José Carlos
Alho Barros Bastos

Deep Learning for the Diagnosis of Alzheimer's Disease with 18F-FDG PET Neuroimaging

Aprendizagem Profunda para o Diagnóstico da Doença de Alzheimer com Neuroimagem 18F-FDG PET

José Carlos
Alho Barros Bastos

Deep Learning for the Diagnosis of Alzheimer Disease with 18F-FDG PET Neuroimaging

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requesitos necessários à obtenção do grau de Mestrado em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Doutor Filipe Miguel Teixeira Pereira da Silva, Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro, e da Doutora Pétia Georgieva, Professora Associada do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

o júri / the jury

presidente / president

Professor Doutor Augusto Marques Ferreira da Silva
Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

vogais / examiners committee

Doutor Pedro Gabriel Dias Ferreira
Professor Auxiliar do Departamento de Ciência de Computadores da Faculdade de Ciências da Universidade do Porto (Arguente Principal)

Professor Doutor Filipe Miguel Teixeira Pereira da Silva
Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (Orientador)

agradecimentos / acknowledgements

Agradeço ao orientador Filipe Silva e à co-orientadora Pétia Georgieva por toda a paciência e apoio prestado.
Agradeço ao Prof. Vítor Santos e à equipa do LAR (DEM-UA) pela disponibilidade de recursos computacionais ao longo do meu tempo de dissertação. Desta forma, ajudaram a levar o trabalho aqui descrito a uma conclusão bem-sucedida.
Agradeço a colaboração prestada pelos Drs. Diogo Borges Faria e João Duarte Pinto, da Atrys Portugal Medicina Molecular Porto, na conceptualização do tema de dissertação que deu origem ao trabalho desenvolvido.
Por fim, quero deixar um agradecimento à minha família e amigos por todo o suporte durante o meu percurso académico.

Palavras-Chave

Doença de Alzheimer, neuroimagem FDG-PET, dataset ADNI, redes neuronais convolucionais, transferência de aprendizagem, aprendizagem profunda personalizada

Resumo

Doença neurodegenerativa é um termo utilizado para uma série de condições incuráveis e debilitantes que afetam o sistema nervoso humano. Destas condições, a doença de Alzheimer (DA) é a mais preocupante, tanto pelo número de pessoas afetadas como pelos elevados custos em tratamento médico. Os principais desafios associados a esta doença estão relacionados com os sintomas subtis, o rápido desenvolvimento de incapacidade e ao longo período de tempo durante o qual os pacientes necessitarão de cuidados especiais. Pesquisas recentes têm sido dedicadas ao desenvolvimento de ferramentas computacionais capazes de ser integradas nos procedimentos médicos como complemento para apoiar o diagnóstico precoce e tratamentos adequados. Esta dissertação procura estudar a aplicação de técnicas de aprendizagem profunda (AP) na classificação automatizada da DA. Este estudo tem como foco principal o papel da neuroimagem PET como biomarcador de doenças neurodegenerativas, especialmente na classificação de pacientes saudáveis em comparação com pacientes com DA. Imagens PET do metabolismo cerebral de glucose com flúor-18 (^{18}F) fluorodesoxiglucose (^{18}F FGD) foram obtidas através da base de dados da Alzheimer's Disease Neuroimaging Initiative (ADNI). O dataset pré-processado é usado para treinar duas redes neurais convolucionais (RNCs). A arquitetura da primeira RNC procura explorar a transferência de aprendizagem como uma solução promissora para o problema dos dados através da utilização de um modelo Inception V3 2D, da Google, previamente treinado num dataset maior. Esta abordagem requer um passo de pré-processamento onde dados volumétricos PET são convertidos numa imagem bidimensional que por sua vez será os dados de entrada do modelo pré-treinado. A segunda abordagem involve uma RNC 3D personalizada de maneira a utilizar os padrões espaciais presentes nos volumes PET através de filtros 3D e camadas de pooling 3D. O estudo comparativo foca-se no desempenho e robustez dos dois modelos ao lidar com a disponibilidade limitada de dados classificados. O desempenho dos classificadores é avaliado através de um processo de validação cruzada, atribuindo uma pontuação de 83.62% à RNC 2D e de 86.80% à RNC 3D. Os resultados obtidos contribuem para análise da eficácia destes métodos no diagnóstico da DA. Tendo em conta as melhorias expectáveis, estas poderam ser consideradas abordagens promissoras e de acordo com o atual estado da arte.

Keywords Alzheimer's disease, FDG-PET neuroimaging, ADNI dataset, convolutional neural networks, transfer learning, custom deep learning

Abstract

Neurodegenerative disease is the term used for a range of incurable and debilitating conditions affecting the human's nervous system. Amongst these conditions, Alzheimer's Disease (AD) is responsible for the greatest burden both for the number of people affected and for the high costs in medical care. The challenges of the disease are related to the subtle symptoms, the increasing pace of disability and the long period of time over which patients will require special care. Recent research efforts have been dedicated to the development of computational tools that can be integrated into the workflow of doctors as a complement to support early diagnosis and targeted treatments. This dissertation aims to study the application of Deep Learning (DL) techniques for the automated classification of AD. The study focuses on the role of PET neuroimaging as a biomarker of neurodegenerative diseases, namely in classifying healthy versus AD patients. PET images of the cerebral metabolism of glucose with fluorine 18 (^{18}F) fluorodeoxyglucose (^{18}F FGD) were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database. The pre-processed dataset is used to train two Convolutional Neural Networks (CNNs). The first CNN architecture aims to explore transfer learning as a promising solution to the data challenge by using a 2D Inception V3 model, from Google, previously trained on a large dataset. This approach requires a preprocessing step in which the PET volumetric data is converted into a two-dimensional input image which is the input to the pre-trained model. The second approach involves a custom 3D-CNN to take advantage of spatial patterns on the full PET volumes by using 3D filters and 3D pooling layers. The comparative study highlights the performance and robustness of these two models in dealing with the limited availability of the labelled data. The performance of the estimators is evaluated through a cross-validation procedure, giving a score of 83.62% for the 2D-CNN and 86.80% for the 3D-CNN. The results achieved contribute to the understanding of the effectiveness of these methods in the diagnosis of AD. Given the expected margin for improvements, they can be considered promising and in line with the current state of the art.

Contents

Contents i

List of Figures iii

List of Tables vii

1 Introduction 1
- 1.1 Motivation . . . 2
- 1.2 Objectives . . . 2
- 1.3 Dissertation Outline . . . 3

2 State-of-the-Art 5
- 2.1 Neurodegeneration and Alzheimer's Disease . . . 5
- 2.2 Brain Imaging in the Diagnosis of AD . . . 7
  - 2.2.1 Magnetic Resonance Imaging (MRI) . . . 8
  - 2.2.2 Positron Emission Tomography (PET) . . . 10
- 2.3 Literature Review on Deep Learning for AD . . . 11
  - 2.3.1 CNNs in Medical Imaging . . . 11
  - 2.3.2 CNNs for Classification and Diagnosis of AD . . . 13

3 Materials and Methods 17
- 3.1 Work Context . . . 17
- 3.2 Neural Network Architectures and Deep Learning . . . 19
  - 3.2.1 Components of an Artificial Neural Network . . . 20
  - 3.2.2 Loss Function . . . 21
  - 3.2.3 Activation Function . . . 21
  - 3.2.4 Optimization Algorithm . . . 24
- 3.3 Two-Dimensional CNNs . . . 25
  - 3.3.1 The Architecture of CNNs . . . 25
  - 3.3.2 An Example of a CNN for Binary Classification . . . 28
  - 3.3.3 Application of 2D CNNs for Volumetric Data . . . 33
- 3.4 Three-Dimensional CNNs . . . 34
  - 3.4.1 3D Patch-level CNN . . . 35
  - 3.4.2 ROI-based CNN . . . 35
  - 3.4.3 3D Subject-level CNN . . . 36

4 Experiments and Results **37**
4.1 Dataset Overview 37
4.2 Transfer Learning Approach (2D Model) 38
4.2.1 Data Pre-Processing 38
4.2.2 2D Slice-level Model 39
4.2.3 Train-CV-Test Results 46
4.2.4 Dropout Analysis 48
4.3 Custom Architecture Approach (3D Model) 50
4.3.1 Data Pre-Processing 50
4.3.2 3D Subject-level Model 51
4.3.3 Train-CV-Test Results 56
4.3.4 Dropout Analysis 58

5 Conclusions **61**

References **63**

List of Figures

2.1 Illustration of a healthy brain (left) vs. an Alzheimer's disease brain (right) . 6
2.2 Patient positioned for a MR study of the brain 9
2.3 MRI scans showing: Left image - the increase in ventricular size of a patient with AD (bottom) compared with a healthy subject (top) using axial and sagittal projections; Right image - the volume loss of grey matter associated with hippocampal atrophy using coronal and sagittal projections 9
2.4 Single-modality vs. multi-modality (left) and imaging modalities (right) . . . 13
2.5 Prevalence of each deep model used in AD detection from neuroimaging data 14

