired up. That leaves 41 neutrons. 28 of these fill the first four shells. Then 4 goes in $2p_{3/2}$, 6 in $1f_{5/2}$ and 2 in $2p_{1/2}$. This leaves one unpaired neutron which ends up in $1g_{9/2}$ and $^{73}_{32}$Ge has spin 9/2.

Clearly there are many nuclei that theoretically could be used in medical imaging. Outside of very specialised techniques, however, the most common nucleus in the human body, $^{1}_{1}$H is used.

Precession

As we saw above, opposite spin states are normally degenerate, i.e. have no energy difference. That would mean that they would always be equally populated, leading to no net magnetisation.

In the presence of a strong magnetic field, however, the spin states are prised apart, energetically. In fact, the energy difference between two states is proportional to the applied field B_0, which could be expressed as:

$$\Delta E = \hbar B_0 \gamma \Delta J \tag{3.2}$$

ΔE is here the energy needed/released when the spin state is increased/decreased by ΔJ. The constant γ is a characteristic of the particular nucleus in question and is called the gyromagnetic ratio, $\hbar \triangleq h/2\pi$ is the reduced Planck constant where $h \triangleq 6.62607015 \cdot 10^{-34}$J·s is the Planck constant by the definition of the SI system.

If we assume that nuclei can spontaneously flip between spin states with a certain probability and assume that they have reached equilibrium, we know that the occupancy of the energy states follow a Boltzmann distribution. The ratio of nuclei in a lower l and in a higher h energy state is thus:

$$\frac{N_l}{N_h} = e^{\frac{\Delta E}{k_B T}} = e^{\frac{\hbar B_0 \gamma \Delta J}{k_B T}} \tag{3.3}$$

Here $k_B \triangleq 1.380649 \cdot 10^{-23}$ J·K^{-1} is Boltzmann's constant by the definition of the SI system. In typical cases this means that the excess of nuclei in the lower energy state is only a very small fraction. The resulting field of that excess is nevertheless enough to measure.

While every nucleus only can be in a limited amount of spin states, the bulk magnetisation is not restricted in this way. On the contrary, the more particles we consider together, the better the classic Newton–Maxwell mechanics approximation gets. We can thus normally see the equilibrium bulk magnetisation as a vector parallel to the z-axis (by convention) in a three dimensional coordinate system. For reasons that will become clear below, we will introduce a coordinate system that has this z-axis constant but the x- and y-axis spinning with an angular frequency of:

$$\omega_0 = \gamma B_0 \tag{3.4}$$

This is called the Larmor frequency. If we apply a magnetic field B_1 rotating at this angular frequency, that is standing still in our new coordinate system, we will rotate the bulk magnetisation about the field an angle proportional to the time integral of B_1. In other words this is the frequency that the magnetic dipole of the kernel naturally precesses. Often the rotation angle, or flip angle as it is known, is all one needs to know, so a B_1 pulse can be referred to as a 90°-pulse, for instance, without specifying the exact shape. We note that small deviations to precession frequency will make this rotation ineffective if it takes place over long enough time, effectively bending the magnetisation back half of the time. From this we can see that we can make pulses strike broadly, affecting all nuclei of a certain type, by applying short high-intensity pulses. Conversely, we can affect a more specific subset of nuclei by applying long low-intensity pulses.

Excitation and Relaxation

Let's consider a nucleus with spin ½, or if you will with two spin states.

When the bulk magnetisation has a component in the xy-plane, it precesses and we talk about nuclei being excited. This also causes a signal that could be picked up by an antenna, which is how instruments such as MRI scanners or NMR spectrometers acquire their input.

If we wait, we'll see the registered signal die out in what looks like an exponential manner. This is called relaxation. It has two components.

If we rotate the bulk magnetisation into the plane, it is obvious that equation 3.3 no longer holds true. Because of the z-component of the bulk magnetisation being zero, the ratio must in fact be one. This means that the states do not hold their equilibrium distribution. By spontaneous energy exchange with other particles this equilibrium will be restored. This is called longitudinal relaxation and proceeds with the time constant T_1. This kind of relaxation follow the formula below:

$$M_z(t) = M_z^0(1 - e^{-t/T_1}) + M_z(t_0)e^{-t/T_1} \tag{3.5}$$

The equation 3.5 describes the T_1 component of the relaxation. The time t_0 is at the start of the relaxation, but after the excitation.

The second kind of relaxation is called transverse relaxation and has the time constant T_2. This is the decay of the signal because of several *random* processes that all contribute to the loss of coherence of magnetic dipole moments perpendicular to the z-axis.

$$M_r(t) = M_r(t_0)e^{-t/T_2} \tag{3.6}$$

t_0 is defined as above. Because the transversal parts behave the same, I use r to mean either x or y and will continue to do so whenever applicable.

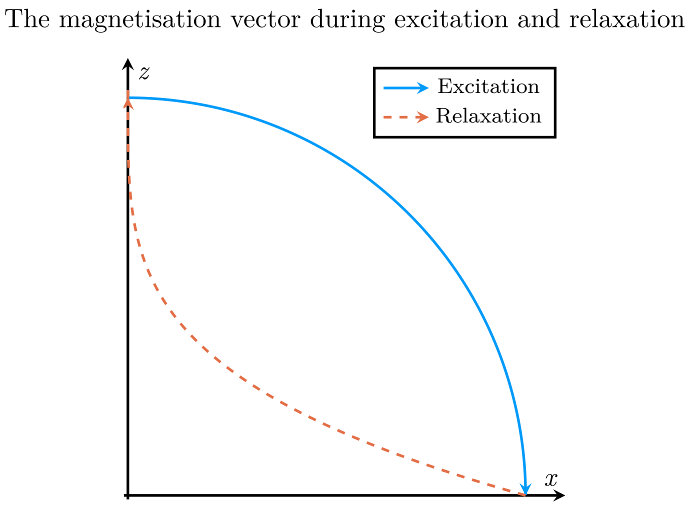

The magnetisation vector during excitation and relaxation

Figure 3.2: Illustration of the magnetic dipole vector (in a rotating frame) precessing 90° about the y-axis (not seen in illustration) during excitation and how it decays back to its equilibrium state during relaxation.

These random interactions can be temporary fluctuations of the field experienced by different nuclei or dipolar interactions between nuclei or indeed the loss of transverse magnetisation accompanying T_1 decay.

The combined T_1 and T_2 decay of the bulk magnetisation vector is illustrated in figure 3.2.

If one would make a 90°-flip and register a decaying signal we could empirically determine the time constant. We call that time constant T_2^*. It is always less than T_2. This is because of yet other reasons for nuclei to get out of phase. T_2^* is caused both by the random fluctuations described by T_2 and other *non-random* differences in the environments of individual nuclei.

This means that to measure T_2 we have to give a 180° pulse after a certain time and then look at the amplitude after the same amount of time again.

We would be left with the decay due to random dephasing but would have reversed the effects of non-random dephasing.

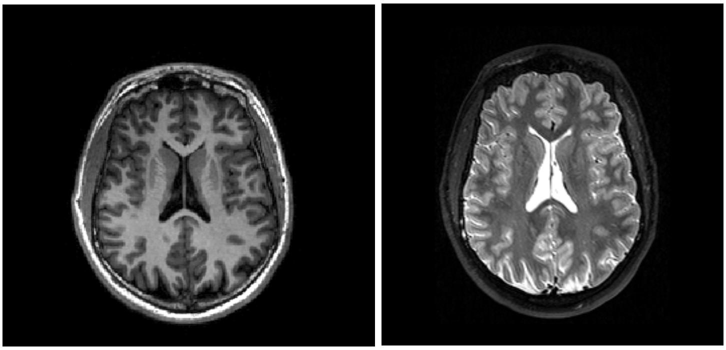


Figure 3.3: An example of T1-weighted (left) and T2-weighted (right) images of the brain of one subject.

In medical imaging, the acquisition sequence is often adjusted so that the contrast in voxel intensities reflects variations in either of these time constants. This often gives good contrast between relevant tissue types. Specifically, T_1-weighting is good for looking at most soft tissues and can be regarded as the default. It gives a good contrast between grey and white matter in the brain. T_2-weighting is particularly good at showing inflammation, since inflamed tissues have a higher water content, which in this case makes them brighter. See Figure 3.3 for one example of T1- and T2-weighted images.

MR Signals

There are several different processes that can be observed in the acquired signal. Below some of them are described.

Free Induction Decay

The free induction decay or FID is the signal observed in the xy-plane after an excitation, as the nuclei precess freely until they return to their equilibrium state. The relaxation is approximately exponential. As mentioned earlier, the time constant is T_2^*, but one must take into account that the coils don't rotate with the Larmour frequency so the actual signal will be:

$$M_{\mathbb{C}} = M_r(t_0)e^{-t/T_2^*}e^{i\omega t} \tag{3.7}$$

Where $M_{\mathbb{C}} \triangleq M_x + iM_y$ is the identification of the magnetisation in the xy-plane with the complex plane.

Spin Echoes

This is a component of many pulse sequences where a 180°-pulse is applied halfway through the evolution time. The pulse reverses all spins, causing nuclei that have gone out of phase to refocus, provided that the variations in the local magnetic field are stable.

3.2 The MRI Scanner

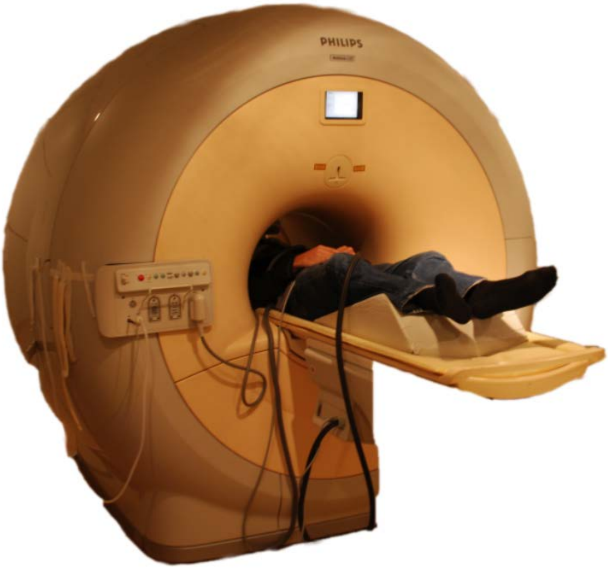


Figure 3.4: A 1.5 Tesla MR scanner here used for neuroimaging.

The MRI machine itself consists of a large magnet with a narrow hole or bore in the middle, see Figure 3.4. Inside the bore a strong and highly homogeneous static magnetic field can be achieved. Field strengths mostly lie in the range 0.5-3.0 T, though magnets for MRI-use are available — though very expensive — with field strengths up to several times that, for human use. The bore is as a rule horizontal and typically has a motorised table that slides in and out of the bore. The very strong magnetic field is usually caused by a current running through a superconducting coil around the subject in a lying position. The magnetisation of the permanent field is thus in the direction from feet to head (or vice versa).

In the wall of the bore are several non-superconducting coils in different orientations. They can be used to interact with the magnetic field in different ways. Some compensate for inhomogeneities of the permanent magnetic field, some are there to generate gradients in different directions and crucially one needs a coil to send the radio pulses and possibly acquire the resulting signal. Often special receiving antennas adapted to the part of the body being examined, are placed on or around the subject. The housing of the electromagnet typically has several layers of insulation and cryostats to keep the coil superconducting. Most materials used in the coils require liquid helium cooling, but to prevent the helium boiling off too quickly, the outside of the helium cryostat is often cooled by a liquid nitrogen cryostat.

4

Machine Learning and Data Analysis

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

Tom M. Mitchell

Machine learning (ML) is the construction of algorithms that can learn from data and generalise its "knowledge" to new situations. The quote above by Tom Mitchell formalises this somewhat.

This technology, and especially deep learning (DL), has gained massive popularity over the last couple of decades due to advances both in the construction of models, and in the hardware used to run them (especially graphics cards, or GPUs). Another factor is the availability of large open data sets, which is especially important in the medical imaging field since sharing of sensitive data can be complicated due to ethics and GDPR [38]. In the literature [39], DL has for medical images mainly been used for classification (e.g. healthy or diseased) [40] and segmentation (to for example save time in the clinical workflow) [41]. In this work, DL was used to process medical images with a view to inferring clinically relevant information from them, focusing on brain diseases.

4.1 History and some old techniques

Machine learning goes back to the early days of computing. The first rudimentary neural network model was suggested at a theoretical level by Canadian psychologist Donald Hebb in 1949 [42, 43]. The simulation of such a network had to wait until the early sixties because of the state of computing at the time. Nevertheless, several more specialised learning models aimed at playing particular games was implemented in the intervening time [44, 45].

While neural networks remained of theoretical interest, their use struggled with practical problems such as numerical instability and high computational demands during much of the twentieth century, and other techniques, such as support vector machines and decision tree learning among others, became popular.

A support vector machine is originally an attempt to find a linear condition $\mathbf{w}^\mathbf{T}\mathbf{x} - \mathbf{b} = \mathbf{0}$ that separates the data points ($\mathbf{x}_i \in \mathbb{R}^n$) as well as possible [46], but non-linear transformation of the vector space by means of positively (semi-)definite kernels has made the method more generally applicable. Extensions for regression analysis and multiclass classification has also been developed.

A decision tree is a way of hierarchically arrive at a classification or formula for some attribute (in which case it is often referred to as a regression tree) to be inferred for a data point [47]. Decision tree learning deals with algorithms saying how and when a decision (or regression) tree should be refined, given a set of prelabeled training data.

4.2 Principal Component Analysis

One way of finding patterns in a given collection of vectors of a fixed length (that could for example represent an image of a certain size) is to employ principal component analysis (PCA).

The covariance matrix $\mathbf{R}$ of all data $\mathbf{X}$ (where each of n datapoints $\in \mathbb{R}^m$ is a column in $\mathbf{X}$) is generated and then diagonalised by singular value decomposition, so that a set of orthonormal vectors in the data space is generated (as columns of the diagonalising matrix $\mathbf{V}$). This can easily be done in such a way that the vectors are determined in order of diminishing corresponding eigenvalue, which is practical, because in most cases, only the first few vectors are needed.

$$\mathbf{R} = (\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T, \text{ where } \mu_{ij} \triangleq \frac{1}{n}\sum_{j=1}^{n} X_{ij} \tag{4.1}$$

$$\mathbf{R} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T \tag{4.2}$$

By using the first few of these vectors as a basis, the data points can be represented with high fidelity in a new space with massively reduced number of dimensions.

Alternatively, seeing as a few principal components contain most of the variation in a data set, they could conceivably be used as a proxy for the whole set. I take this approach in paper III where principal components of slices of a brain volume are used in lieu of the whole volume, as input to a convolutional neural network estimating brain age.

4.3 Preprocessing in general and in fMRI

When using medical images with statistical methods and/or ML it is often necessary or at least beneficial to preprocess them. ML generally requires less preprocessing because as the model "learns" it can, to a certain degree, do some of the preprocessing steps as part of the learnt transformation.

A T1-weighted structural MRI volume, may need some interpretation. Using a toolkit such as FSL [48–51], a segmentation of the brain volume by tissue type (gray matter, white matter, cerebrospinal fluid) may provide ML models with information that would be difficult for it to learn directly from a training set. In cases where this is not done, it often helps to scale the intensity to lie between zero and unity, though different normalising transformations that often are used repeatedly in modern ML models, make this less important,

For more traditional statistical methods, preprocessing usually include some sort of registration to an atlas, so that all the brain volumes are in the same place. This usually requires non-linear deformation. This is not necessary when using convolutional neural networks, as they are translation tolerant.

Another form of preprocessing, that might be important in more traditional computations but rarely are necessary for ML, is noise reduction. This is often accomplished by means of a simple low pass filter, though more advanced techniques such as adaptive filtering might be necessary in some cases.

In functional imaging (e.g. fMRI), metabolic activity in the brain is seen over time. This requires more preprocessing. Firstly, it is unlikely that a subject stays perfectly still for the duration of the scan. This can at least to some degree be fixed using head motion correction. Secondly, rather than acquiring an entire volume in an instant, slices are acquired more or less continuously. To get volumes representing the same time throughout, some sort of interpolation in time must be carried out. Thirdly, the signal could be overlaid with drifting trends and other confounders, which must be removed usually by including them as regressors in the data analysis step. On top of this, previously mentioned steps might also be needed. The registration of each time frame is often done by mapping to a structural volume of the same brain by rigid motion transformation, and combining this transformation with the previously mentioned mapping to an atlas.

The intensity values in an fMRI dataset correspond to metabolic activity in a voxel at a certain time[1]. This is usually not, in itself, very easily interpreted. What is more enlightening is how the activity time series correlate to some task that is carried out or indeed to time series of other voxels.

Depending on point of view, one can regard the subsequent statistical analysis that typically uses correlation of time series in some form to obtain a certain measure for each voxel[2], collapsing the time dimension, as a last preprocessing step before using the data to train a convolutional neural network.

In theory the ML model could use the four-dimensional data and learn directly from it using a stack of four-dimensional convolution layers. This would not be practical with present day hardware because of the enormous amount of data that would have to be processed.

4.4 Neural Networks

Inspired by a fledgling understanding of how our brain works, neural networks try to use a minimalist model of nerve cells with some adjustable parameters, typically organised in layers, to learn tasks.

A real life neuron just has one efferent neurite, the axon. It typically branches and attach to different places but the output is the same. On the afferent side however a neuron has synapses with many others, by means of their axons attaching to the afferent neurites, the dendrites. We assume that the neuron in some way integrates its in-signals to make an outsignal. We also assume that it can be tweaked in its respons to individual in-signals to effect learning.

The popular model of a neuron in a neural network, shown in figure 4.1, is thus something that adds its in-signals x_i all weighted by some real number w_i[3](possibly including a constant term b as well). This result is then usually

[1] Strictly speaking it is the metabolic activity during a period slightly before the acquisition time. Normally this is adjusted by convoluting the assumed hemodynamic respons into the activity regressor.

[2] Such measures include for example the correlation to low frequency oscillations, correlation with adjacent voxels and correlation with mirrored voxel (in opposite hemisphere)

[3] For implementation purposes, a floating-point number.

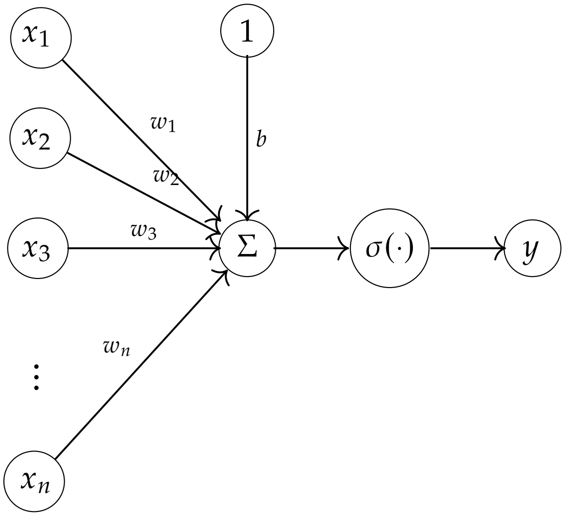


Figure 4.1: A simplistic mathematical model of a neuron.

passed through a non-linear activation function σ and that is the output y of the neuron.

$$y = \sigma(\mathrm{w}^T \mathrm{x} + b) \tag{4.3}$$

Equation 4.3 contain the same information in a formula, for sake of clarity and for the benefit of the less visually minded.

This is the basis for all artificial neural networks. The difference comes of whence the inputs are taken. In Principal neurons such as described before could be connected any which way, and while it would make it much harder to think about, all the necessary calculations that will be discussed below, would — in Principal — be possible to carry out.

Dimensionality

While it is not a part of the theory of neural networks as such, some conventions about what dimensions are considered to exist in the data, should be mentioned. This is largely to establish a clear terminology for the rest of this chapter. A data set has a number of data points, often accompanied by a label for each data point. A data point could be many things such as a picture, a sentence, a brain volume from an MR scan and so on, potentially passed through and transformed by a set of layers. As this data point is processed it often acquires channels[4], which is an extra dimension in the data point, which can be seen as "parallel versions". Sometimes it can be useful to consider the data points to have channels already before they are passed to the network.

Usually it is most efficient to let a neural network handle multiple data points as a unit. These units are called batches and the data points each has an index in the "batch dimension".

Apart from this the data point usually have an inner structure, which the network needs to be aware of. This can typically be a number of spatial coordinates and/or time. This is what is normally referred to as the dimensionality of the data.

I here define an element of a data point as a subset of the data point with fixed coordinates in all dimensions except for the channel dimension.

Layers

In practice, networks are usually organised into layers, each of which get one input (one datapoint transformed by zero or more layers) and using this and zero or more neurons (each with their own weights) and in some cases other information, such as for example random numbers, make one output. Note firstly that this "one" input or output could be of a very high dimension and secondly that often several consecutive data points are treated together as a batch, but this does not let the information from one data point interact with that of another, as it flows through the model[5].

Dense layers

Dense layer is a commonly used term for what is sometimes more descriptively called a "densely connected layer". This is perhaps the easiest way of imagining the organisation of a layer. The output of the layer is the output of all its neurons, and all of the input goes to each neuron so that the interconnection between a layer L_i and a following dense layer D_{i+1} is the complete bipartite graph $K_{|L_i|,|D_{i+1}|}$. This means that the dense layer needs $|L_i| \cdot |D_{i+1}|$ weights[6]. A network consisting of such layers is shown in figure 4.2.

[4] To avoid confusion, the term "feature" will not be used when dimensions of tensors are discussed, as it overlaps substantially with the term "channel". To the extent that the term is used, the denotation is the same as for "channel", but with a slight nuance difference in connotation.

[5] There are some exceptions, such as batch normalisation, where the data flow in principle is (very weakly) interconnected.

[6] Or $(|L_i| + 1) \cdot |D_{i+1}|$ if a bias should be included and a constant isn't considered part of the output of L_i.

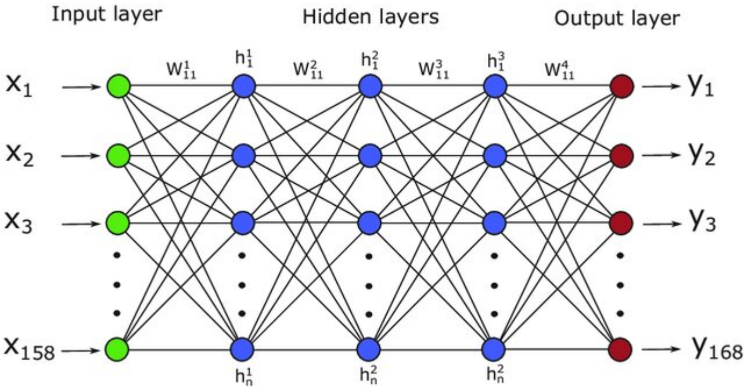


Figure 4.2: A network of densely connected layers. Activations are not shown. Note that this network has four layers of this type, as indicated by the four sets of weights. The "input layer" does not do any calculations, but is needed in many frameworks to import the data to the opaque structures that are used when the model executes. Image from [52], available uncer CC license.

This type of layer can express a very wide variety of transformations, but on the other hand uses much memory and computational power for its large number of weights. When processing for example images, a dense layer would require weights in the order of magnitude of the number of pixels squared. Clearly this is only viable for very small pictures.

Convolutional layers

In this type of layer the output is a convolution of a kernel of weights and the input, optionally plus a bias term. For this reason, convolutional layers only have to learn a small number of filters (kernels), which is much more efficient. The discrete convolution is here defined as:

$$(f * g)(b, n, c_{out}) = \sum_{\forall m, c_{in}} f(b, n - m, c_{in}) g(m, c_{in}, c_{out}) \tag{4.4}$$

$$f, g \in \mathbb{N}^{d+2} \rightarrow \mathbb{R};\ n, m \in \mathbb{N}^d;\ c_{in}, c_{out} \in \mathbb{N}$$

Here f is the data point entering the layer, g is the kernel, n is the coordinate in the dimensions the data intrinsically has, for example two for a (grey scale) picture, and d is the number of such dimensions. The summation index should be understood as: all values for which the summand is defined, in other words, for which neither index fall outside its respective tensor.

This is just the typical variety of convolution layer but there are many variations. One can take longer steps in the convolution, possibly different step lengths in different dimensions, scaling down the data. Alternatively the kernel could spread out the information from one element over larger, possibly overlapping (in which case the resulting value is the sum of all points covering it) areas for scaling up. The latter is often referred to as transposed convolution or inverse convolution, neither term being strictly correct.

There are also variations on what to do near edges, where the overlap of the kernel is not complete. Should some other value such as nearest value, mirrored value or zero be used for values outside the data, when convoluting, or should such elements not be part of the output?

Convolutional layers are especially important in treating multidimensional data, such as images, videos, neuroimaging volumes, among others. In this way only enough parameters to make up a small kernel, need to be fitted. The built-in translational invariance is also often a boon.

Activation layers

Often this is referred to simply as an activation function, which makes sense, especially for stateless simple functions. The point is that the function is applied to each value passed to it. This is done to break up the linearity of the neuron, whether it be part of a dense layer or a convolutional layer.

The point is that any combination of linear transformations is linear, which means that a network without non-linear activations would have a very limited repertoire.

Commonly used functions include the rectified linear unit, often abbreviated ReLU, and variations thereof.

$$\mathrm{ReLU}(x) = \begin{cases} 0 & x < 0 \\ x & x \leqslant 0 \end{cases} \tag{4.5}$$

Often activation layers have the function to shape the data in some way. If one wants a model to express a probability for example, it can be good to use a sigmoid[7] activation function on the output layer. Likewise if one wants a vector of n mutually exclusive probabilities, such as with a classification, the softmax[8] function can be employed.

[7] $\mathrm{sigmoid}(x) = \frac{1}{1+e^{-x}}$

[8] $\mathrm{softmax}(x_i) = \frac{e^{x_i}}{\sum_{k=1}^{n} e^{x_k}}$

Auxiliary layers

There are also usually a plethora of other layers that are useful in different situations. Different normalisation layers do a transformation of the data — usually affine — to change the average values and dispersion in a way that makes the training more efficient. Pooling layers reduce the dimensions of an image by saving the mean or the maximum of a small neighbourhood. A way of regularising activations can be to put in dropout layers in one's network. Such layers pass on as much activation as it receives on average by zeroing a fraction φ of the inputs but also divide each input by $(1 - \varphi)$.

Convolutional Neural Networks

In processing images, convolutional layers have a possibility to learn to produce many useful features, one in each channel coming out of a layer. Typically an early layer will home in on some simple features. Exactly how simple, is governed by how large the kernel is, but one can imagine things such as lines and edges with different orientations. In general, later layers learn more and more advanced features. At the end of a convolutional neural network (CNN) is often a dense layer, to perform classification or regression. See Figure 4.3 for a typical CNN.

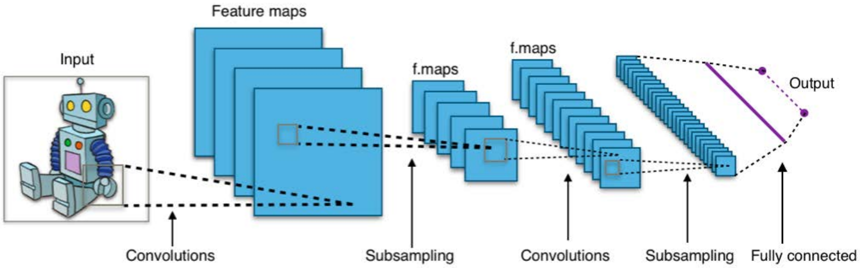


Figure 4.3: A typical CNN, consisting of convolutional layers (which perform convolution with many filters), pooling layers and a final fully connected dense layer which performs the classification or regression. CNNs can today contain hundreds of layers, and therefore require large datasets for training. By Aphex34 - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=45679374

CNNs have been used to a large degree in the related computer vision field, where it is rather easy to create or locate large datasets. During the last decade, the CNNs have become deeper and more complex (e.g. more than 100 layers), thereby requiring more training data, and CNNs pre-trained on the large ImageNet dataset [53] have therefore instead been fine tuned on smaller datasets. However, many applications in computer vision use 2D images and therefore 2D CNNs, while neuroimaging data is 3D or even 4D (fMRI and dMRI). The number of papers using 3D CNNs is substantially lower, and there are few available pre-trained 3D CNNs. Regarding augmentation, most deep learning frameworks have built-in support for 2D augmentation, but not for 3D augmentation [54]. For this reason, two of the papers in this thesis use 2D projections from 3D volumes, inspired by Langner et al. [55], to be able to use more mature 2D CNNs and to substantially lower the training time.

4.5 Loss and Optimisation

We have so far seen how an artificial neural network can transform data with different operations, and we know that these are governed by a large number of weights, so they should be able to accomplish many tasks given the right weights. This begs the question, how should the weights be set?

Firstly there should be a loss function. This is a function that increases as our model in some way strays further from the goal. Alternatively if we have a function that measures goodness, we can use the negative of this as loss. In supervised learning, this typically means that every result is in some way compared to a gold standard, previously known labels for the training data.

If we want a model to estimate a quantity y, we might choose as a loss function, calculated on a batch of size n:

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Here y_i represent the true values and $\hat{y}_i$ the model's estimates. Consider the loss on a certain batch to be a function of all model parameters θ. To simplify we also consider a linear stack of layers without branching or joining. This is mostly for notational convenience. It is still possible to calculate the gradient with respect to all parameters in a network with a more complicated topology. We then know the direction in which the loss decreases the fastest, locally, in the parameter space, as $-\nabla_\theta \mathcal{L}$. This gradient, in turn, can be calculated, because:

$$\frac{d\mathcal{L}}{d\theta_i} = \frac{d\mathcal{L}}{da_n} \left(\prod_{k=i+1}^{n} \frac{da_k}{d\theta_k} \cdot \frac{d\theta_k}{da_{k-1}} \right) \cdot \frac{da_i}{d\theta_i}$$

Where a_i is the activations and θ_i the weights from layer i. After having calculated the direction of steepest descent we are ready to make a small adjustment to the parameters, which will ideally cause the loss to be lower (at least on average) on the next batch. A simple way to update the parameters is by the formula:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_\theta \mathcal{L}$$

Where η is the so called "learning rate". The simple logic is that changing the parameters in this direction decreases the loss, locally. Depending on how long step we take, the gradient changes more or less and with a big η we may end up far past where we should have stepped to. This is why hyperparameters like η often have to be tweaked manually.

The formula above is in no way the only alternative. As long as the parameter step is in a direction that is less than 90° from the gradient, it should decrease the loss, at least for small enough η. Frameworks typically let you chose between ready-made algorithms, with some variable parameters, or possibly writing one yourself.

These optimisers usually have state in addition to η. This can be used to remember the last direction and then upgrading the parameters along a direction which is a weighted average of the computed gradient and the last step direction. This implements inertia in the search for lower loss. This turns out to be beneficial in many cases.

4.6 Ensembles

As ML models train, we see that they learn both the distribution from which the training data is sampled, but also particularities to the specific observations in the exact training set that was used. This leads to a certain amount of overfitting. That is to say that models are almost always at least a little bit better when exposed to the actual training set, than when exposed to some, in principle equivalent, test set.

It is then quite likely that different randomisations of the same model could pick up, at least partly different, training set specific information.

By the same token, a set of approximately equally good but non-identical models could perhaps learn more mutually complementary training set specific information.

With this motivation, averaging the output of several models should be expected to be, at least a little bit better, than each model on its own. This is the idea behind ensembling.

An ensemble is in this context, a set of separately trained models whose output is, in some way, weighted together. In paper III this is tested and a certain improvement noted, with regression models estimating brain age, where the aggregation method consists of taking the arithmetic average of the models' estimations.

When doing classification tasks, an alternative could be to let the individual models vote which class a data point should belong to.

4.7 Julia

In my work with ML, I have come to use the relatively new programming language Julia [56]. As this is not the most common choice, and as it has come about for a specific reason, I will touch upon the subject briefly.

When experimenting with ML models it is typically necessary to implement them in some kind of programming language. The most common language used in ML as of 2023 is Python. It is probably preferred because it combines an easy and clean syntax, good and clean abstractions such as classes, generators, closures, et.c. and the convenience of debugging environments with rich introspection facilities.

The Problem

Because Python is an interpreted language, the code that runs highly optimised numerical operations, possibly utilising specialised hardware such as gpu's, has to be written in some suitable low-level language and then linked to Python. Often, general numerical computations may be done using the NumPy-library. The ML aspects are typically handled by some kind of framework that enables training models on gpu's, such as PyTorch or TensorFlow with its more high-level companion Keras.

The downside to this is that one is restricted to using the operations defined by the framework. One could of course define all one's operations in the low-level language that is compiled and linked in. This would however require a lot of knowledge of the internals of the framework used, and would offer none of the advantages of using Python. Because the frameworks do a lot of optimisation in the background, the primitives that are handled are a lot more complicated than what they appear on the surface. Typically the multidimensional arrays that are commonly used in ML are not big chunks of memory with numbers in, but rather represents a node in a computational graph and stores all sort of information about where its data comes from and/or is passed on to. This makes it impossible to employ operations on such "arrays", other than the ones defined in the framework. Also, "array objects" (often referred to as tensors) from one framework doesn't work with transformations from another framework, the tensors belong to different and unrelated classes in different frameworks.

What one is faced with here is sometimes called the two-language problem. Python is relatively easy to program, but it doesn't quite get everything we want done. One would then have to use some other low-level language to ameliorate the situation, in the worst case ending up with none of the benefits and all of the inconvenience – unmaintainable code that still doesn't quite deliver what one wants.

To reiterate, we can run models in Python reasonably fast by using inflexible frameworks, but only if all operations that we might want can be easily expressed from the primitives available.

One should also bear in mind that the when this code is running it's jumping in and out of the Python interpreter code and to and from specially linked routines. The code run in the interpreter is still slow and the dynamically linked libraries form watertight barriers over which no optimisations can be made. In fact, the code run in the Python interpreter can be very slow, often dependent on what exact language construct out of a set of seemingly equivalent ones, is being used.

In paper I I implemented everything in Python, trying to take advantage of the Keras library as much as possible, but also paying a very high penalty for the augmentation code that was written in plain NumPy.

For the rest of my work, resulting in papers II and III, I therefore decided to switch to the programming language Julia.

The Solution

The Julia programming language and its implementation has some unusual properties. It comes with a large runtime environment that can be programmed interactively, much like the Python interpreter. The important difference, however, is that the source code is compiled into highly optimised native machine code on the fly. The compiler is a part of the runtime environment and it is invoked in the background as needed to have all the current code in a compiled state. Usually, the compiled code isn't even saved to permanent storage and the compilation process can be revisited at any time if needed. This also has some downsides. There are often little pauses as code is compiled. In computation-intensive applications this is more than made up for by the faster execution, once the code is compiled.

In this way, even the most time critical parts of an algorithm can be written in ordinary Julia. The resident just-in-time compiler can also utilise different backends, so that code meant to be run on for example a GPU, can be written using the same language as the rest of the code.

5

Discussion

The work put forth here is about trying to infer some quite intangible properties, like the presence of a neuropsychiatric disorder from functional MRI, though that paper specifically addressed if augmentation could help in that situation. Obviously, it would be next to impossible for a human to digest all the data in a functional MRI scan. With computer models and enough training data, such classifications will probably become possible in the future (with higher sensitivity, and specificity, so that it can be used with some confidence in screening for example). This, of course, assumes that the information really is available to find in the first place.

The goals have been several, but more than anything to try to understand what medical imaging information and perhaps how much, that is needed for different machine learning tasks. To what complexity models need to be built, given some fairly precise specifications.

Had I known what I know now, when I started, I still hadn't been able to derail a worldwide pandemic, that featured somewhat inconveniently in my graduate studies. I would probably have abandoned Python for Julia earlier. This really is an important technology that put GPU programming, and all sorts of very powerful programming concepts at the fingertips of people without requiring them to learn a language much harder than Python, and then let them use that for all programming needs, including writing GPU kernels.

In future work on brain age prediction, it would be of interest to, for example, use 2D CNNs pre-trained on ImageNet [53] or RadImageNet [57], instead of training the networks from scratch. This can enable the use of much deeper CNNs, which have the potential to further lower the prediction error. The main problem with ImageNet is that it does not contain medical images, and it would therefore be very interesting to compare CNNs pre-trained on ImageNet and RadImageNet. While the UK biobank dataset is huge from a medical imaging perspective, it is rather small from a deep learning perspective (ImageNet contains several million images). To increase the number of

2D projections, one could perform random augmentations in 3D and save 2D projections after each random transformation. Another interesting idea is to use vision transformers [58] instead of CNNs, as they can use long-range dependencies to further reduce the prediction error. Altogether, most of these proposals need to consider the balance between fast training and the size of the prediction error.

Bibliography

[1] Hans-Ulrich Wittchen, Frank Jacobi, Jürgen Rehm, Anders Gustavsson, Mikael Svensson, Bengt Jönsson, Jes Olesen, Christer Allgulander, Jordi Alonso, Carlo Faravelli, et al. "The size and burden of mental disorders and other disorders of the brain in Europe 2010". In: *European neuropsychopharmacology* 21.9 (2011), pp. 655–679.

[2] Tony Charman. "The prevalence of autism spectrum disorders". In: *European Child amp; Adolescent Psychiatry* 11.6 (2002), pp. 249–256.

[3] F.H. Doyle, J.M. Pennock, J.S. Orr, J.C. Gore, G.M. Bydder, R.E. Steiner, I.R. Young, H. Clow, D.R. Bailes, M. Burl, D.J. Gilderdale, and P.E. Walters. "Imaging of the brain by nuclear magnetic resonance". In: *The Lancet* 318.8237 (1981), pp. 53–57.

[4] Walter J. Huk and Günther Gademann. "Magnetic resonance imaging (MRI): Method and early clinical experiences in diseases of the central nervous system". In: *Neurosurgical Review* 7.4 (1984), pp. 259–280.

[5] Daniel L. Kent. "The Clinical Efficacy of Magnetic Resonance Imaging in Neuroimaging". In: *Annals of Internal Medicine* 120.10 (1994), p. 856.

[6] Stefan Klöppel, Ahmed Abdulkadir, Clifford R Jack Jr, Nikolaos Koutsouleris, Janaina Mourão-Miranda, and Prashanthi Vemuri. "Diagnostic neuroimaging across diseases". In: *Neuroimage* 61.2 (2012), pp. 457–463.

[7] Karla L Miller, Fidel Alfaro-Almagro, Neal K Bangerter, David L Thomas, Essa Yacoub, Junqian Xu, Andreas J Bartsch, Saad Jbabdi, Stamatios N Sotiropoulos, Jesper LR Andersson, et al. "Multimodal population brain imaging in the UK Biobank prospective epidemiological study". In: *Nature neuroscience* 19.11 (2016), pp. 1523–1536.

[8] Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. "The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism". In: *Molecular psychiatry* 19.6 (2014), pp. 659–667.

[9] Anders Eklund, Thomas E Nichols, and Hans Knutsson. "Reply to Brown and Behrmann, Cox, et al., and Kessler et al.: Data and code sharing is the way forward for fMRI". In: *Proceedings of the National Academy of Sciences* 114.17 (2017), E3374–E3375.

[10] Russell A Poldrack, Chris I Baker, Joke Durnez, Krzysztof J Gorgolewski, Paul M Matthews, Marcus R Munafò, Thomas E Nichols, Jean-Baptiste Poline, Edward Vul, and Tal Yarkoni. "Scanning the horizon: towards transparent and reproducible neuroimaging research". In: *Nature reviews neuroscience* 18.2 (2017), pp. 115–126.

[11] Hendrik Szurmant and George W Ordal. "Diversity in chemotaxis mechanisms among the bacteria and archaea". In: *Microbiology and molecular biology reviews* 68.2 (2004), pp. 301–319.

[12] Kenneth H. Nealson, Terry Platt, and J. Woodland Hastings. "Cellular Control of the Synthesis and Activity of the Bacterial Luminescent System". In: *Journal of Bacteriology* 104.1 (1970), pp. 313–322.

[13] A. L. Hodgkin and A. F. Huxley. "A quantitative description of membrane current and its application to conduction and excitation in nerve". In: *The Journal of Physiology* 117.4 (1952), pp. 500–544.

[14] David E. Goldman. "Potential, impedance and rectification in membranes". In: *Journal of General Physiology* 27.1 (1943), pp. 37–60.

[15] A. L. Hodgkin and B. Katz. "The effect of sodium ions on the electrical activity of the giant axon of the squid". In: *The Journal of Physiology* 108.1 (1949), pp. 37–77.

[16] Javier Baladron, Diego Fasoli, Olivier Faugeras, and Jonathan Touboul. "Mean-field description and propagation of chaos in networks of Hodgkin-Huxley and FitzHugh-Nagumo neurons". In: *The Journal of Mathematical Neuroscience* 2.1 (2012).

[17] A. Galves and E. Löcherbach. "Infinite Systems of Interacting Chains with Memory of Variable Length—A Stochastic Model for Biological Neural Nets". In: *Journal of Statistical Physics* 151.5 (2013), pp. 896–921.

[18] Michael G. Paulin and Joseph Cahill-Lane. "Events in Early Nervous System Evolution". In: *Topics in Cognitive Science* 13.1 (2019), pp. 25–44.

[19] Simon Conway Morris and Jean-Bernard Caron. "*Pikaia gracilens* Walcott, a stem-group chordate from the Middle Cambrian of British Columbia". In: *Biological Reviews* 87.2 (2012), pp. 480–512.

[20] Rodrigo T. Figueroa, Danielle Goodvin, Matthew A. Kolmann, Michael I. Coates, Abigail M. Caron, Matt Friedman, and Sam Giles. "Exceptional fossil preservation and evolution of the ray-finned fish brain". In: *Nature* 614.7948 (2023), pp. 486–491.

[21] James H Cole, Rudra PK Poudel, Dimosthenis Tsagkrasoulis, Matthan WA Caan, Claire Steves, Tim D Spector, and Giovanni Montana. "Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker". In: *NeuroImage* 163 (2017), pp. 115–124.

[22] Vishnu M Bashyam, Guray Erus, Jimit Doshi, Mohamad Habes, Ilya M Nasrallah, Monica Truelove-Hill, Dhivya Srinivasan, Liz Mamourian, Raymond Pomponio, Yong Fan, Lenore J Launer, Colin L Masters, Paul Maruff, Chuanjun Zhuo, Henry Völzke, Sterling C Johnson, Jurgen Fripp, Nikolaos Koutsouleris, Theodore D Satterthwaite, Daniel Wolf, Raquel E Gur, Ruben C Gur, John Morris, Marilyn S Albert, Hans J Grabe, Susan Resnick, R Nick Bryan, David A Wolk, Haochang Shou, and Christos Davatzikos. "MRI signatures of brain age and disease over the lifespan based on a deep brain network and 14468 individuals worldwide". In: *Brain* 143.7 (2020), pp. 2312–2324.