3.1 Dropout application example 18
3.2 Artificial intelligence vs. machine learning vs. deep learning 19
3.3 Basic elements of a deep neural network 20
3.4 ReLU Function 22
3.5 Sigmoid Function 22
3.6 Tanh Function 22
3.7 Best hidden layer activation function for each structure 22
3.8 Best output layer activation function for each type of problem 23
3.9 Local optima vs. global optimum 24
3.10 Structure of a CNN, consisting of convolutional, pooling, and fully-connected layers 26
3.11 Detecting patterns and features through a filter application in a convolutional layer 26
3.12 Downsampling by taking the maximum value on a given patch of data through a 2x2 max-pooling layer 27
3.13 Fully connected layer placement on a basic CNN architecture 27
3.14 Kaggle's dogs vs. cats dataset - Cat pictures 28
3.15 Kaggle's dogs vs. cats dataset - Dog pictures 29
3.16 Schematics of the CNN model used during the Kaggle's dataset exercise . . . 29
3.17 Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (no dataset augmentation) 30
3.18 Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (offline dataset augmentation) 31
3.19 Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (online dataset augmentation) 32
3.20 Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (offline + online dataset augmentation) 33

3.21 2D slice-level approach of 3D volume processing 33
3.22 Prevalence of each approach to 3D input data management 34
3.23 3D patch-level approach to dealing with volumetric input data 35
3.24 Region of interest selection in a ROI-based approach to dealing with volumetric input data 36
3.25 3D subject-level approach to dealing with volumetric input data 36

4.1 Dataset's class distribution 37
4.2 Dataset's gender distribution 37
4.3 Dataset's age distribution 38
4.4 Collage of 2D slices extracted from volumetric PET scans 39
4.5 Cross-validation method 40
4.6 Inception V3 architecture and corresponding modules 41
4.7 Number of parameters per Inception V3's sub-networks 42
4.8 LeNet5 architecture 45
4.9 2D binary classification - Network model comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; pre-trained ImageNet weights initialization, with the exception of LeNet5) 46
4.10 2D binary classification - Inception V3 sub-networks comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; pre-trained ImageNet weights initialization) 47
4.11 2D binary classification - Optimizer comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization) . . . 47
4.12 2D binary classification - Weights initialization impact (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; SGD optimizer) 48
4.13 2D binary classification - Dropout rates comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer) 49
4.14 2D binary classification - Training accuracy per epoch (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer) 49
4.15 2D binary classification - Loss per epoch during training (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer) 50
4.16 3D binary classification - 3D pre-processing terminal output example 51
4.17 3D binary classification - 3D data preparation terminal output example . . . 52
4.18 3D binary classification - Base custom 3D-CNN architecture 54
4.19 3D binary classification - Custom variations comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; 50% dropout rate; batch size = 2) 56

4.20 3D binary classification - Learning rate comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; custom 4 architecture; 50% dropout rate; batch size = 2) 57
4.21 3D binary classification - Resolution comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; 0.001 learning rate; custom 4 architecture; 50% dropout rate; batch size = 2) 57
4.22 3D binary classification - Batch size comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; 50% dropout rate) 58
4.23 3D binary classification - Dropout rates comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2) 58
4.24 3D binary classification - Training accuracy per epoch (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2) 59
4.25 3D binary classification - Loss per epoch during training (Dropout trial - Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2) 59

List of Tables

2.1 CNN applications in medical imaging 12
2.2 CNN applications in medical imaging (neurological system) 13
2.3 CNNs for AD classification 15

4.1 2D Binary Classification - Balanced Dataset 43
4.2 2D Binary Classification - Imbalanced Dataset 44
4.3 2D Binary Classification - Imbalanced Dataset (LeNet5 & ResNet50) 45
4.4 3D Binary Classification - Learning Rate Comparison 53
4.5 3D Binary Classification - Resolution Comparison 53
4.6 3D Binary Classification - Custom Architecture Comparison 55

Chapter 1

Introduction

Neurodegenerative disease is the term used for a range of incurable and debilitating conditions affecting the neurons of the human's nervous system, being strongly linked to age. Amongst these conditions, Alzheimer's Disease (AD) is responsible for the greatest burden both for the number of people affected, over 7 million people in Europe and an estimated 47 million worldwide, and for the high costs in medical care [1]. The challenges of AD are due to its subtle onset, the ever-increasing pace of disability and the long period of time over which patients will require special care. Living longer means that more people may be affected in coming decades, resulting in increased efforts for both doctors, caregivers and families.

In order to support AD specialists throughout the diagnosis process, several studies have been performed aiming to significantly increase the chances for an earlier detection of the disease, so that it would be possible to promptly tackle it. Over the last years, Machine Learning (ML) has made important contributions in this line of investigation. It provides an effective framework for the automatic diagnosis of brain disorders through the use of computational algorithms able to adapt to a given dataset and correlate it to its corresponding medical attributes.

The study of Artificial Neural Networks (ANNs) originates from the ambition to build computational models based on the human brain's anatomy and behaviour. They comprise a high number of interconnected computational nodes (referred to as neurons), working in a distributed way to collectively self-optimise through learning. Convolutional Neural Networks (CNNs) are a subcategory of ANNs specially effective when dealing with images or video datasets. The recent successes of convolutional networks have made them the prevalent architecture for dealing with classification, detection, and segmentation tasks in various application areas of medical imaging.

Currently, some of the most popular applications of deep CNNs, besides medical imaging, include time series predictions (e.g., weather forecasting, traffic flow) using 1-dimensional CNN implementations [2, 3], object detection using image recognition [4, 5, 2], biometric identification through print, facial or writing recognition [6, 5, 2], handwriting character recognition [6, 5], and style transferring which involves learning from a specific artistic style and trying to reassemble another image in order to fit that same style [6]. A whole new realm of possibilities is opening up to these investigations with the ever growing ability to gather the labelled data required to create robust high-quality datasets. At the same time, looking at the advances made in terms of calculation capacity, it seems to be only a matter of time before these technologies enter the routine of medical care.

1.1 Motivation

Nowadays, there is a broad consensus that AD appears decades before its first manifestation, advances to the prodromal phase in which the patient starts to experience the early symptoms and evolves to the terminal stage with the progressive loss of basic cognitive skills. Apart from the search for a cure for AD, the most recent efforts are aimed at developing new computational tools that can be integrated into the workflow of doctors as an useful complement to support a decision. In this context, two developments are contributing to meet the clinical need of early detection and treatment monitoring. First, there have been, over the last few years, permanent advances in neuroimaging modalities and AD bio-markers [7]. There are a wide variety of neuroimaging modalities associated to the AD diagnosis, being Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET) the most common ones.

PET is a modality that begins to play a key role in the current realm of AD diagnosis methods. In particular, fluorodeoxyglucose (FDG)-PET is a procedure showing a huge potential to assess the risk or presence of the disease in a very early stage. PET images of the cerebral metabolism of glucose with fluorine 18 (^{18}F) fluorodeoxyglucose (FDG) provide representations of neuronal activity closely linked to the initial manifestations of AD [8]. This is the essential aspect to give medical professionals the possibility of an early diagnosis, when it is more effective [9]. Preferably, the treatment should be initiated before any sort of major mental damage has taken place [10].

The second development to highlight is related to the various successes of deep learning techniques in multiple medical imaging problems [11]. These results show the potential of these techniques to transform preventive healthcare and computerized diagnosis. The rise of DL has prompted the search for solutions to improve the AD diagnosis based on neuroimaging data. Accordingly, several studies have highlighted the importance of DL-based diagnostic systems using either MRI or PET scans [12, 13], while others address the integration of multimodality information, such as FDG-PET and T1-weighted MRI images [14, 15, 16]. One of the main challenges to be faced when applying DL for AD classification is the reduced number of annotated samples available in public datasets, mainly when it comes to ^{18}F-FDG PET scans.

1.2 Objectives

This dissertation aims to study the applicability of deep learning techniques in the context of AD diagnosis with ^{18}F-FDG PET images. The proposed study focuses on how to leverage deep convolutional architectures for classifying healthy versus AD patients, even with a limited dataset collected from the Alzheimer's Disease Neuroimaging Initiative (ADNI). In line with this, the main objectives to be achieved are the following:

- `Volumetric CNNs`: To study different CNN-based techniques able to process 3D PET image data.

- `Transfer Learning vs. Custom Deep Learning`: To provide a comparative study between the usage of transfer learning versus training a network from scratch for binary classification of Alzheimer's disease.

- `Techniques to Prevent Overfitting`: To evaluate the impact of different techniques, such as cross-validation, dropout and transfer learning, when dealing with overfitting due to a limited amount of labeled data available for training.

For this purpose, the intention is to understand the role of deep architectures for handling PET data, by comparing two CNN models in terms of predictive performance. The first CNN architecture explores transfer learning as a promising solution to the data challenge using a pre-trained model. The second architecture involves a custom developed 3D-CNN to take advantage of spatial patterns on the full PET volumes by using 3D filters and 3D pooling layers. At the same time, the study addresses the role of different techniques to prevent the occurrence of overfitting and its impact in terms of predictive performance.

1.3 Dissertation Outline

The following chapters will work as the framework for this dissertation:

- Chapter 2 - `State-of-the-Art` - is dedicated to the analysis of the current advancements related to the research fields pertinent to this dissertation's development, such as neurodegenerative diseases, medical imaging modalities, and literature review on deep learning and its possible applications for AD diagnosis.
- Chapter 3 - `Materials and Methods` - provides the work context, overviews the deep learning techniques and its principles, especially regarding two dimensional and three dimensional CNNs.
- Chapter 4 - `Experiments and Results` - is the core of this dissertation. It includes an analysis of the neuroimaging dataset and covers all of the results obtained throughout the main study, describing every adjustment made and its corresponding performance impact.
- Chapter 5 - `Conclusions` - presents final remarks, key takeaways, and directions for future work.