[23] Catherine Lord, Edwin H Cook, Bennett L Leventhal, and David G Amaral. "Autism spectrum disorders". In: *Neuron* 28.2 (2000), pp. 355–363.

[24] Sabine M Klauck. "Genetics of autism spectrum disorder". In: *European Journal of Human Genetics* 14.6 (2006), pp. 714–720.

[25] Rebecca Muhle, Stephanie V. Trentacoste, and Isabelle Rapin. "The Genetics of Autism". In: *Pediatrics* 113.5 (2004), e472–e486.

[26] Sven Sandin, Paul Lichtenstein, Ralf Kuja-Halkola, Christina Hultman, Henrik Larsson, and Abraham Reichenberg. "The Heritability of Autism Spectrum Disorder". In: *JAMA* 318.12 (2017), p. 1182.

[27] Jane Libbey, Thayne Sweeten, William McMahon, and Robert Fujinami. "Autistic disorder and viral infections". In: *Journal of NeuroVirology* 11.1 (2005), pp. 1–10.

[28] D Kinney, K Munir, D Crowley, and A Miller. "Prenatal stress and risk for autism". In: *Neuroscience amp; Biobehavioral Reviews* 32.8 (2008), pp. 1519–1532.

[29] Hannah Gardener, Donna Spiegelman, and Stephen L. Buka. "Perinatal and Neonatal Risk Factors for Autism: A Comprehensive Meta-analysis". In: *Pediatrics* 128.2 (2011), pp. 344–355.

[30] Tara L. Arndt, Christopher J. Stodgell, and Patricia M. Rodier. "The teratology of autism". In: *International Journal of Developmental Neuroscience* 23.2–3 (2005), pp. 189–199.

[31] Werner Poewe, Klaus Seppi, Caroline M Tanner, Glenda M Halliday, Patrik Brundin, Jens Volkmann, Anette-Eleonore Schrag, and Anthony E Lang. “Parkinson disease”. In: *Nature reviews Disease primers* 3.1 (2017), pp. 1–21.

[32] David S Knopman, Helene Amieva, Ronald C Petersen, Gäel Chételat, David M Holtzman, Bradley T Hyman, Ralph A Nixon, and David T Jones. “Alzheimer disease”. In: *Nature reviews Disease primers* 7.1 (2021), p. 33.

[33] Tao Ai, John N. Morelli, Xuemei Hu, Dapeng Hao, Frank L. Goerner, Bryan Ager, and Val M. Runge. “A Historical Overview of Magnetic Resonance Imaging, Focusing on Technological Innovations”. In: *Investigative Radiology* 47.12 (2012), pp. 725–741.

[34] Peter A Bandettini. “Twenty years of functional MRI: the science and the stories”. In: *Neuroimage* 62.2 (2012), pp. 575–588.

[35] Tan Khoa Nguyen, Henrik Ohlsson, Anders Eklund, Frida Hernell, Patric Ljung, Camilla Forsell, Mats Andersson, Hans Knutsson, and Anders Ynnerman. “Concurrent volume visualization of real-time fMRI”. In: *IEEE/EG International Symposium on Volume Graphics*. Eurographics-European Association for Computer Graphics. 2010, pp. 53–60.

[36] Anders Eklund, Ola Friman, Mats Andersson, and Hans Knutsson. “A GPU accelerated interactive interface for exploratory functional connectivity analysis of fMRI data”. In: *IEEE international conference on image processing*. 2011, pp. 1589–1592.

[37] Susumu Mori and Peter B Barker. “Diffusion magnetic resonance imaging: its principle and applications”. In: *The Anatomical Record: An Official Publication of the American Association of Anatomists* 257.3 (1999), pp. 102–109.

[38] Joel Hedlund, Anders Eklund, and Claes Lundström. “Key insights in the AIDA community policy on sharing of clinical imaging data for research in Sweden”. In: *Scientific Data* 7.1 (2020), p. 331.

[39] Dinggang Shen, Guorong Wu, and Heung-Il Suk. “Deep learning in medical image analysis”. In: *Annual review of biomedical engineering* 19 (2017), pp. 221–248.

[40] Arpit Kumar Sharma, Amita Nandal, Arvind Dhaka, and Rahul Dixit. “Medical image classification techniques and analysis using deep learning networks: a review”. In: *Health informatics: a computational perspective in healthcare* (2021), pp. 233–258.

[41] Xiangbin Liu, Liping Song, Shuai Liu, and Yudong Zhang. “A review of deep-learning-based medical image segmentation methods”. In: *Sustainability* 13.3 (2021), p. 1224.

[42] Donald O. Hebb. *The organization of behavior: A neuropsychological theory*. New York: Wiley, 1949. ISBN: 0-8058-4300-0.

[43] Peter M. Milner. “The Mind and Donald O. Hebb”. In: *Scientific American* 268.1 (1993), pp. 124–129.

[44] A. L. Samuel. “Some Studies in Machine Learning Using the Game of Checkers”. In: *IBM Journal of Research and Development* 3.3 (1959), pp. 210–229.

[45] A. L. Samuel. “Some Studies in Machine Learning Using the Game of Checkers. II—Recent Progress”. In: *IBM Journal of Research and Development* 11.6 (1967), pp. 601–617.

[46] Corinna Cortes and Vladimir Vapnik. “Support-vector networks”. In: *Machine Learning* 20.3 (1995), pp. 273–297.

[47] J. R. Quinlan. “Induction of decision trees”. In: *Machine Learning* 1.1 (1986), pp. 81–106.

[48] Y. Zhang, M. Brady, and S. Smith. “Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm”. In: *IEEE Transactions on Medical Imaging* 20.1 (2001), pp. 45–57.

[49] Stephen M. Smith, Mark Jenkinson, Mark W. Woolrich, Christian F. Beckmann, Timothy E.J. Behrens, Heidi Johansen-Berg, Peter R. Bannister, Marilena De Luca, Ivana Drobnjak, David E. Flitney, Rami K. Niazy, James Saunders, John Vickers, Yongyue Zhang, Nicola De Stefano, J. Michael Brady, and Paul M. Matthews. “Advances in functional and structural MR image analysis and implementation as FSL”. In: *NeuroImage* 23 (2004), S208–S219.

[50] Mark W. Woolrich, Saad Jbabdi, Brian Patenaude, Michael Chappell, Salima Makni, Timothy Behrens, Christian Beckmann, Mark Jenkinson, and Stephen M. Smith. “Bayesian analysis of neuroimaging data in FSL”. In: *NeuroImage* 45.1 (2009), S173–S186.

[51] Mark Jenkinson, Christian F. Beckmann, Timothy E.J. Behrens, Mark W. Woolrich, and Stephen M. Smith. “FSL”. In: *NeuroImage* 62.2 (2012), pp. 782–790.

[52] Håvard S. Ugulen, Daniel Koestner, Håkon Sandven, Børge Hamre, Arne S. Kristoffersen, and Camilla Saetre. “Neural network approach for correction of multiple scattering errors in the LISST-VSF instrument”. In: *Optics Express* 31.20 (2023), p. 32737.

[53] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. 2009, pp. 248–255.

[54] Marco Domenico Cirillo, David Abramian, and Anders Eklund. "What is the best data augmentation for 3D brain tumor segmentation?" In: *IEEE International Conference on Image Processing (ICIP)*. 2021, pp. 36–40.

[55] Taro Langner, Johan Wikström, Tomas Bjerner, Håkan Ahlström, and Joel Kullberg. "Identifying morphological indicators of aging with neural networks on large-scale whole-body MRI". In: *IEEE transactions on medical imaging* 39.5 (2019), pp. 1430–1437.

[56] Kaifeng Gao, Gang Mei, Francesco Piccialli, Salvatore Cuomo, Jingzhi Tu, and Zenan Huo. "Julia language in machine learning: Algorithms, applications, and open issues". In: *Computer Science Review* 37 (2020), p. 100254.

[57] Xueyan Mei, Zelong Liu, Philip M Robson, Brett Marinelli, Mingqian Huang, Amish Doshi, Adam Jacobi, Chendi Cao, Katherine E Link, Thomas Yang, et al. "RadImageNet: an open radiologic deep learning research dataset for effective transfer learning". In: *Radiology: Artificial Intelligence* 4.5 (2022), e210315.

[58] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *ICLR* (2020).

Paper I

Evaluation of augmentation methods in classifying autism spectrum disorders from fMRI data with 3D convolutional neural networks

Johan Jönemo
David Abramian
Anders Eklund

Diagnostics, **2023**, *13*(17), 2773; doi: 10.3390/diagnostics13172773

[1] Available at https://creativecommons.org/licenses/by/4.0/

diagnostics

MDPI

Article

Evaluation of Augmentation Methods in Classifying Autism Spectrum Disorders from fMRI Data with 3D Convolutional Neural Networks

Johan Jönemo [1,2], David Abramian [1,2] and Anders Eklund [1,2,3,*]

[1] Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, 581 83 Linköping, Sweden
[2] Center for Medical Image Science and Visualization (CMIV), Linköping University, 581 83 Linköping, Sweden
[3] Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, 581 83 Linköping, Sweden
[*] Correspondence: anders.eklund@liu.se

Abstract: Classifying subjects as healthy or diseased using neuroimaging data has gained a lot of attention during the last 10 years, and recently, different deep learning approaches have been used. Despite this fact, there has not been any investigation regarding how 3D augmentation can help to create larger datasets, required to train deep networks with millions of parameters. In this study, deep learning was applied to derivatives from resting state functional MRI data, to investigate how different 3D augmentation techniques affect the test accuracy. Specifically, resting state derivatives from 1112 subjects in ABIDE (Autism Brain Imaging Data Exchange) preprocessed were used to train a 3D convolutional neural network (CNN) to classify each subject according to presence or absence of autism spectrum disorder. The results show that augmentation only provide minor improvements to the test accuracy.

Keywords: functional MRI; resting state; deep learning; augmentation; autism

check for updates

Citation: Jönemo, J.; Abramian, D.; Eklund, A. Evaluation of Augmentation Methods in Classifying Autism Spectrum Disorders from fMRI Data with 3D Convolutional Neural Networks. *Diagnostics* **2023**, *13*, 2773. https://doi.org/10.3390/diagnostics13172773

Academic Editor: Daniele Giansanti

Received: 9 August 2023
Revised: 24 August 2023
Accepted: 25 August 2023
Published: 27 August 2023

1. Introduction

Ever since the emergence of magnetic resonance imaging (MRI) in the 1980s, the absence of ionizing radiation and the flexibility of the acquisition procedure have made this an increasingly important imaging modality in the clinical sciences. The lack of contrast between different tissues in the brain and the interference of the mineralized tissue around it when using X-ray techniques make MRI especially useful in neuroimaging.

While a wide variety of neurological conditions can be diagnosed with MRI, psychiatric anomalies have proven illusive to detect. Presumably, this is because these affect many systems distributed throughout the brain and their manifestations are likely subtle as well as time variant. Furthermore, psychiatric anomalies can vary a lot between subjects. Functional MRI (fMRI) is a technique that seems particularly suited to capture this information, as it generates rich 4D data which can be used for studying brain activity as well as brain connectivity. In this work, it is investigated if deep-learning-based diagnosis of autism from resting state fMRI data can be further improved using 3D augmentation.

1.1. Resting State fMRI

Resting state fMRI has since 1995 been used to study brain connectivity [1,2]. A major advantage compared to task fMRI is that subjects can simply rest during the whole experiment, which normally takes 5–10 min (resulting in some 150–600 brain volumes, or put differently some 50,000 time series), instead of performing different tasks such as finger tapping or mental calculations. This makes it possible to include subjects which for some reason cannot perform certain tasks. A simple measure of the connectivity between two locations in the brain, called functional connectivity, is the correlation between the two corresponding time series, but several more advanced methods also exist. To limit the size of the 2D correlation matrix, the correlations are normally calculated between the mean time series of some 100–200 brain parcels (instead of some 50,000 voxels). The brain can be divided according to different (resting state) networks, such as the default mode network and the auditory network, and different diseases often affect specific networks.

1.2. Autism

Autism spectrum disorder (ASD) is a disorder characterized by certain features in social communication, and restricted, repetitive, or unusual sensory–motor behaviours [3]. The prevalence of ASD is 1–5% in developed countries [4]. The subject of autism has been studied extensively in recent years, and technology has already contributed to the development of treatments for autism, in terms of rehabilitation and communication.

Due to the lack of reliable biomarkers, the diagnosis is usually based on behaviour, which is very time consuming. Recent work has demonstrated that motor abnormalities can be very informative for detection of ASD [5,6], and that machine learning can be used to shorten the behavioral diagnosis [7]. As ASD results from early altered brain development and neural reorganisation [8,9], it should be possible to derive objective biomarkers from neuroimaging data to aid professionals (paediatricians, psychiatrists, or psychologists) in diagnosising ASD. Here, machine learning can be used to learn informative traits from the high-dimensional fMRI data.

1.3. Machine Learning for Diagnosis of ASD

Several large collaborative efforts have been made to collect and share neuroimaging data of healthy controls as well as diseased [10,11]. ABIDE (Autism Brain Imaging Data Exchange) [12] is one such effort that make available data for 539 subjects diagnosed with ASD as well as 573 typical controls. The ABIDE data originate from 17 sites, and the subjects were aged 7–64 years (median 14.7 years across groups). Using machine learning in an endeavour to classify (resting state) fMRI data according to the presence or absence of ASD has become increasingly popular recently. This classification can be performed in several ways, either using estimated functional connectivity network matrices (2D) or using derivatives (3D volumes), such as weighted and binarized degree centrality, as different approaches to compress the 4D fMRI data. In this work, 3D volumes are used, as it is not obvious how to augment network matrices.

The ASD classification problem seems hard in that accuracies seldom rise to more than 70% when the model classifies unseen data [13–18]. While 1112 subjects is a very large fMRI dataset, it is still small from a deep learning perspective (for example, the popular ImageNet database [19] contains several million images). To further increase the size of the training dataset, and to make convolutional neural networks (CNNs) robust to transformations such as rotation, data augmentation is often used [20,21]. In previous work. it was demonstrated that 3D augmentation for brain tumor segmentation significantly improves the segmentation accuracy [22]. In this work, the purpose is instead to see if 3D augmentation can help train a better ASD classifier, as well as what kind of augmentation techniques work the best.

1.4. Related Work

Several other researchers have used the same ABIDE dataset to train deep learning models for classification [16–18,23,24], but do not mention anything about augmentation. In a recent review on deep learning for autism by Khodatars et al. [25], only advanced augmentation techniques, such as generative adversarial methods (GANs), are briefly mentioned, but training a GAN requires a very large dataset to start from and there is very little work published on 3D GANs. Some researchers have employed resampling techniques wherein shorter time series have been cropped out of longer ones [13,14], typically for the double purpose of getting an augmented data set while also eliminating the extra complication of variable length sequences. Ji et al. [26] instead applied augmentation to the estimated network matrices. In our study, by contrast, different preprocessing pipelines are used to extract all relevant information from the time dimension, and manipulate data only in the spatial domain.

2. Materials and Methods

2.1. Data

Preprocessing of 4D resting state fMRI data is a complex process involving many different steps, and there is no consensus regarding what the optimal pipeline or toolbox is [27]. Head motion is a major problem in resting state fMRI, as it can, for example, result in erroneous group differences if two cohorts differ in the mean amount of head motion [28,29]. All processing pipelines therefore perform head motion correction, and use additional steps to further suppress motion related signal. ABIDE preprocessed [30] (http://preprocessed-connectomes-project.org/abide/, accessed on 10 February 2023) shares preprocessed ABIDE [12] data from structural MRI and resting state fMRI in various forms. As all the preprocessing has been completed, the focus in this work is on the machine-learning-based diagnosis, and other researchers can use the same preprocessed data to reproduce the presented findings. Resting state derivatives (3D volumes where the time dimension has been collapsed into different forms of statistics) resulting from two pipelines were downloaded from ABIDE preprocessed, for 1112 subjects.

One pipeline was the connectome computation system (CCS) [31], which performs slice timing correction, motion realignment, and global intensity normalisation. The data were cleaned from confounders by performing regression with the estimated head movement parameters, the time-dependent global mean intensity, as well as regressors for linear and quadratic drift. Each time series was also band pass filtered (0.01–0.1 Hz). This preprocessing corresponds to the strategy called global_filt. Each subject was, furthermore, registered to the MNI152 brain template using boundary based rigid body registration [32] for functional to anatomical registration, and FLIRT and FNIRT for anatomical to template registration [33].

Another such pipeline was "data processing assistant for resting-state fMRI" (DPARSF) [34]. It also performs slice timing correction and motion reallignment, but does not perform any intensity normalisation. The same confounders are corrected for and the same band pass filtering is performed, whereupon functional to anatomical registration was performed with ordinary rigid body methods and anatomical to MNI152 brain template registration completed using DARTEL [35].

After preliminary testing of the 10 available derivatives available in ABIDE preprocessed (amplitude of low frequency fluctuations (ALFF), weighted and binarized degree centrality, dual regression, weighted and binarized eigenvector centrality, fractional ALFF, local functional connectivity density (LFCD), regional homogeneity (REHO), voxel-mirrored homotopic connectivity (VMHC)), the REHO derivative was chosen for comparing different augmentation strategies. Regional homogeneity is a measure of correlation between a voxel's time series and those of its neighbours [36], based on the non-parametric rank correlation statistic known as Kendall's Coefficient of Concordance (KCC) [37]. Each derivative volume from the resting state fMRI data has a size of $61 \times 73 \times 61$ voxels (each $3 \times 3 \times 3$ mm^3), which is fed into the 3D CNN described below. See Figure 1 for a preprocessed fMRI volume and the REHO derivative from the CCS pipeline, downloaded from ABIDE (https://s3.amazonaws.com/fcp-indi/data/Projects/ABIDE_Initiative/Outputs/ccs/filt_global/func_preproc/OHSU_0050147_func_preproc.nii.gz, accessed on 1 August 2023; https://s3.amazonaws.com/fcp-indi/data/Projects/ABIDE_Initiative/Outputs/ccs/filt_global/reho/OHSU_0050147_reho.nii.gz, accessed on 1 August 2023). The 539 subjects with ASD and the 573 controls were split 70/15/15 into training, validation, and test sets.

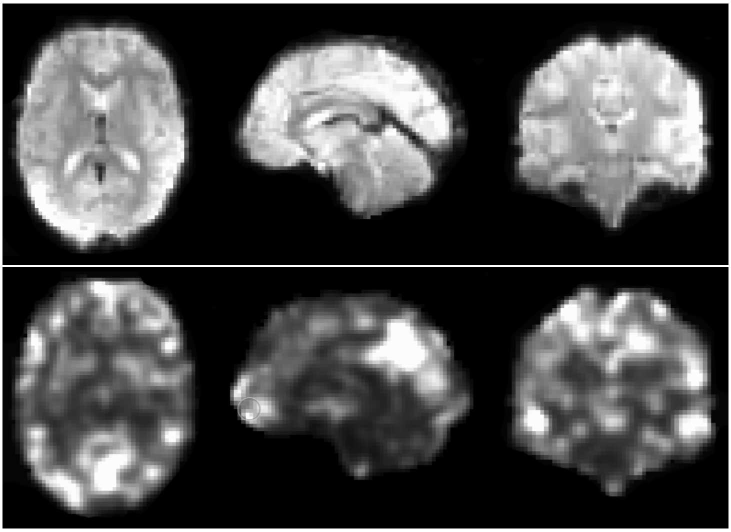


Figure 1. (**Top**): an fMRI volume obtained after preprocessing with the CCS pipeline. (**Bottom**): the REHO derivative obtained from the preprocessed 4D fMRI dataset, used by the 3D CNN to classify each subject as control or ASD. Different types of 3D augmentation were applied to each REHO volume, in an attempt to improve the test accuracy. Several other derivatives are available in ABIDE preprocessed, but were not used in this study due to time-consuming training.

2.2. Deep Learning

CNNs are often used for deep-learning-based classification and segmentation of image data, as learning a number of small filters is much more efficient compared to training a dense network (which models the relationship between all pixels in an image, instead of only looking at local correlations). While 2D CNNs are much more common, they are easily extended to 3D as convolution can be performed in any number of dimensions. Unfortunately, existing deep learning frameworks do not support 4D convolutions, which would be required to directly classify 4D fMRI data. The 3D CNN used in this work was implemented using Keras and consists of three convolutional layers (with ReLU activation), max-pooling layers, a dense layer with 16 nodes, and a final one-node layer with sigmoid activation. The first and second convolutional layers contain 8 filters each (size $3 \times 3 \times 3$), and the last convolutional layer uses 16 filters. The total number of trainable parameters in the 3D CNN is approximately 450 k. The CNN was trained with the Adam optimizer with a learning rate of 10^{-5} and a batch size of 16. To prevent overfitting, early stopping was used with a patience of 50 epochs. The training was run until validation accuracy did not improve, and the model was then restored to the state when the last improvement was seen. As an alternative, the models were also trained for 150 epochs with no conditional stopping. To obtain more robust estimates of the test accuracy, 10-fold cross validation was used and the mean test accuracy was calculated.

2.3. Augmentation

There are many types of augmentation that can be useful in 3D. Rotation, flipping, and scaling (zooming in or out) are common for training 2D CNNs, and can also easily be applied in 3D. Elastic (non-linear) deformations are common when training segmentation networks, but perhaps not as common for classification. Brightness augmentation can for example help if the data have been collected at several different MR scanners, as they normally generate data with different brightness [22].