Chapter 2

State-of-the-Art

This chapter provides an overview of the topics and concepts related to the work to be carried out within the scope of this dissertation, including a literature review on deep learning techniques for diagnosis of Alzheimer's disease. Section 2.1 briefly describes aspects related to neurodegenerative diseases and, in particular, the challenges posed by AD. Section 2.2 is dedicated to describing medical imaging technologies commonly used in the diagnosis of AD. Section 2.3 reviews the literature of some related works in order to provide a better understanding of current achievements, limitations and opportunities associated with the application of deep learning techniques in the diagnosis of AD. The focus is placed on the literature demonstrating the added-value of FDG-PET imaging.

2.1 Neurodegeneration and Alzheimer's Disease

Neurodegenerative diseases are a spectrum of medical disorders that are characterized by neurodegeneration, a biological process that causes a progressive loss of neurological function and structure. In practice, neurodegenerative diseases arise for unknown reasons and they reach a point of irreversible and progressive degeneration and/or death of nerve cells [17]. Despite the origins of these neural anomalies still being far from completely understood, there is a considerable increase in the incidence rate with the subject's age. Over the past few years, there has been a significant effort aimed at tackling the challenges of neurodegenerative diseases, namely a better understanding of the disease, novel tools for early detection, and improvements in patient care.

Among the hundreds of different neurodegenerative disorders, only a handful have gained the attention of the international community due to their highly disabling effects and associated costs, including Alzheimer's disease, Parkinson's disease (PD), Huntington's disease (HD), and amyotrophic lateral sclerosis (ALS). Each brain disorder gives rise to different manifestations according to the type of neurons affected, from difficulties with walking, balance and coordination in PD [18] to loss of cognitive functions in dementia [19]. In this context, Alzheimer's disease is documented as the most common cause of dementia worldwide (responsible for 60 to 80% of cases), affecting roughly 30% of people over the age of 85[20]. Dementia is the term used to refer to a set of symptoms marked by decline in memory, reasoning or other cognitive functions.

According to the EU Joint Programme for Neurodegenerative Diseases [1], dementia cases currently stand as one of neurodegenerative diseases' greatest burden, representing annual healthcare costs of approximately 130 billion euros. It is estimated that Alzheimer's disease affects over 7 million people in Europe alone (worldwide the estimate reaches 47 million people). The challenge facing the European society is even greater because the disease is strongly linked to age. Currently, 15% of the population is over 65, being expected to reach 25% by 2030. In the same line of thought, estimates from the Alzheimer's Association and the World Health Organization (WHO) [21] suggest that there are currently over 35 million people in the world with AD and that this number might increase to more than 100 million people by the year 2050. AD is particularly expensive due to its subtle onset, the gradual levels of debilitation and the average duration between 2 and 10 years over which the condition can extend.

As previously mentioned, the manifestation of AD is generally greater in the elderly population, but there are some rare cases in which the disease has family roots. Between 5% to 10% of all documented AD cases, the patient is subject to specific genetically dominant traits that lead to the inheritance of the disorder and, consequently, to the early development of symptoms, sometimes as early as the age of 50. It is also worth noting that AD cases usually have higher rates in women than in men [22]. In any case, dementia is the first symptom to manifest itself, followed by mood and behavioral deviations, and lastly, severe memory loss, disorientation and aphasia, the inability to understand or produce speech.

Existing treatments for neurodegenerative diseases are limited, addressing the symptoms rather than the cause [23]. Consequently, there is a generalized agreement that early and differential diagnosis of dementia are the key factors to promptly provide patients with the appropriate treatment [24]. Technological advances in recent years have contributed to enhance clinical diagnosis, rather than excluding other causes of cognitive deterioration. These developments include both improvements in neuroimaging techniques and analysis methods. The innovations with respect to neuroimaging technologies are reflected in better image quality and both higher spatial and temporal resolutions. These advances allow the quantitative assessment of the brain's morphology, perfusion (i.e., fluid passage through the circulatory system), metabolism and function [25]. Figure 2.1 illustrates the physical changes between the brain of an healthy individual and the brain of an individual with Alzheimer's disease.

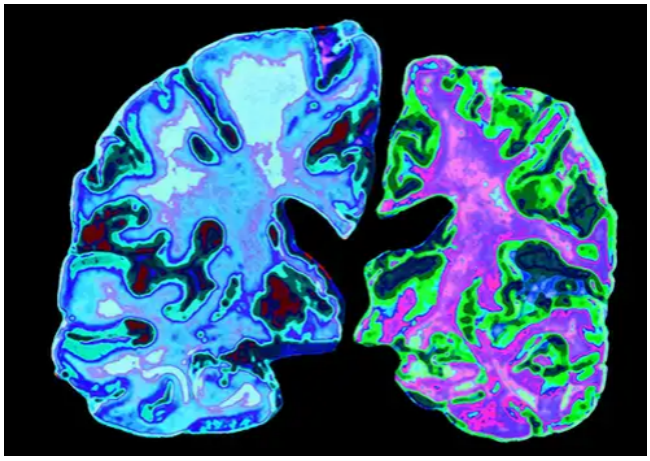


Figure 2.1: Illustration of a healthy brain (left) vs. an Alzheimer's disease brain (right) [26]

In addition to the chemical changes that may occur, the brain of an Alzheimer's disease patient is considerably smaller since the brain shrinks down significantly. Furthermore, when looking at a healthy patient, their brain folds are very full and closely packed together. In contrast, an Alzheimer's disease brain presents much less depth in its folds and a much wider separation between them. Anatomically speaking, the cerebral structures most affected by AD are the hippocampus and the cerebral cortex [27]. The former is an intricate brain structure, situated in the temporal lobe, mainly responsible for learning and memory related functions. The latter is a layer of neural tissue associated with the cerebrum and has various different sensory related functions as well as controlling some other tasks relating to speech, thinking and memory.

Machine learning is another example of an advanced technology that has had a major impact in the diagnosis of dementia [28]. Instead of a subjective visual interpretation, the integration of these methods in computer-aided diagnosis systems aims to provide a quantitative, reproducible and unbiased evaluation (i.e., unaffected by expert knowledge). A reference study published by Kloppel et al. in Brain [29] showed the role of computerized diagnostic methods to provide a more standardised level of diagnostic accuracy for MRI scan-based dementia diagnosis. This study compared the results achieved using support vector machines (SVMs) to those obtained by radiologists when discriminating sporadic Alzheimer's disease from normal ageing and from fronto-temporal lobar degeneration (FTLD). On average, the results indicated that automated computer-based diagnosis were equal to, or superior to, that achieved by six radiologists with different levels of experience, making them a useful support tool to help reach an appropriate decision.

More recently, deep learning models appear to be particularly effective for classifying brain images [30], such as those acquired from structural MRI, functional MRI, and PET scans. The main advantage of deep learning models is that they do not need the complex process of feature extraction required over traditional machine learning. Deep neural networks allow training a complete target system represented by a single model, bypassing the traditional pipeline design, through a concept commonly called end-to-end learning [31]. Nevertheless, training deep neural networks is a challenge since algorithms require extensive supervision, exhaustive manual labelling of data and considerable computational resources. In this context, the combined efforts from the academia and industry have been important to convert the amount of available data into useful datasets for research purposes. One of the most important initiatives is the Alzheimer's Disease Neuroimaging Initiative (ADNI) whose repository includes MRI, PET, genetics, cognitive test, cerebrospinal fluid (CSF) and blood biomarker data [32].

2.2 Brain Imaging in the Diagnosis of AD

Medical imaging is a technique designed to generate visual representations of the internal anatomy and function of some organs and tissues of the patient. This allows the specialist to complete a thorough examination without needing to perform any type of invasive or potentially dangerous procedure. Modern medicine has at its disposal a set of imaging modalities whose benefits differ according to the specific region of interest (e.g., brain, heart, lungs, etc). The comparison between modalities takes, normally, into account two fundamental aspects: the image quality in terms of spatial resolution and contrast, and the effects of ionizing radiation on the patient's body (depending on the energy of the radiated particles).

The use of imaging for diagnosis is associated with some form of electromagnetic radiation such as, for example, visible light in endoscopy, X-rays in mammography and computed tomography (CT), radio waves in magnetic resonance, and gamma rays in nuclear medicine. Every form of energy used in these previously mentioned procedures requires not only the ability to penetrate tissues, but also the ability to interact with those same tissues in order to be able to create some type of visual reproduction of the desired internal structure. In nuclear medicine, a radioactive substance is administered to the subject (injected or ingested) which will originate physiological reactions recorded from within the body. A trade-off must be achieved such that the amount of energy used in the acquisition process provides high image quality to reach a clear verdict, but without jeopardizing the patient's safety.

The current diagnosis of AD relies largely on neuropsychological tests and neuroimaging biomarkers [33]. Ideally, the test must comply with a set of criteria such as being non-invasive, reproducible, and inexpensive. A promising approach is the use of biochemical markers that are present in the cerebrospinal fluid [34]. CSF biomarkers that come closest to fulfilling the above requirements are β-amyloid protein of 42 amino acids (Aβ1-42), total tau protein (T-tau) and hyper-phosphorylated tau (P-tau) in CSF [35]. However, assessment of these biomarkers still means obtaining CSF through a lumbar puncture, which is an unpleasant, invasive, and time-consuming procedure.

Recently, the diagnostics can be performed in an early stage, even in the prodromal stage of the disease also referred to as mild cognitive impairment (MCI), in a non-invasive and reproducible way. The biomarkers for early AD diagnosis that are currently in use reflect the deposition of amyloid (CSF Aβ1-42 or PET with amyloid ligands), formation of neurofibrillary tangles (CSF P-tau), neuronal degeneration (CSF T-tau), changes in brain metabolism (FDG-PET), as well as neuronal loss and volumetric changes in brain structures that cause the disease's symptoms, such as the hippocampus through magnetic resonance imaging of the brain [36].

Next is a brief description of the two most used imaging techniques in the diagnosis of AD, now widely available. On the one hand, magnetic resonance imaging can answer the most relevant questions related to morphology and physiology of the disease. The introduction of high-field MR scanners (3T) into clinical practice has opened new possibilities in brain imaging, allowing to detect small vessels, small brain structures, brain fibres and lesions that measure only a few millimetres in size [37]. On the other hand, a PET scan captures the activity of the brain after a radioactive "tracer" is absorbed into the blood stream [38].

2.2.1 Magnetic Resonance Imaging (MRI)

Magnetic resonance imaging is a non-invasive technique used for disease diagnosis by producing detailed 3D pictures of the anatomy and the physiology of the body [39]. MRI scanners (Figure 2.2) produce a magnetic field used to force the alignment of the atomic nuclei (usually hydrogen protons) within the body tissues with that same field. Then, a radio signal is used to disturb the axis of rotation of these protons against the magnetic field. The MRI sensors detect the amount of energy released as the protons realign with the magnetic field whenever the radiofrequency signal is turned off. This signal is processed to form an image of the body reflecting the density of the atomic nuclei in a specific region. The speed at which protons realign with the magnetic field allows to obtain the contrast among tissues. Unlike CT and PET-scans, MRI does not involve the use of ionizing radiation, although the body is exposed to powerful magnetic fields and fluctuating radio signals.

Figure 2.2: Patient positioned for a MR study of the brain [40]

The diagnostic criteria for patients with AD include loss of brain volume on anatomical MRI. The image on the left of Figure 2.3 illustrates the neuronal loss observed in patients with AD when compared with healthy subjects by measuring the volume of the grey matter (neurons) and the white matter (axons). Usually, the volume loss is more significant in specific brain regions, such as the hippocampus [41]. This brain structure is associated with memory functions and it can be affected even in early stages of the disease. According to Wang et al., 2006 [42], the reduction of the hippocampal volume can predict conversion from prodromal stages to AD with about 80% accuracy. The image on the right side compares the hippocampi of a patient with that of a healthy patient, showing a volume loss associated with atrophy in AD. The cortical thickness measurement of the entire brain mantle is an alternative volumetric method of interest for AD. Lerch et al., 2008 [43] have demonstrated an accuracy of more than 90% in distinguishing AD patients from healthy controls.

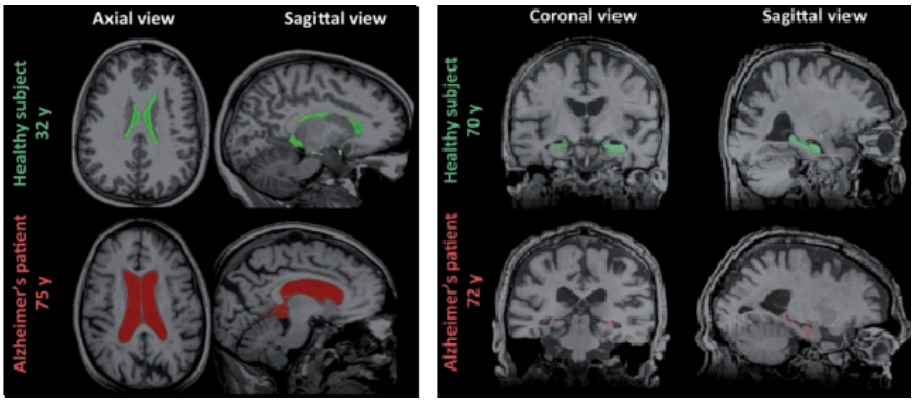


Figure 2.3: MRI scans showing: Left image - the increase in ventricular size of a patient with AD (bottom) compared with a healthy subject (top) using axial and sagittal projections; Right image - the volume loss of grey matter associated with hippocampal atrophy using coronal and sagittal projections. Adapted from [44]

2.2.2 Positron Emission Tomography (PET)

Positron Emission Tomography is a medical imaging modality that consists of conducting measurements of physiological functions, offering quantitative analyses which in its turn provides the ability to monitor a disease's progression over time. These quantitative analyses are generated by examining the data obtained from blood flow, metabolism, neurotransmitters or radiolabeled drugs. The speciality of nuclear medicine involves the application of a radioactive substance, called a radioactive tracer, into the body and the posterior observation of the emitted radiation in the organ or tissue being examined. In particular, the PET system detects gamma radiation emitted through a reaction by a positron-emitting radioactive isotope (e.g., oxygen-18, fluorine-18, carbon-11), which is introduced into the body on a carrier biomolecule. This biomolecule corresponds to a chemical substance commonly used by the organ or tissue during its operating process.

Fluorine-18 is one of the most commonly used tracers in positron emission tomography. This radioactive tracer is attached to compounds like glucose as is the case with 2-18F-fluoro-2-deoxyglucose (^{18}F-FDG) for the measurement of brain metabolism [45]. Medical imaging with ^{18}F-FDG allows the diagnostic of several neurological dysfunctions, including dementia, epilepsy, and movement disorders [46, 47, 48]. The quality of FDG-PET raw images depends on a set of factors such as the patient preparation, the correct acquisition, and the reconstruction parameters [49]. Patients should be fasting and they are required to stay in quiet conditions prior (a few minutes) and after the injection (30-45 minutes).

Next, the patient is carefully positioned in the scanner and the acquisition of images begins, which can last between 10 to 30 minutes. Afterwards, a computer analyses the gamma rays and uses the information to create a three-dimensional image of the tracer concentrations, representing the metabolic activity of a particular organ or tissue. The final volumetric image is compiled from the frames acquired through an iterative technique which combines information about the geometric features of the scanner and about the dispersion of the radioactive isotope. An attenuation correction is applied to consider properties of the tissues through which the photon passes. Once reconstructed, the PET images can also be co-registered with CT or MR for anatomical localisation and structural comparisons.

There are several commercial software packages available to help medical specialists quantify the information related to neurodegenerative disorders, like CortexID (General Electrics), Scenium (Siemens), BRASS (Hermes), Vista (MIM), among others. These software packages provide a co-registration step to standard anatomical templates allowing statistical comparisons against normal patients, being performed using regions of interest (ROIs) or in voxel-by-voxel basis.

Decreased brain glucose consumption, known as hypometabolism, is seen as one of the earliest signs of neural degeneration, being associated with AD progression [50]. However, there is a substantial overlap between the regions of the brain where ^{18}F-FDG PET scans show hypometabolism, making the task of recognising patterns and regularities by nuclear medicine specialists difficult. Patients with both MCI and AD show hypometabolism in the regions of the posterior cingulate and the parietotemporal cortices, with variable frontal lobe involvement (more frequent in advanced stages). As opposed to this, glucose consumption is usually preserved in the primary visual cortices, striatum, thalamus, and primary sensorimotor cortices. Additionally, the asymmetric involvement of these areas is a classical pattern of AD in FDG-PET, being an important aspect in a visual evaluation [51].

FDG-PET represents a valuable and unique tool able to estimate local cerebral rate of glucose consumption. Thereby, PET may point out biochemical changes that underlie the onset of a disease before anatomical changes can be detected by other modalities such as CT or MRI. The presence of hypometabolism patterns precede the typical pattern of brain atrophy estimated from MRI. According to some authors [52], the added value of FDG-PET, over other AD biomarkers, is more associated with the follow-up of the disease's progression than its diagnosis. However, PET scans have a disadvantage which is its cost of operation. Single-photon emission computed tomography (SPECT) is a less expensive imaging process with similarities in what concerns the use of gamma rays and radioligands.

Modern PET scanners are combined with CT or MRI to provide, in the same machine and session, both anatomic and metabolic information. Simultaneous PET-MRI scanner technology allows a better understanding of the brain function/dysfunction as reported in [53, 54]. MRI represents a gold-standard imaging modality for numerous indications, while a great number of specific PET tracers are available today to assess functional and molecular processes in the brain.

2.3 Literature Review on Deep Learning for AD

This section provides an overview of current knowledge and achievements on the application of ML techniques for medical care. It includes a description of the most common deep learning models for medical imaging, mainly those case studies which are related to the diagnosis of AD rooted on convolutional neural networks. At the same time, the main aspects of the field will be summarised in order to put into perspective the directions to explore in upcoming chapters, as well as the works published to date who laid the groundwork for this dissertation work.