While 2D augmentation functions are included in many deep learning frameworks such as Keras and Pytorch, the support for 3D augmentation is normally lacking. As mentioned by Chlap et al. [21], many researchers use 2D augmentation even if the data are 3D. The 3D augmentation used here is adapted from that of Cirillo et al. [22] and is written in Python/NumPy [38], without facilities for running on a GPU. The 3D augmentation techniques tested in this study are:

- *Flipping*: flipping of the x-axis or not.
- *Rotation*: rotation applied to each axis with angles randomly chosen from a uniform distribution with range between -7.5 and 7.5 degrees, -15 and 15 degrees, -30 and 30 degrees, or -45 and 45 degrees.
- *Scale*: scaling applied to each axis by a factor randomly chosen from a uniform distribution with range ±10% or ±20%.
- *Brightness*: power-law γ intensity transformation with its parameters gain (g) and γ chosen randomly between 0.8 and 1.2 from a uniform distribution. The intensity (I) is randomly changed according to the formula: $I_{new} = g \cdot I^{\gamma}$.
- *Elastic deformation*: elastic (non-linear) deformation with square deformation grid with displacements sampled from from a normal distribution with standard deviation $\sigma = 2, 4, 6$, or 8 voxels [39], where the smoothing is done by a spline filter with order 3 in each dimension.

To investigate the effect of combining different types of augmentation, the CNNs were also trained with the two best-performing augmentation approaches according to the CCS pipeline.

The average training time for a single fold were between five minutes and 2.5 h—depending on the type of on-the-fly augmentation employed, the combination of elastic deformation, and an affine transformation being the slowest–using one Nvidia Tesla V100 graphics card for the early stopping models. For the training with a fixed number of epochs, the average single fold training time was at least 10 min but otherwise in the previously mentioned span. In the longer training runs, it is unlikely that the computation speed was bounded by the speed of the graphics card, as the on-the-fly augmentations were performed on the CPU and could be further optimized. In total, some 600 3D CNNs were trained in order to compare all settings.

3. Results

The results from all the different augmentation techniques, as well as baseline results obtained without augmentation, are presented in Figures 2 and 3 (CCS pipeline) and Figures 4 and 5 (DPARSF pipeline). As the dataset is balanced (similar number of ASD and control subjects), only classification accuracy is reported (instead of more advanced metrics, such as area under the curve and Matthew's correlation coefficient). In general, the 3D augmentation does not have a large effect on the test accuracy. For early stopping with the CCS pipeline, random scaling seems to be the best single augmentation approach, but the mean improvement over 10 cross-validation folds is only about 0.5 percentage units. Small elastic deformations also have a small positive effect, while large deformations give worse results.

With the DPARSF pipeline brightness changes appear to be the best augmentation with an increase of 1.9 percentage units, but with high variance over folds, the improvement is negligible. For a fixed number of training epochs, elastic deformations and rotations or combinations thereof seem to work best, with the best improvement of accuracy being 2.2 percentage units in the CCS pipeline and 2.9 percentage units in the DPARSF pipeline. No statistical test was performed to test if this improvement is significant.

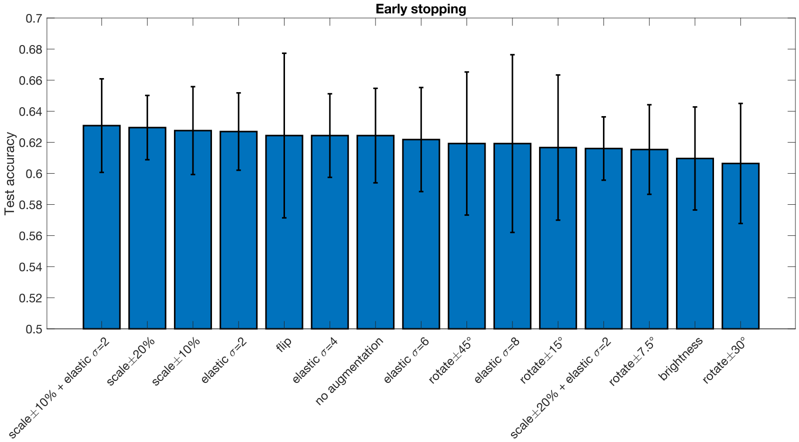


Figure 2. Test accuracy for classifying subjects as healthy or diseased for the ABIDE dataset processed with the CCS pipeline, for different data augmentation approaches. The error bar represents the standard deviation over the 10 cross-validation folds. Note that half of the augmentation approaches result in a test accuracy that is lower compared to the baseline model trained without augmentation, but overall, the differences are small. These results were obtained when using early stopping. Compared to no augmentation, the best augmentation approach increases the test accuracy by 0.6 percentage units.

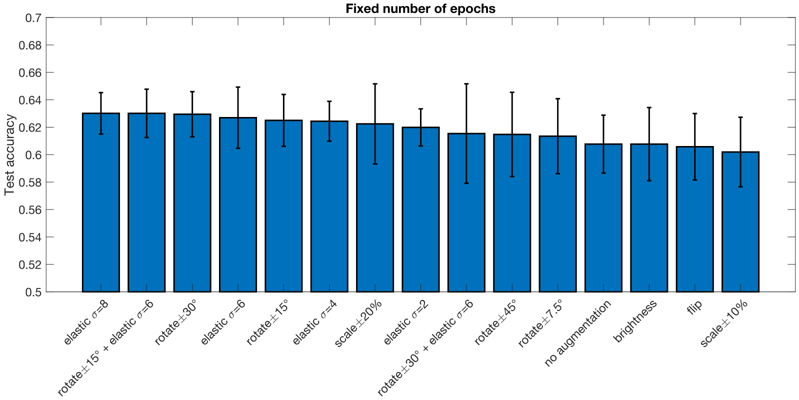


Figure 3. Test accuracy for classifying subjects as healthy or diseased for the ABIDE dataset processed with the CCS pipeline, for different data augmentation approaches. The error bar represents the standard deviation over the 10 cross-validation folds. These results were obtained when using a fixed number of epochs for each training. Compared to no augmentation, the best augmentation approach increases the test accuracy by 2.2 percentage units.

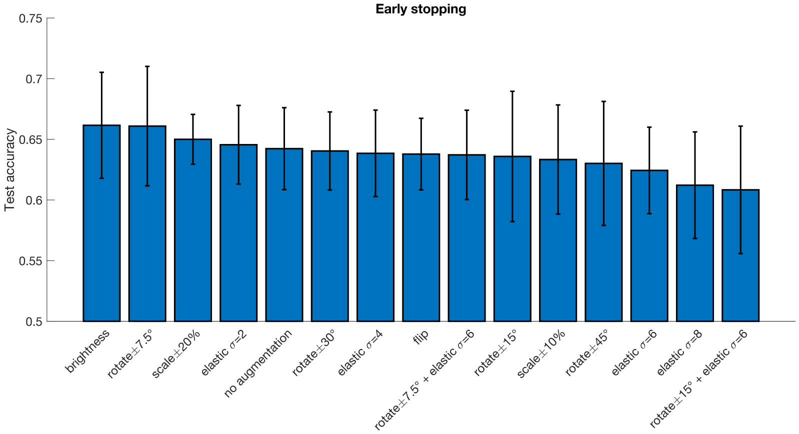


Figure 4. Test accuracy for classifying subjects as healthy or diseased for the ABIDE dataset processed with the DPARSF pipeline, for different data augmentation approaches. The error bar represents the standard deviation over the 10 cross-validation folds. Note that half of the augmentation approaches result in a test accuracy that is lower compared to the baseline model trained without augmentation, but overall, the differences are small. These results were obtained when using early stopping. Compared to no augmentation, the best augmentation approach increases the test accuracy by 1.9 percentage units.

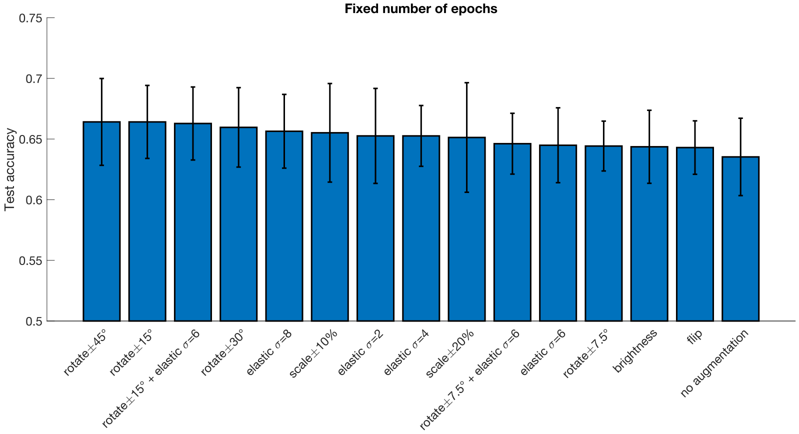


Figure 5. Test accuracy for classifying subjects as healthy or diseased for the ABIDE dataset processed with the DPARSF pipeline, for different data augmentation approaches. The error bar represents the standard deviation over the 10 cross-validation folds. These results were obtained when using a fixed number of epochs for each training. Compared to no augmentation, the best augmentation approach increases the test accuracy by 2.9 percentage units.

4. Discussion

Compared to previous work on 3D augmentation for brain tumor segmentation [22], where several 3D augmentation techniques were shown to significantly improve the segmentation accuracy on the test set, only minor improvements of the test accuracy were found in this study (even though the training accuracy is well above 90%, indicating overfitting). Volume classification is in general a problem which requires more training data compared to volume segmentation, as each volume only represents a single training example, which may partly explain the results.

In this study, brightness augmentation only helps for the DPARSF pipeline with early stopping, while it provided a major improvement for brain tumor segmentation for MR images collected at some 20 different sites [22]. A possible explanation is that the data in this study are not raw MR images, since many preprocessing steps have been used to normalize the intensities to a certain range, and to calculate different derivatives. On the contrary, as the ranges of values in the derivative volumes are not, in general, arbitrary in the same way, brightness augmentation can impair the performance. In DPARSF, no intensity normalization is performed, which may explain why the brightness augmentation results are different compared to the CCS pipeline.

Since all the subjects have been registered to MNI space, it was hypothesized that the results may be different if random transformations are applied to the test volumes, but test time augmentation did not change the findings (results not shown). The presented results are for a single preprocessing strategy (global signal regression and bandpass filtering), and a single derivative, and the preprocessing choice can at least in theory affect how much the augmentation helps.

The focus here has been on classifying ASD and controls, with a binary classifier. ASD criteria are based on DSM-5 criteria, and there are currently three levels of severity. It is possible that using 3D augmentation when training a classifier to distinguish the three severity levels could lead to different results.

The conclusion is that 3D augmentation only provides minor improvements in accuracy (0.6–2.9 percentage units) when training 3D CNNs for classification of ASD versus controls, but the results may be different for an easier task where the baseline test accuracy is for example 80%. The results may also differ for other derivatives in ABIDE preprocessed, and when using several derivatives at the same time using a multi-channel 3D CNN. However, to perform the trainings for many combinations of preprocessing, and for different derivatives, would be very time consuming.

Author Contributions: J.J. and D.A. implemented the 3D CNN and performed the experiments. J.J. and A.E. wrote the paper. A.E. supervised the work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Swedish research council grant 2017-04889, and by the ITEA/VINNOVA funded project Automation, Surgery Support and Intuitive 3D visualization to optimize workflow in IGT SysTems (ASSIST) (grant 2021-01954), and by the Åke Wiberg foundation (grant M22-0088).

Institutional Review Board Statement: This research study was conducted retrospectively using anonymized human subject data made available in open access by ABIDE preprocessed. Ethical approval was not required as confirmed by the ethics committee of Linköping.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study; see ABIDE for details.

Data Availability Statement: The datasets analyzed for this study can be found at http://preprocessed-connectomes-project.org/abide/, accessed on 10 February 2023.

Acknowledgments: Primary support for the work by Adriana Di Martino was provided by the (NIMH K23MH087770) and the Leon Levy Foundation. Primary support for the work by Michael P. Milham and the INDI team was provided by gifts from Joseph P. Healy and the Stavros Niarchos Foundation to the Child Mind Institute, as well as by an NIMH award to MPM (NIMH R03MH096321).

Conflicts of Interest: A.E. has previously received hardware from Nvidia. Otherwise, the authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Biswal, B.; Zerrin Yetkin, F.; Haughton, V.M.; Hyde, J.S. Functional connectivity in the motor cortex of resting human brain using echo-planar MRI. *Magn. Reson. Med.* **1995**, *34*, 537–541. [CrossRef] [PubMed]
2. Van Den Heuvel, M.P.; Pol, H.E.H. Exploring the brain network: A review on resting-state fMRI functional connectivity. *Eur. Neuropsychopharmacol.* **2010**, *20*, 519–534. [CrossRef] [PubMed]
3. Lord, C.; Elsabbagh, M.; Baird, G.; Veenstra-Vanderweele, J. Autism spectrum disorder. *Lancet* **2018**, *392*, 508–520. [CrossRef] [PubMed]
4. Lyall, K.; Croen, L.; Daniels, J.; Fallin, M.D.; Ladd-Acosta, C.; Lee, B.K.; Park, B.Y.; Snyder, N.W.; Schendel, D.; Volk, H.; et al. The changing epidemiology of autism spectrum disorders. *Annu. Rev. Public Health* **2017**, *38*, 81–102. [CrossRef] [PubMed]
5. Simeoli, R.; Milano, N.; Rega, A.; Marocco, D. Using technology to identify children with autism through motor abnormalities. *Front. Psychol.* **2021**, *12*, 635696. [CrossRef] [PubMed]
6. Milano, N.; Simeoli, R.; Rega, A.; Marocco, D. A deep learning latent variable model to identify children with autism through motor abnormalities. *Front. Psychol.* **2023**, *14*, 1194760. [CrossRef] [PubMed]
7. Wall, D.P.; Dally, R.; Luyster, R.; Jung, J.Y.; DeLuca, T.F. Use of artificial intelligence to shorten the behavioral diagnosis of autism. *PLoS ONE* **2012**, *7*, e43855. [CrossRef] [PubMed]
8. Bauman, M.L.; Kemper, T.L. Neuroanatomic observations of the brain in autism: A review and future directions. *Int. J. Dev. Neurosci.* **2005**, *23*, 183–187. [CrossRef]
9. O'Reilly, C.; Lewis, J.D.; Elsabbagh, M. Is functional brain connectivity atypical in autism? A systematic review of EEG and MEG studies. *PLoS ONE* **2017**, *12*, e0175870. [CrossRef]
10. Mennes, M.; Biswal, B.B.; Castellanos, F.X.; Milham, M.P. Making data sharing work: The FCP/INDI experience. *Neuroimage* **2013**, *82*, 683–691. [CrossRef]
11. Poldrack, R.A.; Gorgolewski, K.J. Making big data open: Data sharing in neuroimaging. *Nat. Neurosci.* **2014**, *17*, 1510–1517. [CrossRef] [PubMed]
12. Di Martino, A.; Yan, C.G.; Li, Q.; Denio, E.; Castellanos, F.X.; Alaerts, K.; Anderson, J.S.; Assaf, M.; Bookheimer, S.Y.; Dapretto, M.; et al. The autism brain imaging data exchange: Towards a large-scale evaluation of the intrinsic brain architecture in autism. *Mol. Psychiatry* **2014**, *19*, 659. [CrossRef] [PubMed]
13. Dvornek, N.C.; Ventola, P.; Pelphrey, K.A.; Duncan, J.S. Identifying autism from resting-state fMRI using long short-term memory networks. In *Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, 10 September 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 362–370.
14. Huang, Z.A.; Zhu, Z.; Yau, C.H.; Tan, K.C. Identifying autism spectrum disorder from resting-state fMRI using deep belief network. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 2847–2861. [CrossRef] [PubMed]
15. Arbabshirani, M.R.; Plis, S.; Sui, J.; Calhoun, V.D. Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls. *Neuroimage* **2017**, *145*, 137–165. [CrossRef] [PubMed]
16. Heinsfeld, A.S.; Franco, A.R.; Craddock, R.C.; Buchweitz, A.; Meneguzzi, F. Identification of autism spectrum disorder using deep learning and the ABIDE dataset. *Neuroimage Clin.* **2018**, *17*, 16–23. [CrossRef] [PubMed]
17. Yang, X.; Schrader, P.T.; Zhang, N. A deep neural network study of the ABIDE repository on autism spectrum classification. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 1–6. [CrossRef]
18. Thomas, R.M.; Gallo, S.; Cerliani, L.; Zhutovsky, P.; El-Gazzar, A.; van Wingen, G. Classifying autism spectrum disorder using the temporal statistics of resting-state functional MRI data with 3D convolutional neural networks. *Front. Psychiatry* **2020**, *11*, 440. [CrossRef]
19. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 200 9; pp. 248–255.
20. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
21. Chlap, P.; Min, H.; Vandenberg, N.; Dowling, J.; Holloway, L.; Haworth, A. A review of medical image data augmentation techniques for deep learning applications. *J. Med. Imaging Radiat. Oncol.* **2021**, *65*, 545–563. [CrossRef]
22. Cirillo, M.D.; Abramian, D.; Eklund, A. What is the best data augmentation approach for brain tumor segmentation using 3D U-Net? In Proceedings of the IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021.
23. Guo, X.; Dominick, K.C.; Minai, A.A.; Li, H.; Erickson, C.A.; Lu, L.J. Diagnosing autism spectrum disorder from brain resting-state functional connectivity patterns using a deep neural network with a novel feature selection method. *Front. Neurosci.* **2017**, *11*, 460. [CrossRef]
24. El-Gazzar, A.; Quaak, M.; Cerliani, L.; Bloem, P.; van Wingen, G.; Mani Thomas, R. A hybrid 3DCNN and 3DC-LSTM based model for 4D spatio-temporal fMRI data: An ABIDE autism classification study. In *OR 2.0 Context-Aware Operating Theaters and Machine Learning in Clinical Neuroimaging: Second International Workshop, OR 2.0 2019, and Second International Workshop, MLCN 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, 13–17 October 2019*; Springer: Cham, Switzerland, 2019; pp. 95–102.
25. Khodatars, M.; Shoeibi, A.; Sadeghi, D.; Ghaasemi, N.; Jafari, M.; Moridian, P.; Khadem, A.; Alizadehsani, R.; Zare, A.; Kong, Y.; et al. Deep learning for neuroimaging-based diagnosis and rehabilitation of autism spectrum disorder: A review. *Comput. Biol. Med.* **2021**, *139*, 104949. [CrossRef]
26. Ji, J.; Wang, Z.; Zhang, X.; Li, J. Sparse data augmentation based on encoderforest for brain network classification. *Appl. Intell.* **2022**, *52*, 4317–4329. [CrossRef]

27. Waheed, S.H.; Mirbagheri, S.; Agarwal, S.; Kamali, A.; Yahyavi-Firouz-Abadi, N.; Chaudhry, A.; DiGianvittorio, M.; Gujar, S.K.; Pillai, J.J.; Sair, H.I. Reporting of resting-state functional magnetic resonance imaging preprocessing methodologies. *Brain Connect.* **2016**, *6*, 663–668. [CrossRef] [PubMed]
28. Power, J.D.; Barnes, K.A.; Snyder, A.Z.; Schlaggar, B.L.; Petersen, S.E. Spurious but systematic correlations in functional connectivity MRI networks arise from subject motion. *Neuroimage* **2012**, *59*, 2142–2154. [CrossRef]
29. Eklund, A.; Nichols, T.E.; Afyouni, S.; Craddock, C. How does group differences in motion scrubbing affect false positives in functional connectivity studies? *BioRxiv* **2020**. [CrossRef]
30. Craddock, C.; Benhajali, Y.; Chu, C.; Chouinard, F.; Evans, A.; Jakab, A.; Khundrakpam, B.S.; Lewis, J.D.; Li, Q.; Milham, M.; et al. The Neuro Bureau Preprocessing Initiative: Open sharing of preprocessed neuroimaging data and derivatives. *Front. Neuroinform.* **2013**, *7*, 5.
31. Xu, T.; Yang, Z.; Jiang, L.; Xing, X.X.; Zuo, X.N. A connectome computation system for discovery science of brain. *Sci. Bull.* **2015**, *60*, 86–95. [CrossRef]
32. Greve, D.N.; Fischl, B. Accurate and robust brain image alignment using boundary-based registration. *Neuroimage* **2009**, *48*, 63–72. [CrossRef]
33. Jenkinson, M.; Beckmann, C.F.; Behrens, T.E.; Woolrich, M.W.; Smith, S.M. FSL. *Neuroimage* **2012**, *62*, 782–790. [CrossRef]
34. Yan, C.; Zang, Y. DPARSF: A MATLAB toolbox for "pipeline" data analysis of resting-state fMRI. *Front. Syst. Neurosci.* **2010**, *4*, 1377. [CrossRef]
35. Ashburner, J. A fast diffeomorphic image registration algorithm. *Neuroimage* **2007**, *38*, 95–113. [CrossRef] [PubMed]
36. Zang, Y.; Jiang, T.; Lu, Y.; He, Y.; Tian, L. Regional homogeneity approach to fMRI data analysis. *NeuroImage* **2004**, *22*, 394–400. [CrossRef] [PubMed]
37. Kendall, M.; Gibbons, J.D. *Rank Correlation Methods*; Oxford University Press: New York, NY, USA, 1990.
38. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [CrossRef] [PubMed]
39. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Paper II

Efficient Brain Age Prediction from 3D MRI Volumes Using 2D Projections

Johan Jönemo
Muhammad Usman Akbar
Robin Kämpe
J. Paul Hamilton
Anders Eklund

Brain Sci. **2023**, *13*(9), 1329; doi: 10.3390/brainsci13091329

[2] Available at https://creativecommons.org/licenses/by/4.0/

brain sciences

MDPI

Brief Report

Efficient Brain Age Prediction from 3D MRI Volumes Using 2D Projections

Johan Jönemo [1,2], Muhammad Usman Akbar [1,2], Robin Kämpe [2,3], J. Paul Hamilton [4] and Anders Eklund [1,2,5,*]

[1] Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, 581 83 Linköping, Sweden
[2] Center for Medical Image Science and Visualization (CMIV), Linköping University, 581 83 Linköping, Sweden
[3] Center for Social and Affective Neuroscience, Department of Biomedical and Clinical Sciences, Linköping University, 581 83 Linköping, Sweden
[4] Department of Biological and Medical Psychology, University of Bergen, 5020 Bergen, Norway
[5] Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, 581 83 Linköping, Sweden
* Correspondence: anders.eklund@liu.se

Abstract: Using 3D CNNs on high-resolution medical volumes is very computationally demanding, especially for large datasets like UK Biobank, which aims to scan 100,000 subjects. Here, we demonstrate that using 2D CNNs on a few 2D projections (representing mean and standard deviation across axial, sagittal and coronal slices) of 3D volumes leads to reasonable test accuracy (mean absolute error of about 3.5 years) when predicting age from brain volumes. Using our approach, one training epoch with 20,324 subjects takes 20–50 s using a single GPU, which is two orders of magnitude faster than a small 3D CNN. This speedup is explained by the fact that 3D brain volumes contain a lot of redundant information, which can be efficiently compressed using 2D projections. These results are important for researchers who do not have access to expensive GPU hardware for 3D CNNs.

Keywords: brain age; 3D CNN; 2D projections; deep learning

check for updates

Citation: Jönemo, J.; Akbar, M.U.; Kämpe, R.; Hamilton, J.P.; Eklund, A. Efficient Brain Age Prediction from 3D MRI Volumes Using 2D Projections. *Brain Sci.* **2023**, *13*, 1329. https://doi.org/10.3390/brainsci13091329

Academic Editor: Mohammad Tanveer

Received: 13 August 2023
Revised: 9 September 2023
Accepted: 13 September 2023
Published: 15 September 2023

1. Introduction

Predicting brain age from magnetic resonance imaging (MRI) volumes using deep learning has become a popular research topic recently [1–13]; see Tanveer et al. [14] for a recent review. More traditional machine learning methods such as regression (often using different features such as the size of different brain regions) have also been used for predicting brain age [15–17]. If there is a large difference between the predicted brain age and the biological age of a patient, one can suspect that some disease is present and the difference is therefore an important biomarker [4,18,19]. The motivation behind this is that the brain may age more quickly due to different diseases. Virtually all of the previous deep-learning-based works have used 3D convolutional neural networks (CNNs) to predict brain age, or trained 2D CNNs on all slices in each volume and then combined all the slice predictions for a prediction for the entire volume [2,6,9]. Since 3D CNNs are computationally demanding and require a lot of GPU memory, we therefore propose to instead use 2D projections of the 3D volumes. Compared to previous approaches that use 2D CNNs on volume data [2,6,9], we only use 1–6 images per patient (compared to using all 100–300 slices in a volume).

Using 2D CNNs has many benefits compared to 3D CNNs. For example, 2D CNNs can use cheaper hardware (important for low-income countries), can use networks pre-trained on ImageNet or RadImageNet [20] (there are very few pre-trained 3D CNNs) and in general benefit from the more mature and better optimized 2D CNN ecosystem. They can also have fewer parameters (which can benefit federated learning due to lower bandwith consumption). Furthermore, due to the faster training it is much easier to tune the hyperparameters.

Langner et al. [21] demonstrated that 2D projections of full-body MRI volumes can be used to train 2D CNNs to predict different measures like age. Since brain volumes contain less anatomical variation compared to full-body volumes, it is not clear if the same approach is well suited for brain volumes. Furthermore, Langner et al. only used mean intensity projections, while we also use the standard deviation projections (to better capture the variation between slices).

2. Materials and Methods

2.1. Data

The experiments in this paper are based on T1-weighted brain volumes from 29,035 subjects in UK Biobank [22–24]. The age range is 44–82 years with a resolution of 1 year; see Figure 1 for the age distribution. The subjects were divided into 20,324 for training, 4356 for validation and 4355 for testing. FSL FAST [25] was used for each skull-stripped volume, to obtain maps of gray matter (as they have proven to yield better age predictions compared to raw MRI volumes). These gray matter volumes were zeropadded, symmetrically, to match the largest grid (matrix size), resulting in volumes of 256 × 256 × 208 voxels. Each volume was then projected into six 2D images, which represent the mean and standard deviation across axial, sagittal and coronal slices (for one subject at a time). See Figure 2 for the six projections of one subject. The original dataset is about 1.5 TB as 32 bit floats.

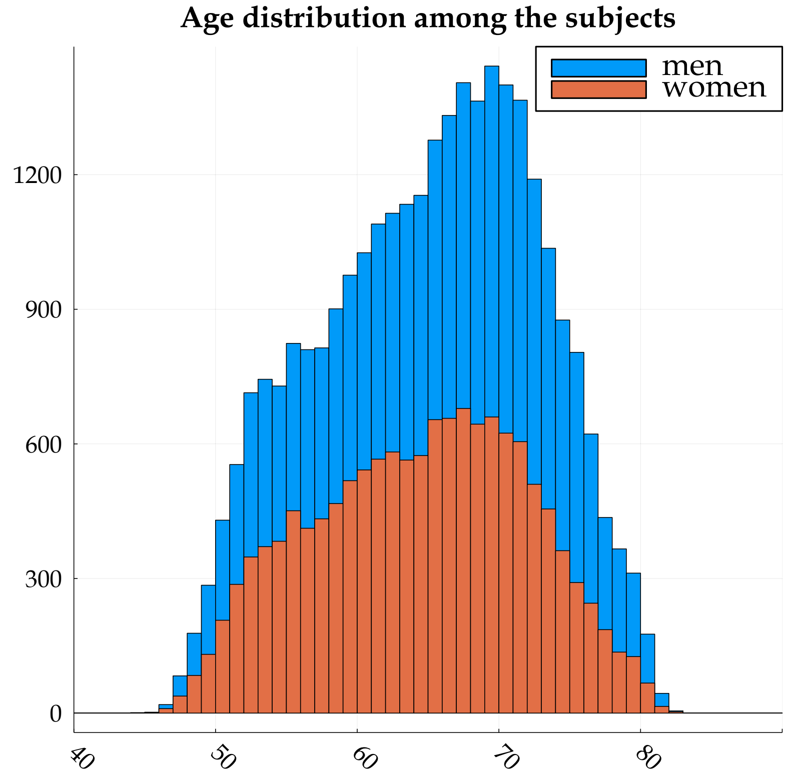


Figure 1. Age distribution for the 29,035 subjects used in this work. The individual bars are further divided to reflect the proportion of each gender within that age group.

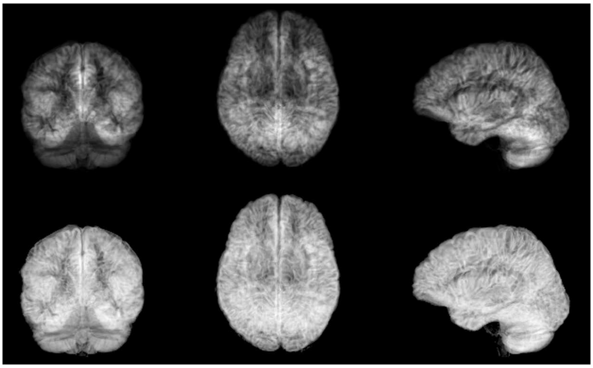


Figure 2. Top: mean grey matter likelihood projections on coronal, axial and sagittal planes, for one subject. **Bottom:** standard deviation grey matter likelihood projections on coronal, axial and sagittal planes, for the same subject.

2.2. Two-Dimensional Projections

In this work, we implemented a set of 2D CNNs using the Julia programming language (version 1.6.4) [26] and the Flux machine learning framework (version 0.12.8) [27], wherein the aforementioned projections—typically with two channels each—were fed into their respective stack of convolutional and auxiliary layers (see Figure 3). Instead of training a single multi-channel CNN, three separate CNNs were trained as the important features for sagittal images may be different from the important features for axial images, for example. Each CNN produced 256 features, which were concatenated and fed into a fully connected layer ending in one node with linear output.

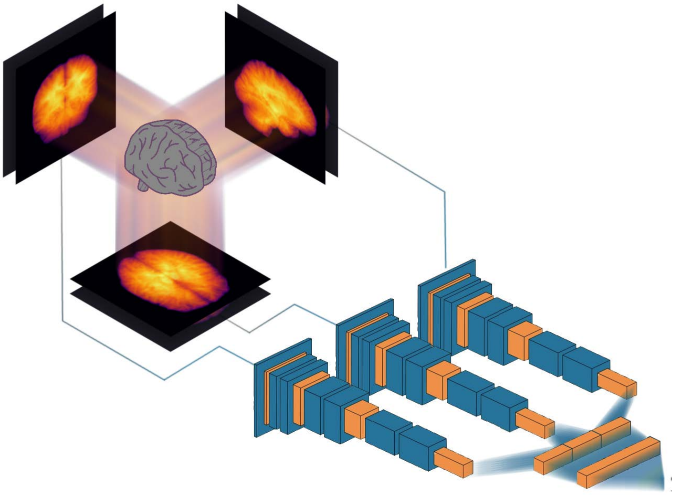


Figure 3. Our proposed approach to obtain efficient brain age prediction using 2D projections of 3D volumes. Each volume is summarized as six 2D images, which represent the mean and standard deviation across axial, sagittal and coronal slices. These 2D images are then fed into three 2D CNNs, and the resulting feature vectors are concatenated and fed into a fully connected layer to predict the brain age.

The models tested had 13 convolutional layers for each projection (axial, coronal or sagittal). The convolutional stacks had 4 filters in the first layer, which then progressed as the resolution was reduced to 256 filters as mentioned earlier. To explore how some hyperparameters affect the accuracy, the number of convolutional layers was increased to 19 and 25. Furthermore, the number of filters per convolutional layer was also decreased by 50% or increased by 100%. The models had from a little more than 0.8 million to over 8 million trainable parameters.

The training was performed using mean squared error (MSE) as a loss function. Batch normalization and dropout regularization (probability 0.2) were used after every second (or for the models with more layers, third or fourth) convolutional layer, or between the dense layers (probability 0.3 or 0.5). In all cases, the layers follow the order convolution/dense layer → batch normalization → activation → dropout → convolution/dense layer, in accordance with the usage in the articles introducing batch normalization and dropout [28,29]. It has been demonstrated that using dropout and batch normalization together can cause disharmony, but we believe this phenomenon to be alleviated by the layers following the dropout that precede the next batch normalization, especially since these layers always include an increase in the number of features, which Li et al. indicate would be helpful [30]. The dropout rate was arrived at empirically during preliminary tests (not published in this article), which also seems to belie any significant dysergies. Optimization was carried out using the Adam optimizer, with a learning rate of 0.003. Training was always performed for 400 epochs, and the weights were saved every time the validation loss decreased. Furthermore, the training was also performed where the weights of the three 2D CNNs were fixed to be the same (here called iso).

Data augmentation was tentatively explored using the Augmentor module [31], wherein an augmentation pipeline was constructed. The augmented data set consisted of the unaugmented set concatenated with three copies that had been passed through a pipeline of small random pertubations in the form of scaling, shearing, rotation and elastic deformation. This set was randomly shuffled for each epoch of training. As of yet, the code has not successfully been made to work with on-the-fly augmentation, nor have we been able to utilize GPUs for these calculations.

Training the networks was performed using an Nvidia (USA) RTX 8000 graphics card with 48 GB of memory. A major benefit of our approach is that all the training images fit in GPU memory (when augmentation was not used), making the training substantially faster since the images did not need to be streamed from the main memory or from the hard drive. One epoch of training with 6 projections from 20,324 subjects took 20–50 s for models with 13 convolution layers per projection (which can be compared to 1 hour for a 3D CNN trained with 12,949 subjects [7]). Our code is available at https://github.com/emojjon/brain-projection-age (accessed on 1 September 2023), and a Julia code for an example network is given in Figure 4.

```
 1 Chain(
 2   Parallel(
 3     var"#61#77"(),
 4 Chain(
 5       Conv((3, 3), 2 => 4, σ, pad=1),   # 76 parameters
 6       Conv((3, 3), 4 => 4, pad=1, stride=2, bias=false),  # 144 parameters
 7       BatchNorm(4, σ),                  # 8 parameters, plus 8
 8       Dropout(0.2),
 9       Conv((3, 3), 4 => 8, σ, pad=1),   # 296 parameters
10       Conv((3, 3), 8 => 8, pad=1, stride=2, bias=false),  # 576 parameters
11       BatchNorm(8, σ),                  # 16 parameters, plus 16
12       Dropout(0.2),
13       Conv((3, 3), 8 => 16, σ, pad=1),  # 1_168 parameters
14       Conv((3, 3), 16 => 16, pad=1, stride=2, bias=false),  # 2_304  p
15       BatchNorm(16, σ),                 # 32 parameters, plus 32
16       Dropout(0.2),
17       Conv((3, 3), 16 => 32, σ, pad=1),  # 4_640 parameters
18       Conv((3, 3), 32 => 32, pad=1, stride=2, bias=false),  # 9_216  p
19       BatchNorm(32, σ),                 # 64 parameters, plus 64
20       Dropout(0.2),
21       Conv((3, 3), 32 => 64, σ, pad=1),  # 18_496 parameters
22       Conv((3, 3), 64 => 64, pad=1, stride=2, bias=false),  # 36_864  p
23       BatchNorm(64, σ),                 # 128 parameters, plus 128
24       Dropout(0.2),
25       Conv((3, 3), 64 => 128, σ, pad=1),  # 73_856 parameters
26       Conv((3, 3), 128 => 128, pad=1, stride=2, bias=false),  # 147_456  p
27       BatchNorm(128, σ),                # 256 parameters, plus 256
28       Dropout(0.2),
29       Conv((4, 4), 128 => 256, σ),      # 524_544 parameters
30       var"#52#67"(),
31     ),
32     Chain(
33             # Omitted for brevity
34     ),
35     Chain(
36             # Omitted for brevity
37     ),
38   ),
39   Dense(768 => 10, σ),                  # 7_690 parameters
40   Dense(10 => 1),                       # 11 parameters
41 )         # Total: 100 trainable arrays, 2_009_369 parameters,
42           # plus 36 non-trainable, 1_512 parameters, summarysize 40.289 KiB.
43
```

Figure 4. A report on a typical network automatically generated by the Flux framework, expressed as Julia code. Here the `Parallel` structure holds the three stacks (represented by `Chain` structures within the Flux framework) of convolutional layers (and some auxiliary layers), which process axial, sagittal and coronal projections. Here, σ denotes the activation function employed after a layer. Because the three stacks are very similar, only the first one is shown. The odd-looking expressions in lines 3 and 30 are anonymous functions used to suitably reformat the data.

3. Results

Table 1 shows the test prediction accuracies and training times for previously published papers (using 3D CNNs, or 2D CNNs on all slices) and our approach using 2D projections. While several papers used the UK Biobank dataset, the test sets are different, which makes a direct comparison of the test accuracy difficult (we would need to implement and train all other networks on our specific data). Table 2 shows the results from changing the hyperparameters, and when training with fewer subjects. As expected, a smaller training set deteriorates the test accuracy. Increasing the number of filters per layer has a small positive effect, while the effect of increasing the number of convolution layers is not so clear.

Table 1. Comparison of our 2D projection approach and previous publications on brain age prediction (using 3D CNNs, or 2D CNNs on all slices), regarding number of training subjects (N), brain age test accuracy (mean absolute error (MAE) in years, RMSE in parenthesis) and training time. Iso here refers to the fact that the three parallel 2D CNNs (for axial, sagittal and coronal projections) are forced to use the same weights. Even though several publications use the UK Biobank data, a direct comparison of the test accuracy is not possible as different test sets, in terms of size and the specific subjects, were used. The available training times were rescaled to a single GPU, if multi-GPU training was mentioned. The training time for our approach is presented for early stopping, and for the full 400 epochs in parenthesis.

Paper/Settings	Approach	N Subjects	Test Accuracy	Parameters	Training Time
Huang et al., 2017 [2]	2D slices	600	4.00 MAE	-	12 h
Cole et al., 2017 [3]	3D CNN	1601	4.16 MAE	889,960	72–332 h
Wang et al., 2019 [4]	3D CNN	3688	4.45 MAE	-	30 h
Jonsson et al., 2019 [5]	3D CNN	809	3.39 MAE	-	48 h
Bashyam et al., 2020 [6]	2D slices	9383	3.70 MAE	-	10 h
Peng et al., 2021 [7]	3D CNN	12,949	2.14 MAE	3 million	130 h
Bellantuono et al., 2021 [8]	Dense	800	2.19 MAE	-	-
Gupta et al., 2021 [9]	2D slices	7312	2.82 MAE	998,625	6.75 h
Ning et al., 2021 [10]	3D CNN	13,598	2.70 MAE	-	96 h
Dinsdale et al., 2021 [11]	3D CNN	12,802	2.90 MAE	-	-
Lee et al., 2022 [12]	3D CNN	1805	3.49 MAE	70,183,073	24 h
Dropout between conv 0.2 dropout rate					
Ours, 3 mean channels	2D proj	20,324	3.55 (4.49)	2,009,261	22 min (3 h 53 min)
Ours, 3 std channels	2D proj	20,324	3.51 (4.43)	2,009,261	24 min (3 h 30 min)
Ours, all 6 channels	2D proj	20,324	3.53 (4.44)	2,009,369	24 min (3 h 26 min)
Ours, all 6 channels, iso	2D proj	20,324	3.46 (4.38)	827,841	25 min (4 h 36 min)
Dropout between dense 0.3 dropout rate					
Ours, 3 mean channels	2D proj	20,324	3.70 (4.66)	2,009,261	22 min (3 h 12 min)
Ours, 3 std channels	2D proj	20,324	3.67 (4.62)	2,009,261	27 min (4 h 27 min)
Ours, all 6 channels	2D proj	20,324	3.56 (4.47)	2,009,369	27 min (3 h 32 min)
Ours, all 6 channels, iso	2D proj	20,324	3.63 (4.56)	827,841	28 min (4 h 23 min)
Dropout between conv 0.2 dropout rate trained with augmentation					
Ours, 3 mean channels	2D proj	20,324 [1]	3.44 (4.31)	2,009,261	>3 days [2]
Ours, 3 std channels	2D proj	20,324 [1]	3.40 (4.33)	2,009,261	>3 days [2]
Ours, all 6 channels	2D proj	20,324 [1]	3.47 (4.40)	2,009,369	>3 days [2]
Ours, all 6 channels, iso	2D proj	20,324 [1]	3.85 (4.80)	827,841	>3 days [2]

[1] The model is trained with an augmented set of 20,324 + 60,972 = 81,296 pseudo subjects, but all are derived from the original 20,324 subjects. [2] This was a preliminary exploration of whether augmentation was motivated. For more competitive speeds, further optimisation is required

Our approach is substantially faster compared to previously published papers, even though we are using the largest training set, while our test accuracy is worse. Using the standard deviation to produce 2D projections leads to a slightly higher accuracy, compared to using the mean across slices. Using both mean and standard deviation projections sometimes provides a small improvement, compared to only using the standard deviation. Forcing the three 2D CNNs to use the same weights (referred to as iso) sometimes leads to a higher accuracy, compared to using three independent CNNs. Data augmentation helps to further improve the accuracy, but is currently much slower. To better visualize the relationship between real and predicted age, these are plotted against each other in Figure 5 for an example model.

Table 2. Here, we show variations of other aspects of the model in order to evaluate their effect. All modifications are relative to the models in the second section of Table 1. The training time for our approach is presented for early stopping, and for the full 400 epochs in parentheses.