2.3.1 CNNs in Medical Imaging

There is an increasing number of examples showing the effectiveness of machine learning (ML) methods for medical care, even if their deployment in real-world environments may require some more time. The special issue published in the IEEE Transactions on Medical Imaging by Greenspan and colleagues [55] drew attention to the impact of DL techniques in the domain of medical imaging. Meantime, the surveys by Hu et al., (2018) [56] and Litjens et al. (2017) [11] contribute to a clear understanding of the principles and methods of artificial neural networks and deep learning, as well as on how these models are applied in different tasks using a wide variety of image modalities.

Nowadays, convolutional networks has become an important tool when it comes to finding increasingly efficient solutions to various problems. Its popularity can be observed by the rate of growth in scientific publications over the last years. Table 2.1 provides a list of research studies organised by their corresponding category, task, medical imaging modality and the publication pertaining to their respective entries. These studies address the problems of detection, segmentation and classification using different diagnostic images.

Category	Task	Modality	Reference
Abdomen and pelvis	Prostate cancer identification	MRI	Wang et al., 2017 [57]
Abdomen and pelvis	Bladder cancer treatment response assessment	CT	Cha et al., 2017 [58]
Abdomen and pelvis	Liver lesions classification	CT	Yasaka et al., 2018 [59]
Breast	Breast lesions classification	MG/US	Kooi et al., 2017 [60], Han et al., 2017 [61]
Chest	Mediastinal lymph nodes classification	PET/CT	Wang et al., 2017 [62]
Chest	Tuberculosis identification	Rad.	Lakhani et al., 2017 [63], Lopes et al., 2017 [64]
Chest	Lung nodules classification	CT	Wang et al., 2018 [65], Ciompi et al., 2017 [66], Song et al., 2017 [67]
Musculoskeletal system	Fracture identification (wrist/hand/ankle)	Rad.	Olczak et al., 2017 [68]
Musculoskeletal system	Hip osteoarthritis identification	Rad.	Xue et al., 2017 [69]
Musculoskeletal system	Bone age assessment	Rad.	Larson et al., 2018 [70], Lee et al., 2017 [71], Spampinato et al., 2017 [72]
Skin	Skin cancer classification	PHO	Esteva et al., 2017 [73]

Note: CT = Computed Tomography, MG = Mammography
MRI = Magnetic Resonance Imaging, PET = Positron Emission Tomography
PHO = Photography, Rad. = Radiography, US = Ultrasound

Table 2.1: CNN applications in medical imaging

Machine learning and deep learning have also been extensively used in neuroimaging studies to devise diagnostic and classification for a number of diseases and disorders, such as neurodegenerative diseases, strokes, epilepsy, schizophrenia and its prodromal stages, autism, abnormal brain development and aging, among others. Table 2.2 provide several examples of these undertakings, being possible to verify the predominance of MRI-base approaches. It is worth note that the information list in these tables has been compiled based on the readings of the following articles [74, 75, 76, 77]. Litjens et al. [11] provides a more extensive analysis on applications of deep learning in the context of medical imaging.

Task	Modality	Reference
Tissue necrosis after CVA prediction	MRI	Stier et al., 2015 [78]
PD identification	SPECT	Choi et al., 2017 [79]
Brain tumor segmentation	MRI	Havaei et al., 2017 [80]
Brain lesion segmentation	MRI	Kamnitsas et al., 2017 [81]
Brain age prediction	MRI	Cole et al., 2017 [82]
Alcoholism detection	MRI	Wang et al., 2018 [83]

Note: CVA = Cerebrovascular Accident
MRI = Magnetic Resonance Imaging, PD = Parkinson's Disease
SPECT = Single-Photon Emission Computed Tomography

Table 2.2: CNN applications in medical imaging (neurological system)

2.3.2 CNNs for Classification and Diagnosis of AD

A systematic review of the cutting edge in AD classification using deep learning can be found in [84]. Authors emphasize some important aspects for understanding the whole scenario of AD diagnosis. First, they state that approximately 73% of neuroimaging studies were performed using single-modality data, leaving the multi-modality category with 27% of the total cases, which can probably be attributed to the significant increase in complexity. As illustrated in Figure 2.4, around 83% of the studies pertain to MRI, 9% refer to fMRI and the last 8% to PET scans. The difference in the number of studies is understandable considering that MRI scans are the most available out of those three modalities mentioned above.

Recently, there have been important advances in the development of PET radiotracers for AD biomarkers, aiming at the diagnosis in the early stages of the disease. For example, the work of Sala et al. [85] demonstrated that patterns of brain hypometabolism represent relevant markers with highly supportive diagnostic and prognostic role. At the same time, there is a growth trend of multimodal solutions that can be explained by the greater availability of computing resources and the emergence of new hybrid technologies combining MRI and PET. These modern scanners take advantage of the strengths of both modalities to produce highly detailed images of the inside of the brain.

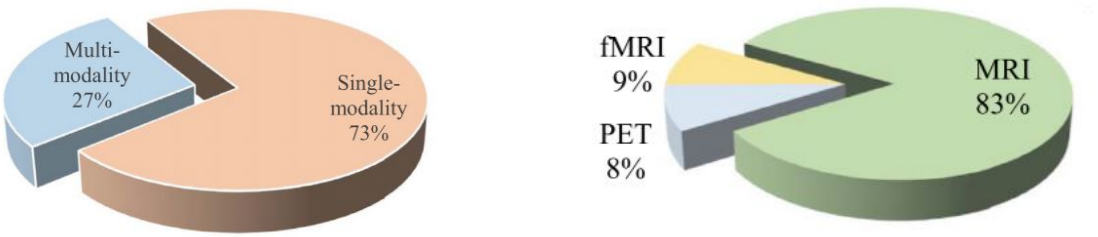


Figure 2.4: Single-modality vs. multi-modality (left) and imaging modalities (right). Taken from [84].

Another aspect to be highlighted is the diversity of deep architectures found in the literature. Nevertheless, it can be said that CNNs are, currently, the state-of-the-art in AD diagnosis. A common approach is to convert the volumetric data into an image to be applied at the input of a 2D convolutional network. Most of the studies transfer the weights from pre-trained networks on the ImageNet database to the target task. This process, known as transfer learning, speeds up training and reduces costs by leveraging previous knowledge. However, dealing with individual slices may discard the depth information. Therefore, many other studies aim to leverage the full information by exploring 3D CNN for learning representations for volumetric data.

As illustrated in Figure 2.5, the use of 2D CNNs is still the most frequent architecture found in the reviewed literature, while its 3D counterpart currently stands at a close second, mainly due to the increasing availability of computational power. The classification process is usually performed in order to assign to a subject one of several classes. A significant part of the reported studies address the binary classification problem, i.e., they consider normal cognitive (NC) against AD. A more challenging task occurs when the discriminative power of deep networks needs to includes early and late stages of mild cognitive impairment. MCI is sometimes subdivided into sMCI (Stable Mild Cognitive Impairment) and pMCI (Progressive Mild Cognitive Impairment) which will eventually develop into AD. Table 2.3 presents some relevant studies performed in this field during the most recent years based on the following works [84, 86, 87, 88].

A third aspect that should be mentioned is the difficulty to compare and/or reproduce results given the heterogeneity of datasets (some of which are not public), pre-processing steps, deep models and performance metrics employed. Although it can be considered that the application of DL techniques in AD diagnosis is still in their initial stages, recent works [12, 29] demonstrate that deep neural networks can outperform radiologist abilities. The coming years may determine the feasibility of these models as a support tool to help clinicians reach an appropriate decision in real clinical environments.

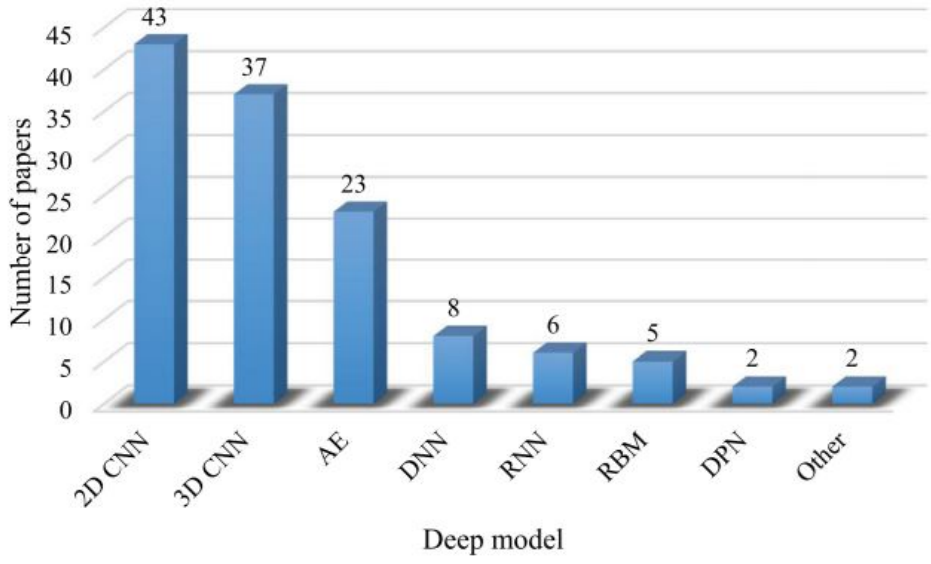


Figure 2.5: Prevalence of each deep model used in AD detection from neuroimaging data [84]

Modality	Classes	Score	Reference
MRI	AD vs CN	98%	Wang et al., 2018 [89]
MRI	AD vs CN	100%	Taqi et al., 2018 [90]
MRI	MCI vs CN	83%*	Qiu et al., 2018 [91]
MRI	Multi-class[1]	57%	Valliani et al., 2017 [92]
MRI	AD vs CN	91%	Liu et al., 2018 [93]
MRI	MCI vs CN	74%*	Li et al., 2018 [94]
MRI	sMCI vs pMCI	74%	Liu et al., 2018 [15]
MRI	sMCI vs pMCI	80%*	Lian et al., 2020 [95]
MRI	AD vs CN	91%	Aderghal et al., 2017 [96]
MRI	AD vs MCI	70%	Aderghal et al., 2017 [96]
MRI	MCI vs CN	66%	Aderghal et al., 2017 [96]
MRI	sMCI vs pMCI	73%	Lin et al., 2018 [97]
MRI	AD vs CN	90%	Bäckström et al., 2018 [98]
MRI	AD vs CN	99%	Asl et al., 2018 [99]
MRI	AD vs MCI	76%	Senanayake et al., 2018 [100]
MRI	AD vs MCI	100%	Asl et al., 2018 [99]
MRI	MCI vs CN	75%	Senanayake et al., 2018 [100]
MRI	MCI vs CN	94%	Asl et al., 2018 [99]
MRI	sMCI vs pMCI	62%	Shmulev et al., 2018 [101]
MRI	Multi-class[1]	95%	Asl et al., 2018 [99]
AV-45 PET	AD vs CN	85%	Punjabi et al., 2019 [87]
AV-45 PET + MRI	AD vs CN	92%	Punjabi et al., 2019 [87]
AV-45 + FDG PET	AD vs CN	96%	Choi et al., 2018 [86]
AV-45 + FDG PET	sMCI vs pMCI	84%	Choi et al., 2018 [86]

Note: CN = Cognitively Normal, AD = Alzheimer's Disease
MCI = Mild Cognitive Impairment, sMCI = Stable MCI, pMCI = Progressive MCI
* = Severely Imbalanced Dataset, [1] = AD vs MCI vs CN

Table 2.3: CNNs for AD classification

Wen et al. [88] provided an overview and reproducibility evaluation concerning the classification of AD using CNNs. Although the study is limited to magnetic resonance data, it helps to understand the limitations and opportunities of different approaches, namely in what concerns the selection of 2D or 3D convolutional models. The reproducibility evaluation is focused on the use of three public datasets: the Alzheimer's Disease Neuroimaging Initiative (ADNI) study, the Australian Imaging, Biomarkers and Lifestyle (AIBL) study and the Open Access Series of Imaging Studies (OASIS). Overall, the different 3D approaches achieved similar performances, while the 2D slice approach was lower. Authors also showed that accuracy scores obtained from severely imbalanced datasets tend to be overly optimistic when compared to those from balanced ones. Furthermore, they noted that some of studies presenting higher performances might have suffered from data leakage which also inflate the corresponding accuracy scores.

Although fewer and more recent, studies involving PET data have shown promising results, particularly in the early stages where the diagnosis is more challenging for clinicians [102]. Choi and colleagues [86] developed an automatic system based on a deep CNN to predict cognitive decline in MCI patients using flurodeoxyglucose (FDG) and florbetapir (AV-45) PET. Authors used images from 139 patients with AD, 171 with MCI and 182 normal subjects obtained from the ADNI database. The CNN was trained using 3-dimensional PET volumes as inputs and it used minimally processed images without spatial normalization. The prediction accuracy of the conversion of mild cognitive impairment to AD was compared with a SVM classifier based on PCA features of FDG and AV-45 PET images. Accuracy of prediction (84.2 %) for conversion to AD in MCI patients outperformed the conventional feature-based quantification approach.

Punjabi et al. [87] also performed a rather interesting study that compared the performance ratings between MRI- and PET-based systems. Initially, authors obtained an 87% accuracy score with a full MRI dataset (missing value from Table 2.3), slightly higher than the 85% obtained with the PET dataset. However, MRI datasets are, generally, larger than the PET ones due to the previously mentioned wider availability of scans. Accordingly, Punjabi and colleagues proceeded to reduce the MRI dataset to the same amount of samples as the ones in the PET dataset. On this second trial, the performance decreased significantly as shown by the 74% accuracy value obtained on the same task. These results are promising in showing the usefulness of PET data, even when compared to those obtained using magnetic resonance.

Another reference study published in Radiology by Ding et al. [12] shows that pre-trained models is a promising strategy to obtain a more standardised level of diagnostic accuracy, even when compared to human experts. However, the most significant result of this study is the fact that it has demonstrated the usefulness of FDG-PET to successfully detect AD about six years before the final medical diagnosis was given [103]. This is a remarkable result as it allows patients to start the required treatments before a substantial manifestation of the disease occurs.

Chapter 3

Materials and Methods

This chapter clarifies the objectives of this dissertation and the methodologies adopted to face the challenges of the proposed study. Section 3.1 provides the context of the work to be carried out, helping to understand its delimiting boundaries. Section 3.2 introduces basic concepts of artificial neural networks and deep learning with relevance for this work. Section 3.3 encompasses the main characteristics of convolutional networks operating on 2D images, as well as how they can be used in problems involving volumetric data. Finally, Section 3.4 presents different strategies previously used to leverage the depth information based on 3D-CNN architectures.

3.1 Work Context

As stated before, deep learning techniques, such as convolutional neural networks, are very data-hungry. In practical terms, DL-based methods require a large amount of training data for generalization, namely for discriminating highly complex patterns such as those occurring in the various stages of AD. However, annotated medical datasets are generally not very extensive, as is the case with a neuroimaging modality like 18F-FDG PET scans. Due to this fact, the occurrence of some level of overfitting will certainly be a reality that will have to be dealt with appropriately. Overfitting occurs when the model adapts excessively to the training samples to the extent that it negatively harms the performance on new data. This could seriously jeopardize the final outcome of the project. Here, the data scarcity problem is considered as the main limiting factor for the AD classification.

Given the limited availability of labelled data, conventional AI techniques, like Support-Vector Machines (SVMs), are frequently used models that could have been considered here. However, this dissertation is solely focused on the applicability of DL-based techniques for the automatic classification of AD using FDG-PET images. More specifically, the main goal is to understand how to leverage deep convolutional architectures for classifying healthy versus AD patients by exploring their representation learning capabilities (instead of careful feature extraction). For that purpose, a comparative study will be carried out centred on two distinct deep models: a pre-trained 2D-CNN model against a custom developed 3D-CNN trained from scratch. Still from a methodological point of view, this study will address the impact of techniques and strategies aiming to diminish the existence of overfitting.

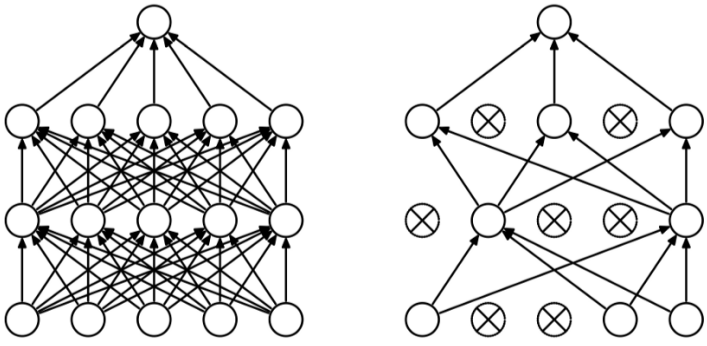
The first CNN architecture aims to explore transfer learning as a promising solution to the data challenge by using a 2D Inception V3 model, from Google, previously trained on a

large dataset. On the one hand, the objective is to explore a pre-trained model adapted to the problem at hand and, then, fine-tuning it based on the target dataset. The idea is to take advantage of the fact that the pre-trained model used a large and general training dataset, while the feature maps learned previously remain relevant and can be re-purposed [104]. On the other hand, this approach requires a pre-processing step in which the PET volumetric data is converted into a two-dimensional image which is the input to the pretrained model. The 2D-CNN deliberately discards the depth information which can result in lower performance. The second approach involves a custom developed 3D-CNN to take advantage of spatial patterns on the full PET volumes by using 3D filters and 3D pooling layers. A custom developed model may help to reduce the complexity both in terms of number of parameters and depth.

Additionally, this study will adopt some other strategies during the development of the two architectures in comparison, including the following:

- `Cross-validation`: The implementation of a k-fold cross-validation procedure would be a logical first option when dealing with a low quantity of data since it allows the network to eventually train on the full dataset (further description on subsection 4.2.2) [105].
- `Dropout`: Implementing dropout layers will make the model neglect a certain quantity of the network's units (see the example in Figure 3.1), based on a chosen probability, which will also lead to a reduction in overfitting. Nevertheless, it will require the model to perform more epochs in order to properly converge [106].
- `Model complexity reduction`: An overly complex model will also contribute to the occurrence of overfitting. A common solution to this problem is the reduction of trainable parameters, while maintaining a solid balance between overfitting and underfitting [106].