Settings	Approach	N Subjects	Test Accuracy	Parameters	Training Time
Dropout between conv 0.2 dropout rate trained using only 2000 subjects					
Ours, 3 mean channels	2D proj	2000	4.05 (5.09)	2,009,261	18 min (22 min)
Ours, 3 std channels	2D proj	2000	4.01 (5.08)	2,009,261	20 min (22 min)
Ours, all 6 channels	2D proj	2000	4.06 (5.13)	2,009,369	7 min (22 min)
Ours, all 6 channels, iso	2D proj	2000	4.13 (5.18)	827,841	8 min (27 min)
Dropout between conv 0.2 dropout rate trained using only 6376 subjects					
Ours, 3 mean channels	2D proj	6376	3.75 (4.74)	2,009,261	7 min (58 min)
Ours, 3 std channels	2D proj	6376	3.73 (4.72)	2,009,261	4 min (58 min)
Ours, all 6 channels	2D proj	6376	3.73 (4.73)	2,009,369	50 min (1 h 7 min)
Ours, all 6 channels, iso	2D proj	6376	3.77 (4.75)	827,841	53 min (1 h 16 min)
Dropout between conv 0.2 dropout rate half as many filters					
Ours, 3 mean channels	2D proj	20,324	3.61 (4.51)	505,037	37 min (2 h 40 min)
Ours, 3 std channels	2D proj	20,324	3.61 (4.57)	505,037	43 min (3 h 3 min)
Ours, all 6 channels	2D proj	20,324	3.49 (4.40)	505,091	17 min (3 h 10 min)
Ours, all 6 channels, iso	2D proj	20,324	3.49 (4.39)	209,167	40 min (4 h 52 min)
Dropout between conv 0.2 dropout rate twice as many filters					
Ours, 3 mean channels	2D proj	20,324	3.45 (4.39)	8,015,333	25 min (4 h 51 min)
Ours, 3 std channels	2D proj	20,324	3.45 (4.37)	8,015,333	23 min (4 h 52 min)
Ours, all 6 channels	2D proj	20,324	3.40 (4.30)	8,015,549	23 min (4 h 55 min)
Ours, all 6 channels, iso	2D proj	20,324	3.42 (4.33)	3,293,773	19 min (5 h 39 min)
Dropout between conv 0.2 dropout rate with 19 convolution layers per stack rather than 13					
Ours, 3 mean channels	2D proj	20,324	3.56 (4.50)	2,599,697	37 min (4 h 24 min)
Ours, 3 std channels	2D proj	20,324	3.49 (4.40)	2,599,697	50 min (4 h 39 min)
Ours, all 6 channels	2D proj	20,324	3.40 (4.28)	2,599,805	31 min (4 h 43 min)
Ours, all 6 channels, iso	2D proj	20,324	3.37 (4.26)	1,024,653	60 min (5 h 44 min)
Dropout between conv 0.2 dropout rate with 25 convolution layers per stack rather than 13					
Ours, 3 mean channels	2D proj	20,324	3.49 (4.41)	3,189,985	1 h 22 min (5 h 29 min)
Ours, 3 std channels	2D proj	20,324	3.47 (4.38)	3,189,985	1 h 20 min (5 h 27 min)
Ours, all 6 channels	2D proj	20,324	3.50 (4.47)	3,190,093	1 h 37 min (5 h 46 min)
Ours, all 6 channels, iso	2D proj	20,324	3.48 (4.38)	1,221,465	1 h 14 min (7 h 26 min)

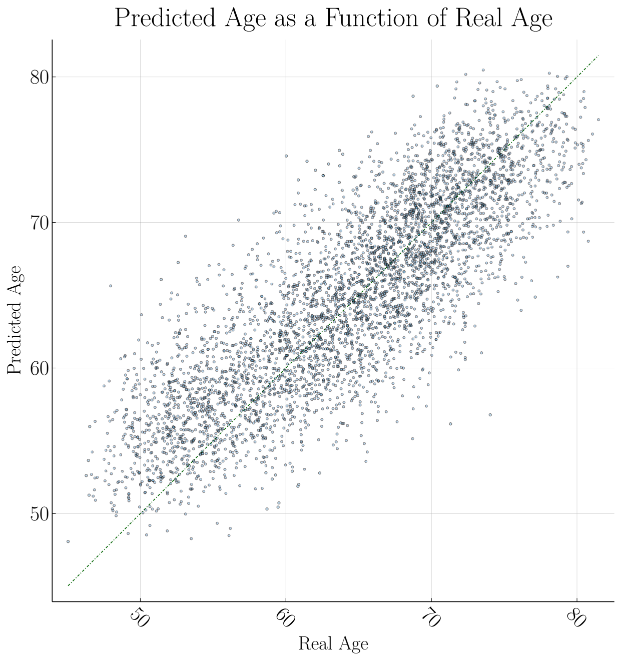


Figure 5. Comparison of real and predicted age in the test set of 4355 subjects, for a model with 19 convolution layers for each projection and using all six channels. The coefficient of determination r^2 is 0.691.

While several measures could be employed to measure the accuracy of the model, we prefer reporting the mean absolute error on the test set and have also included the root of the mean squared error on the same. This is partly because the former is the most common measure to report in models predicting brain age, and the latter was natural to include because we used the mean squared error as the loss function during training (partly because these measures have the unit years, which we feel make them more intuitive). As an example, the coefficient of determination r^2 calculated on the test set for the model visualized in Figure 5 is 0.691. It is, however, uncertain to what extent r^2 lends itself to measure non-linear models such as this.

In a preliminary study, we trained the 2D CNNs repeatedly with 1–6 input projections from the original intensity volumes (the results largely follow the same pattern as grey matter likelihood but with slightly lower accuracy) to see which projections are the most important for the network, resulting in a total of 64 combinations. This was repeated for two learning rates, for a total of 128 trainings. Figure 6 shows the decrease in loss when adding each channel, averaged over said trainings. Clearly, the standard deviation projections are more informative compared to the mean intensity projections.

In the process of training the models, RMSE for both the training set and validation set was observed. While these values are not listed for each model, we noted that for the validation set the values closely follow those for the test set. For the training set, RMSE was typically little more than half that of the test set (at early stopping), indicating some overfitting. As one might expect, this effect became more pronounced as the numbers of trainable parameters grew.

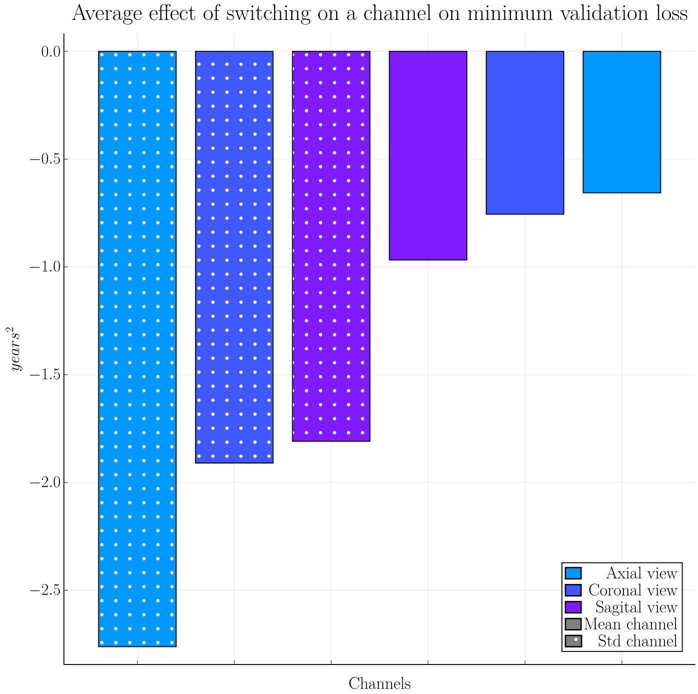


Figure 6. The effect—in the preliminary study on raw intensity volumes—of adding additional channels on the prediction accuracy, averaged over 128 trainings when using different combinations of input channels (64 different input combinations for 2 different learning rates). Adding the standard deviation images (marked with dots in this plot) from the different views has the largest effects and the mean images the smallest.

4. Discussion

Our results show that our 2D projection approach is substantially faster compared to previous work, although several papers do not report the training time. The speedup will, in our case, not be as large for GPUs with smaller memory, as it is then not possible to put all the training images in the GPU memory (for a preliminary test on a 11 GB card, the training took 3–4 times longer, but this can probably be further optimized). Nevertheless, the possibility to use cheaper hardware is important for many researchers. Compared to other 2D approaches, which use all slices in each volume, our 2D projection approach is substantially faster compared to Huang et al. [2] and Bashyam et al. [6], and our accuracy is also better. Compared to Gupta et al. [9], our approach is faster while our accuracy is lower. Our test accuracy is in general slightly worse compared to 3D CNNs, but our work should rather be seen as a proof of concept. It would be interesting to instead use 2D CNNs pre-trained on ImageNet or RadImageNet [20] as a starting point, instead of training from scratch. However, this option is currently more difficult in Flux compared to other machine learning frameworks. Yet another way to improve test accuracy is to use an ensemble of networks. Using the mean prediction of 5–10 networks will most likely improve the accuracy, while still only requiring about 125–250 min of training.

Although our proposed solution results in a lower accuracy compared to much more time-consuming 3D approaches, an approximate brain age estimate can still be valuable for diagnostic purposes. For example, if a person's biological age is 35 years and the predicted brain age is 50 years, a slightly lower or higher prediction will still lead to the conclusion that the person's brain is abnormal.

Langner et al. [21], who used 2D projections of full-body MRI scans (not including the head), obtained a mean absolute error of 2.49 years when training with 23,120 subjects from UK Biobank (training the network took about 8 h). It is difficult to determine if the higher accuracy compared to our work is due to using a VGG16 architecture (pre-trained on ImageNet), or due to the fact that full-body scans contain more information regarding a person's age, or that the full-body scans in UK Biobank contain separate images representing fat and water. No comparison with a 3D CNN is included in their work.

The demographic in the UK Biobank dataset is relatively homogenous (94.6% of participants were of white ethnicity) and there is evidence of a "healthy volunteer" selection bias [32]. Our 2D projection models are therefore expected to perform less well when applied to data from a more diverse population (e.g., regarding neurological disease, brain size, ethnicity, age). However, this is also true for 3D CNNs trained on UK Biobank data. Whether 2D or 3D CNNs are more affected by a more diverse dataset will be explored in future research.

In future work, we also plan to investigate the effect of adding additional images (channels) that represent the third and fourth moment (skew and kurtosis) across slices, since the results indicate that the standard deviation images are more informative compared to the mean intensity images. Another idea is to use principal component analysis (PCA) across each direction, to instead use eigen slices that represent most of the variance. As can be seen in Table 1, adding more channels will not substantially increase the training time as a higher number of input channels will only affect the first layer of each 2D CNN. This is different from adding more training images to a 2D CNN using each slice in a volume independently, where the training time will increase more or less linearly with more images.

5. Conclusions

The conclusion is that using 2D projections from 3D volumes results in large speedups, compared to 3D CNNs. The accuracy is slightly lower with our approach, but we believe that the results can still be used to, for example, detect abnormal brains.

Author Contributions: Conceptualization, A.E.; methodology, J.J., M.U.A., R.K. and A.E.; formal analysis, J.J.; resources, A.E., R.K. and J.P.H.; data curation, R.K. and J.P.H.; writing—original draft preparation, J.J. and A.E.; writing—review and editing, all authors; visualization, J.J.; supervision, M.U.A., A.E. and J.P.H.; project administration, A.E., R.K. and J.P.H.; funding acquisition, A.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the ITEA/VINNOVA project ASSIST (Automation, Surgery Support and Intuitive 3D visualization to optimize workflow in IGT SysTems, 2021-01954), and the Åke Wiberg foundation.

Institutional Review Board Statement: This research study was ethically approved according to the Swedish ethical review authority, application number 2017/17-31.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study, see UK Biobank for details.

Data Availability Statement: The data used in this work are available through UK Biobank, https://www.ukbiobank.ac.uk/.

Conflicts of Interest: AE has previously received graphics hardware from Nvidia. The authors declare no other conflict of interest. The funders had no role in the design of the study; in the collection, analysis, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Bjørk, M.B.; Kvaal, S.I. CT and MR imaging used in age estimation: A systematic review. *J. Forensic Odonto-Stomatol.* **2018**, *36*, 14.
2. Huang, T.W.; Chen, H.T.; Fujimoto, R.; Ito, K.; Wu, K.; Sato, K.; Taki, Y.; Fukuda, H.; Aoki, T. Age estimation from brain MRI images using deep learning. In Proceedings of the IEEE International Symposium on Biomedical Imaging (ISBI), Melbourne, Australia, 18–21 April 2017; pp. 849–852.
3. Cole, J.H.; Poudel, R.P.; Tsagkrasoulis, D.; Caan, M.W.; Steves, C.; Spector, T.D.; Montana, G. Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker. *NeuroImage* **2017**, *163*, 115–124. [CrossRef]
4. Wang, J.; Knol, M.J.; Tiulpin, A.; Dubost, F.; de Bruijne, M.; Vernooij, M.W.; Adams, H.H.; Ikram, M.A.; Niessen, W.J.; Roshchupkin, G.V. Gray matter age prediction as a biomarker for risk of dementia. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 21213–21218. [CrossRef]
5. Jónsson, B.A.; Bjornsdottir, G.; Thorgeirsson, T.; Ellingsen, L.M.; Walters, G.B.; Gudbjartsson, D.; Stefansson, H.; Stefansson, K.; Ulfarsson, M. Brain age prediction using deep learning uncovers associated sequence variants. *Nat. Commun.* **2019**, *10*, 5409. [CrossRef]
6. Bashyam, V.M.; Erus, G.; Doshi, J.; Habes, M.; Nasrallah, I.M.; Truelove-Hill, M.; Srinivasan, D.; Mamourian, L.; Pomponio, R.; Fan, Y.; et al. MRI signatures of brain age and disease over the lifespan based on a deep brain network and 14 468 individuals worldwide. *Brain* **2020**, *143*, 2312–2324. [CrossRef]
7. Peng, H.; Gong, W.; Beckmann, C.F.; Vedaldi, A.; Smith, S.M. Accurate brain age prediction with lightweight deep neural networks. *Med. Image Anal.* **2021**, *68*, 101871. [CrossRef]
8. Bellantuono, L.; Marzano, L.; La Rocca, M.; Duncan, D.; Lombardi, A.; Maggipinto, T.; Monaco, A.; Tangaro, S.; Amoroso, N.; Bellotti, R. Predicting brain age with complex networks: From adolescence to adulthood. *NeuroImage* **2021**, *225*, 117458. [CrossRef]
9. Gupta, U.; Lam, P.K.; Ver Steeg, G.; Thompson, P.M. Improved brain age estimation with slice-based set networks. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 840–844.
10. Ning, K.; Duffy, B.A.; Franklin, M.; Matloff, W.; Zhao, L.; Arzouni, N.; Sun, F.; Toga, A.W. Improving brain age estimates with deep learning leads to identification of novel genetic factors associated with brain aging. *Neurobiol. Aging* **2021**, *105*, 199–204. [CrossRef]
11. Dinsdale, N.K.; Bluemke, E.; Smith, S.M.; Arya, Z.; Vidaurre, D.; Jenkinson, M.; Namburete, A.I. Learning patterns of the ageing brain in MRI using deep convolutional networks. *NeuroImage* **2021**, *224*, 117401. [CrossRef] [PubMed]
12. Lee, J.; Burkett, B.J.; Min, H.K.; Senjem, M.L.; Lundt, E.S.; Botha, H.; Graff-Radford, J.; Barnard, L.R.; Gunter, J.L.; Schwarz, C.G.; et al. Deep learning-based brain age prediction in normal aging and dementia. *Nat. Aging* **2022**, *2*, 412–424. [CrossRef] [PubMed]
13. Pilli, R.; Goel, T.; Murugan, R.; Tanveer, M. Association of white matter volume with brain age classification using deep learning network and region wise analysis. *Eng. Appl. Artif. Intell.* **2023**, *125*, 106596. [CrossRef]
14. Tanveer, M.; Ganaie, M.; Beheshti, I.; Goel, T.; Ahmad, N.; Lai, K.T.; Huang, K.; Zhang, Y.D.; Del Ser, J.; Lin, C.T. Deep learning for brain age estimation: A systematic review. *Inf. Fusion* **2023**, *96*, 130–143. [CrossRef]
15. Beheshti, I.; Ganaie, M.; Paliwal, V.; Rastogi, A.; Razzak, I.; Tanveer, M. Predicting brain age using machine learning algorithms: A comprehensive evaluation. *IEEE J. Biomed. Health Inform.* **2021**, *26*, 1432–1440. [CrossRef]
16. Ganaie, M.; Tanveer, M.; Beheshti, I. Brain age prediction with improved least squares twin SVR. *IEEE J. Biomed. Health Inform.* **2022**, *27*, 1661–1669. [CrossRef] [PubMed]
17. Ganaie, M.; Tanveer, M.; Beheshti, I. Brain age prediction using improved twin SVR. *Neural Comput. Appl.* **2022**, 1–11. [CrossRef]
18. Cole, J.H.; Ritchie, S.J.; Bastin, M.E.; Hernández, V.; Muñoz Maniega, S.; Royle, N.; Corley, J.; Pattie, A.; Harris, S.E.; Zhang, Q.; et al. Brain age predicts mortality. *Mol. Psychiatry* **2018**, *23*, 1385–1392. [CrossRef] [PubMed]
19. Franke, K.; Gaser, C. Ten years of BrainAGE as a neuroimaging biomarker of brain aging: What insights have we gained? *Front. Neurol.* **2019**, *10*, 789. [CrossRef]
20. Mei, X.; Liu, Z.; Robson, P.M.; Marinelli, B.; Huang, M.; Doshi, A.; Jacobi, A.; Cao, C.; Link, K.E.; Yang, T.; et al. RadImageNet: An Open Radiologic Deep Learning Research Dataset for Effective Transfer Learning. *Radiol. Artif. Intell.* **2022**, *4*, e210315. [CrossRef]
21. Langner, T.; Wikström, J.; Bjerner, T.; Ahlström, H.; Kullberg, J. Identifying morphological indicators of aging with neural networks on large-scale whole-body MRI. *IEEE Trans. Med. Imaging* **2019**, *39*, 1430–1437. [CrossRef]
22. Sudlow, C.; Gallacher, J.; Allen, N.; Beral, V.; Burton, P.; Danesh, J.; Downey, P.; Elliott, P.; Green, J.; Landray, M.; et al. UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Med.* **2015**, *12*, e1001779. [CrossRef]
23. Alfaro-Almagro, F.; Jenkinson, M.; Bangerter, N.K.; Andersson, J.L.; Griffanti, L.; Douaud, G.; Sotiropoulos, S.N.; Jbabdi, S.; Hernandez-Fernandez, M.; Vallee, E.; et al. Image processing and Quality Control for the first 10,000 brain imaging datasets from UK Biobank. *Neuroimage* **2018**, *166*, 400–424. [CrossRef]
24. Littlejohns, T.J.; Holliday, J.; Gibson, L.M.; Garratt, S.; Oesingmann, N.; Alfaro-Almagro, F.; Bell, J.D.; Boultwood, C.; Collins, R.; Conroy, M.C.; et al. The UK Biobank imaging enhancement of 100,000 participants: Rationale, data collection, management and future directions. *Nat. Commun.* **2020**, *11*, 2624. [CrossRef] [PubMed]
25. Zhang, Y.; Brady, M.; Smith, S. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Trans. Med. Imaging* **2001**, *20*, 45–57. [CrossRef] [PubMed]

26. Bezanson, J.; Edelman, A.; Karpinski, S.; Shah, V.B. Julia: A fresh approach to numerical computing. *SIAM Rev.* **2017**, *59*, 65–98. [CrossRef]
27. Innes, M. Flux: Elegant machine learning with Julia. *J. Open Source Softw.* **2018**, *3*, 602. [CrossRef]
28. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015; pp. 448–456.
29. Srivastava, N.; Hinton, G.E.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
30. Li, X.; Chen, S.; Hu, X.; Yang, J. Understanding the disharmony between dropout and batch normalization by variance shift. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2682–2690.
31. Bloice, M.D.; Stocker, C.; Holzinger, A. Augmentor: An Image Augmentation Library for Machine Learning. *J. Open Source Softw.* **2017**, *2*, 432. [CrossRef]
32. Fry, A.; Littlejohns, T.J.; Sudlow, C.; Doherty, N.; Adamska, L.; Sprosen, T.; Collins, R.; Allen, N.E. Comparison of sociodemographic and health-related characteristics of UK Biobank participants with those of the general population. *Am. J. Epidemiol.* **2017**, *186*, 1026–1034. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Paper III

Brain Age Prediction Using 2D Projections Based on Higher-Order Statistical Moments and Eigenslices from 3D Magnetic Resonance Imaging Volumes

Johan Jönemo
Anders Eklund

J. Imaging **2023**, *9*(12), 271; doi: 10.3390/jimaging9120271

[3] Available at https://creativecommons.org/licenses/by/4.0/

Journal of
Imaging

MDPI

Brief Report

Brain Age Prediction Using 2D Projections Based on Higher-Order Statistical Moments and Eigenslices from 3D Magnetic Resonance Imaging Volumes

Johan Jönemo [1,2] and Anders Eklund [1,2,3,*]

[1] Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, 581 83 Linköping, Sweden
[2] Center for Medical Image Science and Visualization (CMIV), Linköping University, 581 83 Linköping, Sweden
[3] Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, 581 83 Linköping, Sweden
[*] Correspondence: anders.eklund@liu.se

Abstract: Brain age prediction from 3D MRI volumes using deep learning has recently become a popular research topic, as brain age has been shown to be an important biomarker. Training deep networks can be very computationally demanding for large datasets like the U.K. Biobank (currently 29,035 subjects). In our previous work, it was demonstrated that using a few 2D projections (mean and standard deviation along three axes) instead of each full 3D volume leads to much faster training at the cost of a reduction in prediction accuracy. Here, we investigated if another set of 2D projections, based on higher-order statistical central moments and eigenslices, leads to a higher accuracy. Our results show that higher-order moments do not lead to a higher accuracy, but that eigenslices provide a small improvement. We also show that an ensemble of such models provides further improvement.

Keywords: brain age; 3D CNN; 2D projections; deep learning; principal component analysis; skewness; kurtosis

Citation: Jönemo, J.; Eklund, A. Brain Age Prediction Using 2D Projections Based on Higher Order Statistical Moments and Eigenslices from 3D Magnetic Resonance Imaging Volumes. *J. Imaging* **2023**, *9*, 271. https://doi.org/10.3390/jimaging9120271

Academic Editor: Ebrahim Karami

Received: 19 October 2023
Revised: 30 November 2023
Accepted: 4 December 2023
Published: 6 December 2023

1. Introduction

With the availability of large amounts of openly available magnetic resonance imaging (MRI) data and the relative ease of constructing machine learning models, many turn to training such models to estimate various metrics from MRI volumes [1]. One such metric that seems to have physiological significance in a range of conditions is brain age—that is to say, the apparent age estimated from neuroimaging data [2–4]. This was presented as an important biomarker in categorizing aging subjects by Cole in 2017 [5] and has since been investigated as a biomarker for different forms of dementia [6], where it seems particularly promising for Alzheimers (the difference in brain age and chronological age was well correlated to severity as measured by tau-protein-binding tracer positron emission tomography (tau-PET) within groups with minor cognitive impairment (MCI) and Alzheimer's disease (AD)) [7]. Other researchers have suggested that brain age is correlated with hypertension [8] and severity of depression [9,10], and that it is also predictive of the success of certain interventions for chronic pain [11]. Furthermore, an inflated brain age associated with schizophrenia has been shown to be partly reversed at the onset of medication [12,13]. There are a few recent review articles that give a more thorough explanation of the subject [14–16].

1.1. Related Work on Deep-Learning-Based Brain Age Prediction

There have indeed been many deep learning models for brain age prediction suggested in the recent literature; see Tanveer et al. for a recent review [4]. The goal has often been to minimize the mean absolute error (MAE) between predicted brain age and biological age. More traditional machine learning methods (for example, using the size of different brain regions in a standard regression model) have also been used for predicting brain age [17]. Many of the deep models use 3D convolutional neural networks (CNNs) on whole or possibly down-sampled brain MRI volumes [3,7,18–22], and a large portion of these studies have trained their models with U.K. Biobank data. Such 3D models can be very resource-demanding with respect to processing time and memory consumption, while also suffering from a less mature framework of machine learning software specializing on 3D CNNs and 3D image grid data in general. Other researchers in this field have therefore used slices in one plane from brain volumes in 2D CNNs, weighting the estimates together using some means for the total brain age [23–25]. These techniques still use the same amount of data and so can be quite slow, although likely faster than a corresponding 3D CNN. Furthermore, they have an additional problem, which is how to weight all the slices, which in turn also can be performed with a machine learning model or some other algorithm. Also, these models cannot react to patterns occurring perpendicular to the slices.

1.2. Our Previous Work

In our previous work, we examined the possibility of assessing brain age using deep learning using a limited amount of two-dimensional images derived from brain volume [26], inspired by Langner et al. [27], instead of using each full 3D volume. The result was a substantially faster training, about 25 min compared to the typical 48 h or more for using a 3D network. Howlever, the accuracy was not as good as that of some of the CNNs that we referred to in our previous paper—the best of them had an MAE of 2.14 [20] compared to about 3.40 with our projection approach—but these methods are hard to compare. For example, the model did not differ only in training and test sets (which, of course, is quite natural): it also differed in that it trained a 3D-CNN for 130 h and used an ensemble of 20 such nets.

The specific images used in our previous work [26] were maps of the mean or standard deviation of values along three axes of the brain volume (transversal, sagittal, coronal). We selected these three projections as they are natural and easy to work with. Furthermore, we believe that, for example, using only one of these projections would remove too much information. The exact nature of these values could conceivably be chosen in any number of ways, but among the ones we have tried, we have found grey matter likelihood as computed using the FSL from T1-structural volumes gives the best results. This is also a very common approach used in studies about predicting brain age (e.g., [3,18,20,28]).

1.3. This Work

In this work, we looked at more sources for similar 2D projections that could even better extract the essential information from brain volumes (to further improve the accuracy without increasing training time too much). It should be noted that we here used a looser definition of projection than both its sense in tomography, which corresponds specifically to what we here call the mean channel, and its mathematical meaning of idempotent linear transformation. By projection, we here mean a way to obtain a 2D image from a 3D volume. Figure 1 shows an overview of our 2D projection approach, which, compared to our previous method [26], uses more channels per axis. Specifically, we tried adding skewness and kurtosis to the previous mean and standard deviation maps, thus using up to four 2D projections per axis. Another idea we here pursued was to find essential information in a plane, not by in some way aggregating values along a perpendicular axis but rather by seeing each slice as an example of a two-dimensional representation blueparallel to that particular plane of its volume. In that case, the most informative projections in this set should be available for us to extract by means of principle component analysis (PCA). By seeing each slice (e.g., 256 × 256 pixels) in a volume as a long vector (e.g., length 65,536), it is possible to use PCA to obtain eigenvectors that capture as much variance as possible (of all slices in the volume). These long eigenvectors can then be reshaped back to what we here call eigenslices. We investigated the representations by up to 16 such eigenslices per axis (perpendicular to the decomposed slices).

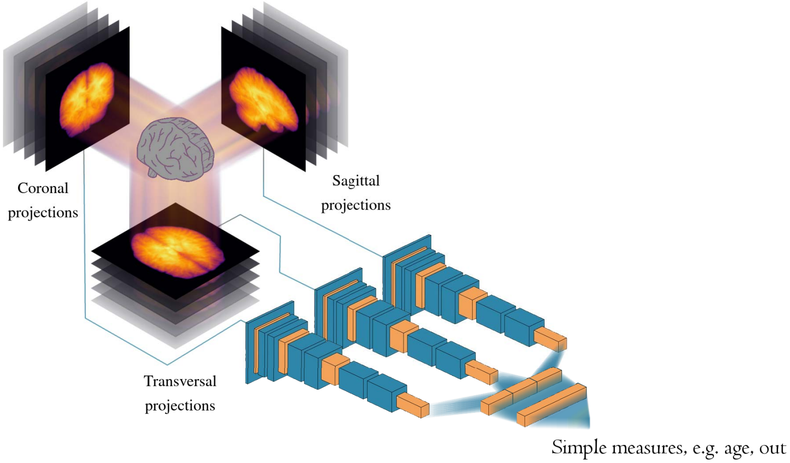


Figure 1. A conceptual illustration of our machine learning model used for brain age prediction. From each brain volume, a number of 2D images (or projections) are created by "collapsing" each of the three spatial dimensions. Two methods of collapsing each dimension were investigated in this work: calculating different statistical moments along each axis and calculating so-called eigenslices perpendicular to each axis. The images are passed to one of three stacks of convolutional and auxiliary layers corresponding to what dimension is missing. The extracted features from the three stacks are concatenated and input to a small dense network, which produces the final brain age estimate.

2. Materials and Methods

2.1. Data

Our dataset consists of 29,035 T1-weighted brain volumes from U.K. Biobank [29–32], which was also used in our previous work [26]. All subjects were scanned using one of four Siemens Skyra 3T scanners with a Siemens 32-channel RF receive head coil, available in Newcastle upon Tyne, Stockport, Reading and Bristol. The sequence used is a 3D MPRAGE sagittal sequence with TI/TR = 880/2000 ms. U.K. Biobank preprocessing of each subject included gradient distortion correction and skullstripping [31]. Due to the skullstripping, all voxels outside the brain were set to zero, meaning that any background noise was ignored. The bias field was already reduced via the on-scanner "pre-scan normalise" option. All volumes were in native space and were not registered to any template as convolutional neural networks do not require objects to be aligned in the way that statistical approaches typically do (and spatial variation can improve generalization). Because of the computation of the likelihood that any particular point inside a voxel is grey matter, all values were clamped to the closed interval from 0 to 1. No further intensity normalization was carried out as preprocessing.

The subjects were divided 70%/15%/15% for training, validation, and testing, respectively. The combined set of training and validation was partitioned in 3 different ways with mutually disjoint validation sets for cross-validation purposes. FSL FAST [33] was used for each skullstripped volume to obtain maps of grey matter. See Figure 2 for one example subject. These grey matter volumes were zero-padded, symmetrically, to match the largest grid size, resulting in volumes of 256 × 256 × 208 voxels with a size of $1 \times 1 \times 1$ mm^3.

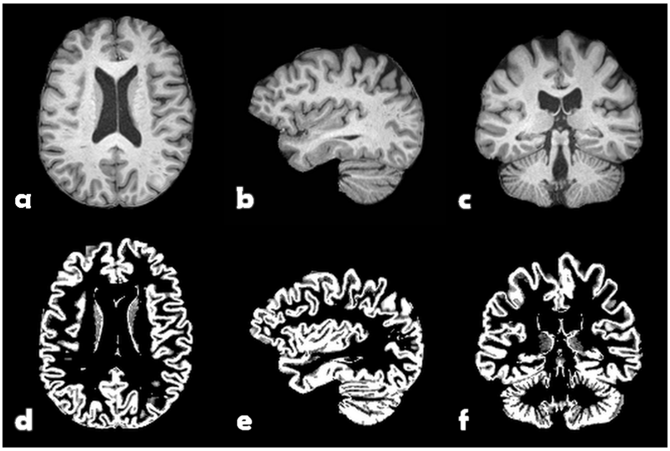


Figure 2. One example subject from U.K. Biobank. Top: Preprocessed T1-weighted volume: (**a**) transversal slice, (**b**) sagittal slice, and (**c**) coronal slice. Bottom: Grey matter probability map: (**d**) transversal slice, (**e**) sagittal slice, and (**f**) coronal slice.

2.2. Higher-Order Statistical Moments

According to our previous investigations, the standard deviation contained more information than the mean, measured by how much the mean average error differed between otherwise identical models with and without one channel [26]. For this reason, it seemed promising to include higher-order statistical moments. We used at most four moment channels. All "intensity" values used from the volumes represent grey matter likelihood. The first two channels were the mean and standard deviation of the voxels lying along a line perpendicular to the projection plane, i.e., the same as in our previous work. For measures of the third and fourth moments (skewness and kurtosis), we calculated the standardized central moments for the voxels lying within the brain, defined as the interval from the first nonzero "intensity" value up to and including the last such value along the aforementioned perpendicular line. Alternatively, for all pairs of coordinates (α, β) in an axis-aligned slice, we only considered the intensities $I_{\alpha\beta}(\gamma)$ for $\gamma \in \operatorname{hull}(\operatorname{supp}(I_{\alpha\beta}))$. When few enough values were considered, these higher moments became numerically unstable or even undefined, so we used a value of zero when the path in the brain was sufficiently short (less than 8 voxels). Using $\vec{\mathbf{x}}$ for a vector extracted from the brain volume along the dimension that is being reduced, the value in each pixel was computed as:

$$\tilde{\mu}_k(\vec{\mathbf{x}}) = n^{k/2-1} \frac{\sum\limits_{i=\alpha}^{\omega}(x_i - \mu_1)^k}{\left(\sum\limits_{i=\alpha}^{\omega}(x_i - \mu_1)^2\right)^{k/2}} - C \tag{1}$$

where k is 3 or 4, and the following definitions and conditions are in effect:

$$\alpha \text{ is the first position for which } x_i \neq 0$$
$$\omega \text{ is the last position for which } x_i \neq 0$$
$$C = \begin{cases} 0 & k = 3 \\ 3 & k = 4 \end{cases}$$
$$n \triangleq \omega - \alpha + 1$$
$$n \geq 8 \quad \left(\therefore \vec{\mathbf{x}} \neq \vec{\mathbf{0}}\right)$$

The constant C in (1), in all likelihood, makes little difference to the CNN since it uses several batch normalization layers. It is, however, common for software libraries to include the 3 for kurtosis, thereby making it the so called excess kurtosis, i.e., kurtosis in excess of that of a normal distribution. An example of the different moments is shown in Figure 3.

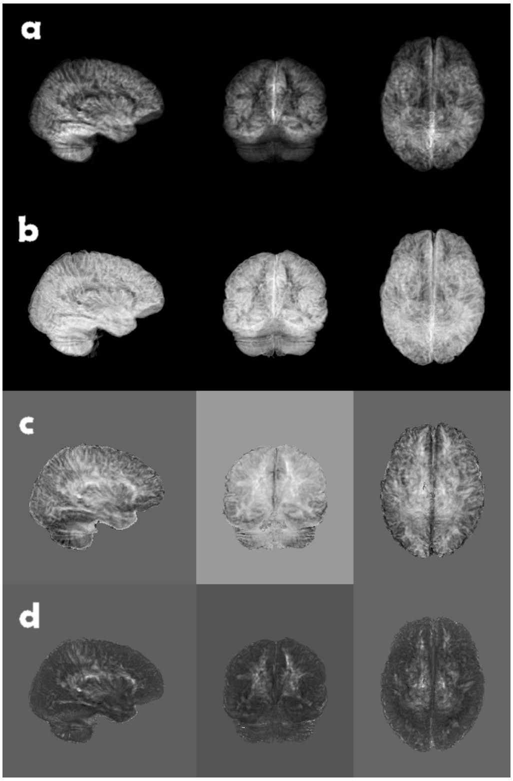


Figure 3. A brain volume of grey matter likelihood, reduced along each coordinate axis, using top (**a**) mean, (**b**) standard deviation, (**c**) skewness, and (**d**) excess kurtosis. Due to the fact that skewness and excess kurtosis can be both positive and negative in conjunction with the normalization of the grey scale, the backgrounds appear as different shades of grey.

2.3. Eigenslices

The other sort of image we here employed was produced from each brain volume, which was regarded as a stack of slices in the plane of two coordinate axes, using PCA. The motivation for this is that of all linear bases of dimension k, by construction, the one obtained by the first k eigenvectors is the one that preserves the most variation in the projected data. It could be noted that this type of 2D image does not have a very intuitive anatomic interpretation. Rather, it represents a way to reduce the amount of data while retaining a large amount of information. The problem of interpretation is thereby deferred to the deep learning network. The assumption is thus that the deep learning model can be trained to use this information, even if a human cannot.

The procedure of generating these slices is performed independently for each subject and each projection axis. If we regard each 2D-slice as a vector (rows or columns could be concatenated), we can assemble these as column vectors into a matrix $\mathbf{M}$. In terms of $\mathbf{M}$, we want to find the eigenvectors with the highest eigenvalues of $\mathbf{M}\mathbf{M}^{\mathbf{T}}$. This is a very large matrix (65,536 × 65,536 or 53,248 × 53,248, for projections of 256 × 256 and 256 × 208 pixels, respectively, but with low rank (the number of nonzero 2D slices)). We therefore employed the technique used by Sirovich et al. in 1987 and Turk et al. in 1991 in the context of facial recognition [34,35], whereby we use the following relationship:

$$\begin{gathered}\mathbf{M^T M}\overrightarrow{\mathbf{v}}_i = \lambda_i \overrightarrow{\mathbf{v}}_i \\ \Downarrow \\ \mathbf{M M^T M}\overrightarrow{\mathbf{v}}_i = \mathbf{M}\lambda_i \overrightarrow{\mathbf{v}}_i \\ \Updownarrow \\ \mathbf{M M^T}(\mathbf{M}\overrightarrow{\mathbf{v}}_i) = \lambda_i(\mathbf{M}\overrightarrow{\mathbf{v}}_i)\end{gathered}$$

In other words, if we compute the eigenpairs $(\lambda_i, \vec{\mathbf{v}}_i)$ of the much smaller $\mathbf{M^T M}$ such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, the corresponding $\mathbf{M}\vec{\mathbf{v}}_i$ is the eigenvectors originally sought, here called eigenslices. See Figure 4 for eigenslices 1 to 4 for one subject.

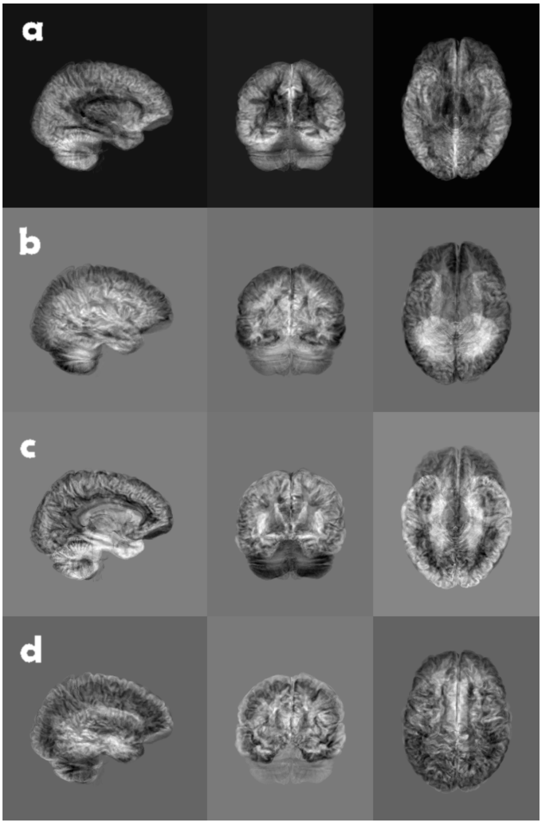


Figure 4. A brain volume of grey matter likelihood, reduced along each coordinate axis, using, from top to bottom, (**a**) eigenslice 1, (**b**) eigenslice 2, (**c**) eigenslice 3, and (**d**) eigenslice 4. The eigenslices were obtained using principal component analysis, where each slice in a volume was seen as a long vector. The eigenslices were calculated for one volume (subject) at a time.

2.4. Two-Dimensional Projection CNN

Figure 1 shows an overview of our 2D projection approach, where the statistical moments or eigenslices are used for brain age prediction in a 2D CNN with three stacks (one per axis: transversal, sagittal, and coronal). Both the generation of the two-dimensional images and all machine learning models were implemented in and run with Julia version 1.8.5 [36]. In the machine learning parts, we made use of the Julia module Flux version 0.13.11 [37]. The training was performed on an Nvidia RTX 8000 graphics card with 48 GB of memory.

Once the two-dimensional images are obtained, they are cached to permanent storage to obviate the need to compute them repeatedly and fed to a machine learning model with three parallel 2D CNN stacks for the different "projection" planes. The stacks are made up of units of convolution → activation → convolution → batch-normalization → activation → dropout, each of which doubles the number of features and halved the resolution along each axis. It also has a capping module to produce a one-dimensional feature vector, which contains another convolutional layer. The model then aggregates the features from all three views and produces an age estimate. It can use either mean square error or mean absolute error as the loss function for training. The used 2D CNN has 13 convolutional layers with 4 filters in the first layer. For more details, see our prior work [26].

The hyperparameters were optimized manually. Different positions for the dropout layers and different dropout rates comparisons are shown in our earlier article [26]. Optimization was performed using the Adam optimiser with a learning rate of 0.003 [38] and a batch size of 32. All models were trained for 400 epochs, but the model state after training was chosen to be the one with the best validation accuracy, as in early stopping. The constant epoch training was performed mainly for more complete speed metrics. Seeing as these models take relatively little time to train and that we already had several of them trained and saved, we also looked at if the models could be used in ensemble to further improve the accuracy. The used code is available at (accessed on 1 December 2023) https://github.com/emojjon/eigen-moments-brain-age.

3. Results

The network was trained repeatedly with different combinations of in-channels. Every variation was furthermore trained several times in order to estimate a measure of dispersion (except for the ensembles, as this would require much more time). All trainings here used the corresponding channels for all three projections (although having different combinations of channels per projection is also supported). Channels 1 to n were used, or just channel n for $n \in \{1, \ldots, 8\}$, for the eigenslices and $n \in \{1, \ldots, 4\}$ for the moments.

It should be said that the trainings were initially run—and possibly rerun—to give an overview to us as researchers and possibly to suggest improvements (for convenience, the models remained mathematically equivalent). After that, new trainings were run so that at least four trainings exists for every combination of parameters here presented. In the cases where more trainings had already been run, all were kept, as having more measurements does not per se affect the expected value of the mean or the standard deviation, if correctly computed.

The results are visualized in Figure 5 and presented in a more comprehensive form in Table 1. Using higher-order moments does not seem to improve the accuracy compared to using mean and standard deviation. As expected, when only using a single eigenslice, the accuracy deteriorates for higher-order eigenslices, as these eigenslices represent less and less of the variance. Using the first two eigenslices leads to a slightly better accuracy (MAE of 3.36 years) compared to using mean and standard deviation (MAE of 3.47 years). Somewhat surprisingly, the performance is reduced when using increasingly more eigenslices together.

Some further variations were preliminarily evaluated but discontinued because they did not provide any advantages. This included all runs involving eigenslices 9 to 16 and—perhaps surprisingly—runs with mean absolute error as the loss function.

The training times for these models were comparatively short. Both the early stopping time and the time to train for 400 epochs, which was considered to be enough to train any of the models, are listed in Table 1. A leap in training time was typically seen between using three channels per projection axis and using four. This is due to the fact that the model estimates the amount of GPU memory needed to fit the training data and resorts to a strategy of uploading smaller parts to GPU memory during each epoch of training should this amount not be available.