Data augmentation is another prominent method to improve the generalization performance when dealing with a significant lack of data. It aims to enrich the diversity of training samples what could be essential in medical classification tasks. However, most data augmentation techniques are hand-crafted and sub-optimal in 3D image processing. The nature of the data used would require a custom data augmentation strategy that was outside the scope of the work. Although they have not been implemented, the use of ensemble methods deserves mention. The motivation behind ensemble models is to combine multiple predictions obtained from different models in order to provide a more robust output decision.



(a) Standard Neural Net (b) After applying dropout.

Figure 3.1: Dropout application example [106]

In what concerns the software and hardware resources, the study employed the Keras deep learning API built on the Tensorflow platform, while the training of the models was carried out on a remote server supported by a NVIDIA GeForce RTX 2080 Ti graphics card. The FDG-PET dataset was obtained from the ADNI platform whose main goals are in line with those of this work, which facilitated the data access request.

3.2 Neural Network Architectures and Deep Learning

Belonging to the artificial intelligence (AI) research field, the end goal of machine learning is the conception and development of autonomous mathematical algorithms that maintain a constant state of self-improvement in order to maximize its accuracy performance when executing a given task. In supervised learning, this evolution is achievable by feeding representative input-output pairs to the algorithm relating to the specified task. This will then allow the machine learning model to "learn" from that same data by analysing certain patterns and features (e.g., glucose levels throughout different areas of the brain) that can possibly establish a direct correlation between the input data and the correct result at the output.

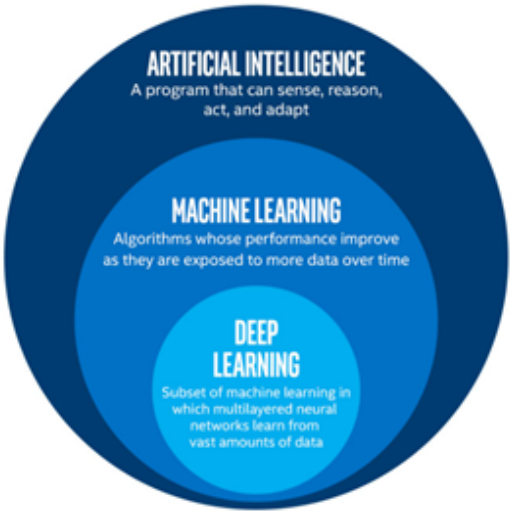


Figure 3.2: Artificial intelligence vs. machine learning vs. deep learning [107]

This preprocessing operation on the data is implemented so that all of the information being handled by the model possesses the same format and general standardization. This dataset is then subdivided into smaller datasets, distributing the information for specified purposes, namely training (generally the largest one), validation and testing.

Maximizing the model's performance will also be directly dependable on the type of task and the data being used. Optimal results can be obtained by changing the type of learning algorithm, loss functions, activation functions, and various hyperparameters associated with the training process. These will then greatly increase the number of possible parameter combinations, which will consequently generate the possibility for a positive progression in the quality of the results.

3.2.1 Components of an Artificial Neural Network

Deep learning, a subsection of the machine learning field, while maintaining the same key objectives as previously mentioned, aims to do so by basing its model's architecture in a structure identical to the ones observed in human biology, more specifically the human brain. These attempts to mimic the human brain's internal framework led to the creation of what is now called artificial neural networks.

Artificial neural networks are composed by numerous node layers, also called neurons or perceptrons. Typically, the main structure of a neural network consists of an input and output layers, where the data will be fed into and subsequently retrieved from, after the training process, and a number of hidden layers in between these two. It is through these hidden layers that the input data will be subjected to varying weighted connections and transfer functions corresponding to each node. The behaviour of a regular neural network and its neurons or nodes can be described by the following equation:

$$f(x) = \sigma(\sum_{i}^{n} x_i \omega_i + b) \tag{3.1}$$

Generally, x stands for the input data being processed, ω is the corresponding connection weight (usually random at first), b is the added bias and σ is the chosen activation function. During the training process, these values will progressively adjust themselves by comparing the output results with the desired ones, gradually increasing the accuracy performance.

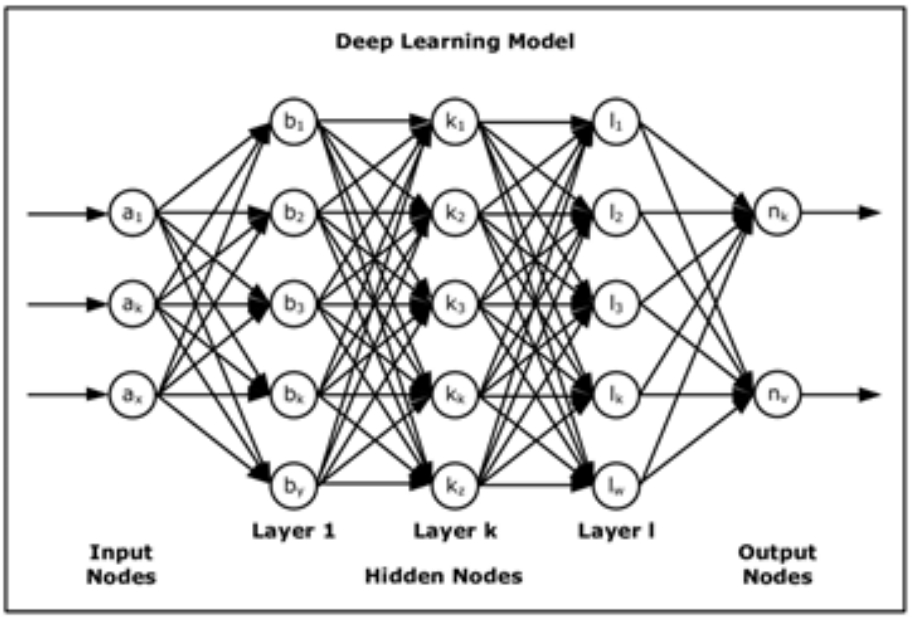


Figure 3.3: Basic elements of a deep neural network [108]

3.2.2 Loss Function

When working with any type of optimization algorithms, there is usually some sort of maximization or minimization being performed on a function somewhere along the process, with that function generally being labeled as objective function (or criterion). With neural networks, that objective function is commonly used for error minimization, being also known by the term: loss function (or cost function). A loss function is a tool that seeks to compile every important parameter that causes a meaningful impact on the algorithm's performance and reduces it down to a scalar value representing the system's error, also known as loss.

The choice for the best loss function to use in a certain problem depends on the type of problem being solved, as well as the corresponding type of activation function and network's output layer structure. In this case, since this project's main task is centred around a binary classification problem, the selected loss function to employ is the binary cross-entropy (or log loss).

3.2.3 Activation Function

Activation functions, also called transfer functions, are responsible for determining how the output value of a neuron connection, on a given network layer, is obtained through the input data and corresponding weights. The main purpose of the activation function may differ depending on where it is applied in the network's structure. An activation function implemented on a hidden layer will be accountable for how adequately the network will learn over time, while on the other hand, if implemented on the output layer, it will establish what type of predictions the model will make. Starting with activation functions for hidden layers, there are three types that are prominently used:

- Rectified Linear Activation (ReLU)
- Logistic (Sigmoid)
- Hyperbolic Tangent (Tanh)

The ReLU activation function is perhaps the most frequently used out of the three alternatives mentioned above and its behaviour can be described by the following expression: $max(0.0, x)$. This means that whatever the input value is, as long as it is a positive number, it will be returned as the output, otherwise the returned value will be 0.

When it comes to the Sigmoid and Tanh activation functions, the approach is actually quite similar in both cases, although differing a little to the previously described ReLU. With the Sigmoid function the received input values will be returned within a 0 to 1 range, with the lower values approaching 0 and the higher values getting closer to 1. Identically, the Tanh activation function follows a similar route but with the output value range being from -1 to 1. The Sigmoid and Tanh function can be respectively calculated by the following expressions:

$$\frac{1.0}{1.0 + e^{-x}}; \frac{e^{x} - e{-x}}{e^{x} + e{-x}} \tag{3.2}$$

It is also important to note that it is recommended to perform some type of normalization process on the input data before starting the training stage in order to achieve the intended results with all of these cases.

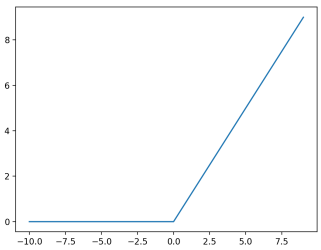


Figure 3.4: ReLU Function

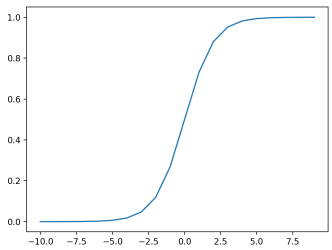


Figure 3.5: Sigmoid Function

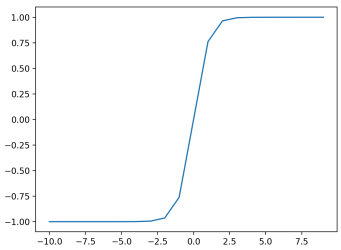


Figure 3.6: Tanh Function

Figures 3.4 to 3.6 represent the "input vs. output" shapes for each one of the three cases examined in this section (taken from [109]). The choice for the type of hidden layer activation function to use, typically the same for all of the hidden layers on that same model, should be selected according to the neural network's structure in order to best suit the model's purpose (Figure 3.7).