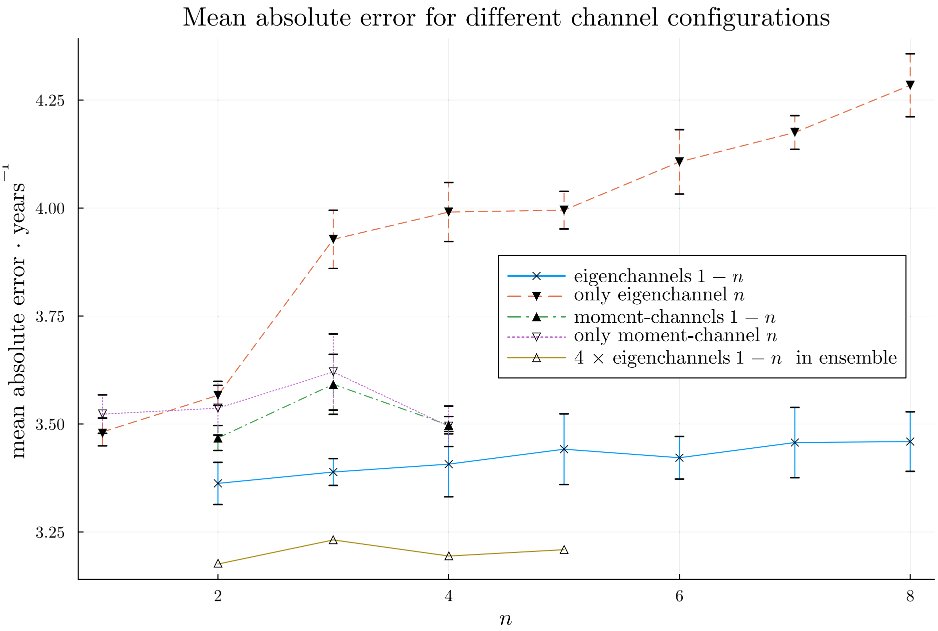


Figure 5. The average MAE for models trained with n channels (along each projection axis) or only the channel numbered n (along each projection axis). The standard deviation was not calculated for the ensembles, but it should be expected to be between $\frac{1}{\sqrt{4}}$ and 1 times that of the eigenchannels 1 to n series for purely algebraic reasons. These values are also shown in Table 1.

We also tried to use four trained models in ensembles for all models trained with the 2–5 first eigenchannels. Only one ensemble of each kind was evaluated, where there is no standard deviation for the MAE. The corresponding dispersion measure for the constituent models transpired from the table. Clearly, the accuracy improved compared to using a single model. For example, the MAE decreased from 3.36 to 3.18 years when using the two first eigenslices.

Table 1. Results for our 2D projection approach regarding number of training subjects (N), brain age test accuracy (mean absolute error (MAE) in years, RMSE in parenthesis), and training time.The means and standard deviations in the eigenchannel parts were derived from sets of at least 4 trainings (if more than 4 such trainings, for various reasons, had been run, these values were also included to obtain better estimates); no set contained more than 11 trainings), whereas the corresponding values for the moment parts were derived from 4 trainings per row. Even though several publications use the U.K. Biobank data, a direct comparison of the test accuracy was not possible as different test sets, in terms of size and the specific subjects, were used. The training times refer to running on a single GPU. The training times are presented for early stopping and for the full 400 epochs in parentheses.

Input	No. Subjects	Test Accuracy	Parameters	Training Time
”moment” channels from 1 to n				
2	20,325	3.47 ± 0.029 (4.38 ± 0.049)	2,009,369	28 m 44 s (3 h 19 m)
3	20,325	3.59 ± 0.069 (4.51 ± 0.079)	2,009,477	1 h 27 m (5 h 7 m)
4	20,325	3.50 ± 0.020 (4.44 ± 0.031)	2,009,585	1 h 50 m (21 h 23 m)
”moment” channel n only				
1	20,325	3.52 ± 0.044 (4.45 ± 0.042)	2,009,261	20 m 51 s (2 h 49 m)
2	20,325	3.54 ± 0.062 (4.46 ± 0.071)	2,009,261	17 m 9 s (3 h 3 m)
3	20,325	3.62 ± 0.088 (4.58 ± 0.081)	2,009,261	2 h 28 m(3 h 48 m)
4	20,325	3.49 ± 0.047 (4.44 ± 0.040)	2,009,261	1 h 39 m (20 h 44 m)
“eigenchannels” from 1 to n				
2	20,325	3.36 ± 0.049 (4.25 ± 0.055)	2,009,369	19 m 53 s (3 h 9 m)
3	20,325	3.39 ± 0.031 (4.28 ± 0.040)	2,009,477	28 m 5 s (4 h 19 m)
4	20,325	3.41 ± 0.076 (4.32 ± 0.088)	2,009,585	1 h 59 m (17 h 5 m)
5	20,325	3.44 ± 0.082 (4.35 ± 0.096)	2,009,693	3 h 57 m (1 d 7 h)
6	20,325	3.42 ± 0.049 (4.34 ± 0.071)	2,009,801	3 h 23 m (1 d 13 h)
7	20,325	3.46 ± 0.081 (4.37 ± 0.093)	2,009,909	6 h 1 m (2 d 2 h)
8	20,325	3.46 ± 0.069 (4.38 ± 0.092)	2,010,017	6 h 8 m (2 d 1 h)

Table 1. *Cont.*

Input	No. Subjects	Test Accuracy	Parameters	Training Time
"eigenchannel" n only				
1	20,325	3.48 ± 0.032 (4.39 ± 0.044)	2,009,261	21 m 8 s (2 h 52 m)
2	20,325	3.57 ± 0.023 (4.50 ± 0.025)	2,009,261	14 m 20 s (2 h 57 m)
3	20,325	3.93 ± 0.067 (4.94 ± 0.072)	2,009,261	20 m 34 s (4 h 39 m)
4	20,325	3.99 ± 0.068 (5.03 ± 0.082)	2,009,261	1 h 52 m (19 h 35 m)
5	20,325	4.00 ± 0.044 (5.02 ± 0.056)	2,009,261	1 h 49 m (1 d)
6	20,325	4.11 ± 0.074 (5.18 ± 0.097)	2,009,261	2 h 22 m (1 d 7 h)
7	20,325	4.17 ± 0.039 (5.22 ± 0.024)	2,009,261	4 h 39 m (2 d 20 h)
8	20,325	4.28 ± 0.073 (5.36 ± 0.090)	2,009,261	11 h 57 m (2 d 11 h)
"eigenchannels" 1 to n ensembles of 4				
2	20,325	3.18 (4.02)	8,037,476	N/A
3	20,325	3.23 (4.09)	8,037,908	N/A
4	20,325	3.19 (4.06)	8,038,340	N/A
5	20,325	3.21 (4.07)	8,038,772	N/A

4. Discussion

This is the continuation of our previous work [26], where we investigated a similar approach but using only the mean and standard deviation over each dimension to obtain six channels. In this work, we included more channels to feed into the network to improve accuracy without increasing training time too much. We added skewness and kurtosis to the projections with mean and standard deviation. We also investigated using eigenslices from the PCA of one "stack" of slices per dimension and subject.

The measure we studied here was brain age, and we trained our models with the assumption that all used subjects should be healthy and thus present a brain age equal to their biological age. It should therefore be noted that less than perfect correlation between brain age and biological age in healthy subjects, as well as a less than perfect classification of who is "healthy", would be part of the error of the models. Furthermore, even a model that could predict the biological age perfectly would have an MAE of 0.25 years because of the rounding of the recorded ages to whole years (for anonymization purposes). All of this taken together means that as we refine our models, the residual deviation from a perfect result should be compared to that of an unknown "best possible result" rather than zero.

For what we here chose to call the "moment" channels (because they largely represent the first through fourth central moments of the grey matter likelihood along an axis-aligned path), the results are hard to interpret. Mainly we saw that the skewness channel seems to perform worse than the others, not only alone: it also seems to confuse models trained with channels 1 to 3 (though strangely not channels 1 to 4).

With the eigenslices, we noticed how each consecutive slice by itself (applied in all three directions) leads to a worse prediction of brain age than the one before it. This is expected as the eigenslices are sorted by their eigenvalues, which in turn should give a measure of the explanatory power of that slice. In the case of slices 1 to n, i.e., all slices up to number n, we noticed a much more even curve although there did not seem to be any benefit to including more than two or perhaps three slices in each dimension.

In general, the explanatory power of the eigenslices tapers off quite fast. This is probably because, although (roughly) corresponding eigenslices might be generated for different brains, as we proceed down the stack, the eigenvalues lie closer and closer to each other and hence the order of corresponding eigenslices can change. This would make the data much harder for our model to learn from. A possible development to the technique could be to change the order of some eigenslices so as to increase their correspondence. The exact algorithm would have to be investigated further. Another solution can in theory be to perform PCA on all subjects concurrently. However, this may be very computationally expensive and would require an extra step to obtain subject specific eigenslices from the group eigenslices. At this point, we have not verified that this extra step is possible, but it would be inspired by the dual regression (spatial and temporal) approach used for 4D functional MRI data [39]. Another idea is to perform PCA on all 20,325 training volumes to first obtain eigenvolumes and then project the volume from each subject on the k first eigenvolumes. Then, one could use these coordinates directly or use them to construct a new volume and calculate 2D projections from it.

We also looked at making ensembles of trained models. The advantage of this is contingent on how highly the errors in different models are correlated. Making ensembles of four models—each with eigenslice channels 1 to n—for a range of ns, we could see a clear improvement in the accuracy.

The measured times are potentially not representative for a model working under optimal conditions because no provisions have been made for picking and mixing channels, only for limiting them to the n first ones along each axis. This means that even if only one channel is used for the training, the lower-numbered channels would still be loaded into memory and either past in its entirety or shuttled on demand in little pieces to the GPU. Should one need to do this on a regular basis, one could write a short definition of a `projection` containing a minimal set of channels to a Julia file and include it like any other projection, specifying a cache name and making sure to provide the bundle of channels in the corresponding directory or include instructions for how to generate them and let them be created on demand.

5. Conclusions

To summarize, to use higher-order moments does not improve the results obtained in our previous study [26], where only mean and standard deviation were used. To instead use the first two eigenslices provides a small improvement, from an MAE of 3.47 to 3.36 years, compared to using mean and standard deviation (but we did not test for statistical significance). It is possible that somehow sorting the eigenslices or using eigenvolumes can further improve the results. Using an ensemble of models provides further improvement, from an MAE of 3.36 to 3.18 years, while the total training time is still much shorter compared to that of 3D CNNs.

Author Contributions: Conceptualization, A.E.; methodology, J.J. and A.E.; formal analysis, J.J.; resources, A.E. data curation, J.J.; writing—original draft preparation, J.J.; writing—review and editing, all authors; visualization, J.J.; supervision, A.E.; project administration, A.E.; funding acquisition, A.E. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the ITEA/VINNOVA-funded project Automation, Surgery Support and Intuitive 3D visualization to optimize workflow in IGT SysTems (ASSIST) (grant 2021-01954), and by the Åke Wiberg foundation (grant M22-0088).

Institutional Review Board Statement: This study was ethically approved according to the Swedish ethical review authority, application number 2017/17-31.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study, see U.K. Biobank for details.

Data Availability Statement: The data used in this work are available through U.K. Biobank, accessed on 1 September 2021, https://www.ukbiobank.ac.uk/.

Conflicts of Interest: A.E. has previously received hardware from Nvidia. Otherwise, the authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Lundervold, A.S.; Lundervold, A. An overview of deep learning in medical imaging focusing on MRI. *Z. Für Med. Phys.* **2019**, *29*, 102–127. [CrossRef] [PubMed]
2. Cole, J.H.; Franke, K. Predicting Age Using Neuroimaging: Innovative Brain Ageing Biomarkers. *Trends Neurosci.* **2017**, *40*, 681–690. [CrossRef] [PubMed]
3. Cole, J.H.; Poudel, R.P.; Tsagkrasoulis, D.; Caan, M.W.; Steves, C.; Spector, T.D.; Montana, G. Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker. *NeuroImage* **2017**, *163*, 115–124. [CrossRef] [PubMed]
4. Tanveer, M.; Ganaie, M.; Beheshti, I.; Goel, T.; Ahmad, N.; Lai, K.T.; Huang, K.; Zhang, Y.D.; Del Ser, J.; Lin, C.T. Deep learning for brain age estimation: A systematic review. *Inf. Fusion* **2023**, *96*, 130–143. [CrossRef]
5. Cole, J.H. Neuroimaging-derived brain-age: An ageing biomarker? *Aging* **2017**, *9*, 1861–1862. [CrossRef] [PubMed]
6. Etminani, K.; Soliman, A.; Davidsson, A.; Chang, J.R.; Martínez-Sanchis, B.; Byttner, S.; Camacho, V.; Bauckneht, M.; Stegeran, R.; Ressner, M.; et al. A 3D deep learning model to predict the diagnosis of dementia with Lewy bodies, Alzheimer's disease, and mild cognitive impairment using brain 18F-FDG PET. *Eur. J. Nucl. Med. Mol. Imaging* **2022**, *49*, 563–584. [CrossRef] [PubMed]
7. Lee, J.; Burkett, B.J.; Min, H.K.; Senjem, M.L.; Lundt, E.S.; Botha, H.; Graff-Radford, J.; Barnard, L.R.; Gunter, J.L.; Schwarz, C.G.; et al. Deep learning-based brain age prediction in normal aging and dementia. *Nat. Aging* **2022**, *2*, 412–424. [CrossRef]
8. Mouches, P.; Wilms, M.; Aulakh, A.; Langner, S.; Forkert, N.D. Multimodal brain age prediction fusing morphometric and imaging data and association with cardiovascular risk factors. *Front. Neurol.* **2022**, *13*, 979774. [CrossRef]
9. Han, L.K.M.; Dinga, R.; Hahn, T.; Ching, C.R.K.; Eyler, L.T.; Aftanas, L.; Aghajani, M.; Aleman, A.; Baune, B.T.; Berger, K.; et al. Brain aging in major depressive disorder: Results from the ENIGMA major depressive disorder working group. *Mol. Psychiatry* **2021**, *26*, 5124–5139. [CrossRef]
10. Dunlop, K.; Victoria, L.W.; Downar, J.; Gunning, F.M.; Liston, C. Accelerated brain aging predicts impulsivity and symptom severity in depression. *Neuropsychopharmacology* **2021**, *46*, 911–919. [CrossRef]
11. Hung, P.S.P.; Zhang, J.Y.; Noorani, A.; Walker, M.R.; Huang, M.; Zhang, J.W.; Laperriere, N.; Rudzicz, F.; Hodaie, M. Differential expression of a brain aging biomarker across discrete chronic pain disorders. *Pain* **2022**, *163*, 1468–1478. [CrossRef]
12. Man, W.; Ding, H.; Chai, C.; An, X.; Liu, F.; Qin, W.; Yu, C. Brain age gap as a potential biomarker for schizophrenia: A multi-site structural MRI study. In Proceedings of the 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Virtual, 1–5 November 2021.
13. Xi, Y.B.; Wu, X.S.; Cui, L.B.; Bai, L.J.; Gan, S.Q.; Jia, X.Y.; Li, X.; Xu, Y.Q.; Kang, X.W.; Guo, F.; et al. Neuroimaging-based brain-age prediction of first-episode schizophrenia and the alteration of brain age after early medication. *Br. J. Psychiatry* **2021**, *220*, 1–8. [CrossRef]
14. Wrigglesworth, J.; Ward, P.; Harding, I.H.; Nilaweera, D.; Wu, Z.; Woods, R.L.; Ryan, J. Factors associated with brain ageing—A systematic review. *BMC Neurol.* **2021**, *21*, 312. [CrossRef]
15. Franke, K.; Gaser, C. Ten years of BrainAGE as a neuroimaging biomarker of brain aging: What insights have we gained? *Front. Neurol.* **2019**, *10*, 789. [CrossRef]
16. Sone, D.; Beheshti, I. Neuroimaging-based brain age estimation: A promising personalized biomarker in neuropsychiatry. *J. Pers. Med.* **2022**, *12*, 1850. [CrossRef]
17. Beheshti, I.; Ganaie, M.; Paliwal, V.; Rastogi, A.; Razzak, I.; Tanveer, M. Predicting brain age using machine learning algorithms: A comprehensive evaluation. *IEEE J. Biomed. Health Inform.* **2021**, *26*, 1432–1440. [CrossRef]
18. Wang, J.; Knol, M.J.; Tiulpin, A.; Dubost, F.; de Bruijne, M.; Vernooij, M.W.; Adams, H.H.; Ikram, M.A.; Niessen, W.J.; Roshchupkin, G.V. Gray matter age prediction as a biomarker for risk of dementia. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 21213–21218. [CrossRef]
19. Jónsson, B.A.; Bjornsdottir, G.; Thorgeirsson, T.; Ellingsen, L.M.; Walters, G.B.; Gudbjartsson, D.; Stefansson, H.; Stefansson, K.; Ulfarsson, M. Brain age prediction using deep learning uncovers associated sequence variants. *Nat. Commun.* **2019**, *10*, 5409. [CrossRef]
20. Peng, H.; Gong, W.; Beckmann, C.F.; Vedaldi, A.; Smith, S.M. Accurate brain age prediction with lightweight deep neural networks. *Med. Image Anal.* **2021**, *68*, 101871. [CrossRef]
21. Ning, K.; Duffy, B.A.; Franklin, M.; Matloff, W.; Zhao, L.; Arzouni, N.; Sun, F.; Toga, A.W. Improving brain age estimates with deep learning leads to identification of novel genetic factors associated with brain aging. *Neurobiol. Aging* **2021**, *105*, 199–204. [CrossRef]
22. Dinsdale, N.K.; Bluemke, E.; Smith, S.M.; Arya, Z.; Vidaurre, D.; Jenkinson, M.; Namburete, A.I. Learning patterns of the ageing brain in MRI using deep convolutional networks. *NeuroImage* **2021**, *224*, 117401. [CrossRef]
23. Huang, T.W.; Chen, H.T.; Fujimoto, R.; Ito, K.; Wu, K.; Sato, K.; Taki, Y.; Fukuda, H.; Aoki, T. Age estimation from brain MRI images using deep learning. In Proceedings of the IEEE International Symposium on Biomedical Imaging (ISBI), Melbourne, VIC, Australia, 18–21 April 2017; pp. 849–852.

24. Bashyam, V.M.; Erus, G.; Doshi, J.; Habes, M.; Nasrallah, I.M.; Truelove-Hill, M.; Srinivasan, D.; Mamourian, L.; Pomponio, R.; Fan, Y.; et al. MRI signatures of brain age and disease over the lifespan based on a deep brain network and 14,468 individuals worldwide. *Brain* **2020**, *143*, 2312–2324. [CrossRef]
25. Gupta, U.; Lam, P.K.; Ver Steeg, G.; Thompson, P.M. Improved brain age estimation with slice-based set networks. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 840–844.
26. Jönemo, J.; Akbar, M.U.; Kämpe, R.; Hamilton, J.P.; Eklund, A. Efficient brain age prediction from 3D MRI volumes using 2D projections. *Brain Sci.* **2023**, *13*, 1329. [CrossRef]
27. Langner, T.; Wikström, J.; Bjerner, T.; Ahlström, H.; Kullberg, J. Identifying morphological indicators of aging with neural networks on large-scale whole-body MRI. *IEEE Trans. Med. Imaging* **2019**, *39*, 1430–1437. [CrossRef]
28. Doan, N.T.; Engvig, A.; Zaske, K.; Persson, K.; Lund, M.J.; Kaufmann, T.; Cordova-Palomera, A.; Alnæs, D.; Moberget, T.; Brækhus, A.; et al. Distinguishing early and late brain aging from the Alzheimer's disease spectrum: Consistent morphological patterns across independent samples. *Neuroimage* **2017**, *158*, 282–295. [CrossRef]
29. Sudlow, C.; Gallacher, J.; Allen, N.; Beral, V.; Burton, P.; Danesh, J.; Downey, P.; Elliott, P.; Green, J.; Landray, M.; et al. UK biobank: An open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Med.* **2015**, *12*, e1001779. [CrossRef]
30. Miller, K.L.; Alfaro-Almagro, F.; Bangerter, N.K.; Thomas, D.L.; Yacoub, E.; Xu, J.; Bartsch, A.J.; Jbabdi, S.; Sotiropoulos, S.N.; Andersson, J.L.; et al. Multimodal population brain imaging in the UK Biobank prospective epidemiological study. *Nat. Neurosci.* **2016**, *19*, 1523–1536. [CrossRef]
31. Alfaro-Almagro, F.; Jenkinson, M.; Bangerter, N.K.; Andersson, J.L.; Griffanti, L.; Douaud, G.; Sotiropoulos, S.N.; Jbabdi, S.; Hernandez-Fernandez, M.; Vallee, E.; et al. Image processing and Quality Control for the first 10,000 brain imaging datasets from UK Biobank. *Neuroimage* **2018**, *166*, 400–424. [CrossRef]
32. Littlejohns, T.J.; Holliday, J.; Gibson, L.M.; Garratt, S.; Oesingmann, N.; Alfaro-Almagro, F.; Bell, J.D.; Boultwood, C.; Collins, R.; Conroy, M.C.; et al. The UK Biobank imaging enhancement of 100,000 participants: Rationale, data collection, management and future directions. *Nat. Commun.* **2020**, *11*, 2624. [CrossRef]
33. Zhang, Y.; Brady, M.; Smith, S. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Trans. Med. Imaging* **2001**, *20*, 45–57. [CrossRef] [PubMed]
34. Sirovich, L.; Kirby, M. Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A* **1987**, *4*, 519–524. [CrossRef]
35. Turk, M.; Pentland, A. Eigenfaces for recognition. *J. Cogn. Neurosci.* **1991**, *3*, 71–86. [CrossRef]
36. Bezanson, J.; Edelman, A.; Karpinski, S.; Shah, V.B. Julia: A fresh approach to numerical computing. *Siam Rev.* **2017**, *59*, 65–98. [CrossRef]
37. Innes, M. Flux: Elegant machine learning with Julia. *J. Open Source Softw.* **2018**, *3*, 602. [CrossRef]
38. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
39. Beckmann, C.F.; Mackay, C.E.; Filippini, N.; Smith, S.M. Group comparison of resting-state FMRI data using multi-subject ICA and dual regression. *Neuroimage* **2009**, *47*, S148. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

FACULTY OF SCIENCE AND ENGINEERING

Linköping Studies in Science and Technology, Licentiate Thesis No. 1986, 2024
Department of Biomedical Engineering

Linköping University
SE-581 83 Linköping, Sweden

www.liu.se