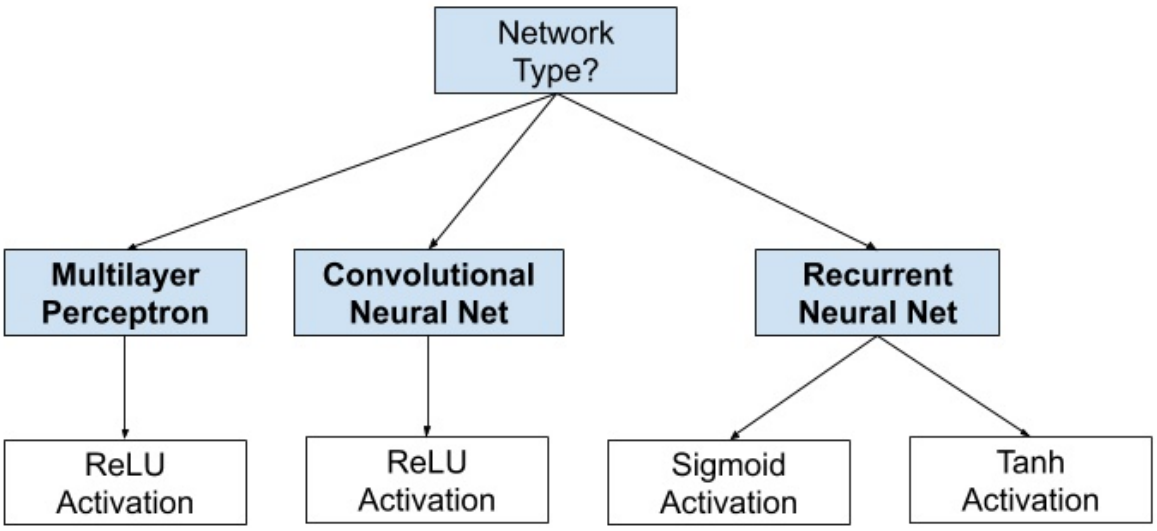


Figure 3.7: Best hidden layer activation function for each structure [109]

By following this guide, it is easily determinable that the best option for this work's hidden layer activation function is the ReLU activation function since the approaches will be using CNN-based techniques. Regarding output layer activation functions, the most commonly used types are the following:

- Linear
- Logistic (Sigmoid)
- Softmax

The linear activation function is essentially just a direct return of the value obtained from the weighted sum of the connections, without transforming the values in any way. Skipping the sigmoid function, as it was already explained during the hidden layer segment, there is the softmax function. The softmax function returns a vector of values at the output corresponding to the respective probabilities for each class belonging to the task's possible predictions and can be calculated as follows:

$$\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.3}$$

where e^{z_i} being the input data vector, e^{z_j} being the output data vector, and K being the number of possible classes in the classifier.

As mentioned previously, the choice for the best output layer activation function to go with should be based on the type of predictions being made with the corresponding model. A helpful visual guide to help choose the best fit for a certain problem can be examined below (Figure 3.8). Once again, based on this information and since the main task will be to perform a binary classification between the two classes of AD and CN, the preferable option for the output layer activation function would be the sigmoid activation function.

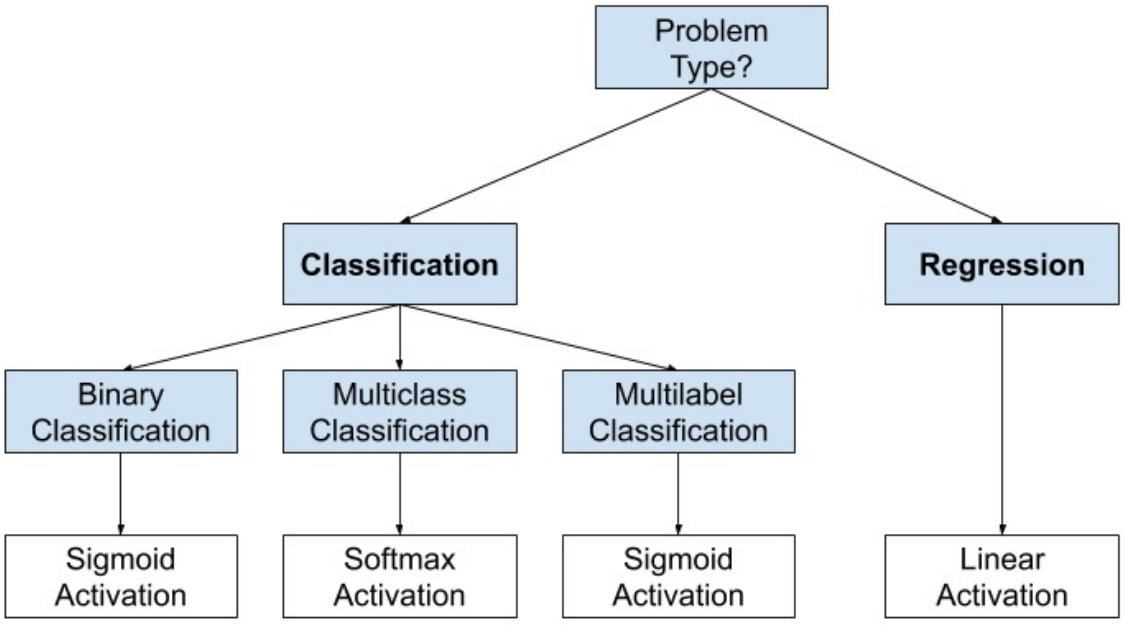


Figure 3.8: Best output layer activation function for each type of problem [109]

3.2.4 Optimization Algorithm

An optimization algorithm, or simply optimizer, is a tool whose main purpose is to select the best suited set of input data in order to maximize/minimize a model's loss function (or objective function). An optimizer can be either differentiable or not differentiable, depending on whether or not a derivative can be determined for any specific input data point. As it is well known, a first-order derivative allows an analysis on the objective function's slope at a certain point. However, when it comes to the derivative of an objective function with multiple input variables, the correct term to use is gradient. Being able to calculate the gradient of an objective function is a major factor when it comes to the optimization process, allowing the implementation of first-order algorithms, which make use of this element in order to utilize it as a sort of guideline to finding the optimal loss. By "following" the decreasing gradient, and choosing an appropriate learning rate value, which defines the "step size" at which the system will move in that search space, a minimum loss value will eventually be found.

This learning rate is yet another hyperparameter whose impact is quite significant when it comes to achieving an optimal model performance. If it is set to too low of a value it will become really time consuming and it might even get stuck on a local optima. On the other hand, if the value is too high, even though that would significantly decrease the amount of time needed, it could miss the global optima due to its larger step size (Figure 3.9).

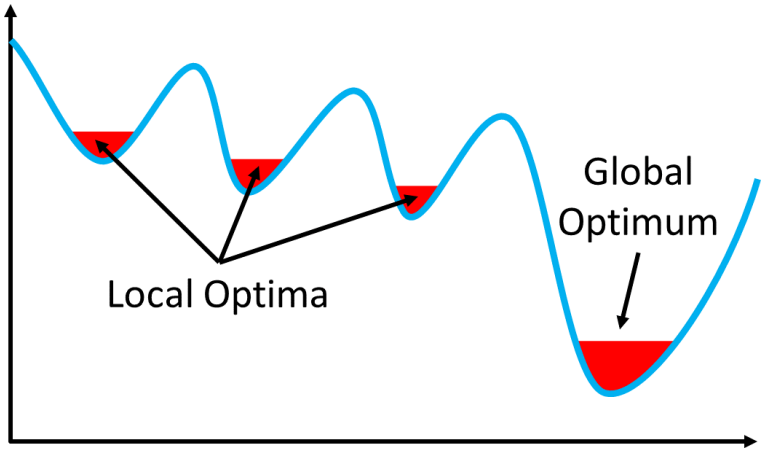


Figure 3.9: Local optima vs. global optimum [110]

Some of the most common first-order algorithms include:

- GD (Gradient Descent): Low computational demands, easy implementation and low complexity but it might get stuck at a local optima and requires a large amount of memory.
- SGD (Stochastic Gradient Descent): Variant of gradient descent. Convergence achieved in less time, with lower memory requirements but it comes with a high variance in the model's parameters and it may not stop after finding global optima.
- Momentum: Solves the variance problem found with SGD while keeping the faster convergence but it comes with one extra hyperparameter that requires a manual and accurate selection.

- Adagrad: Automatically adapts learning rate value during process and is able to train with less data, but has heavy computational demands and the training process is significantly slower due to the continuous learning rate adjustment.

- Adadelta: Extension of Adagrad that looks to solve the decaying learning rate issue. Maintains high computational demands.

- Adam (Adaptive Moment Estimation): Solves decaying learning rate problem and has really fast training and convergence. Also computationally expensive.

Most of these algorithms will be evaluated throughout the work and, depending on the results obtained, the one with the best performance will be selected.

3.3 Two-Dimensional CNNs

Although the final objective of this dissertation will be centered around three-dimensional medical scans, there are some useful employments of 2D-CNNs within the medical imaging realm. The main purpose of this section is to provide a brief introduction into the convolutional models, as well as their associated structure and parameters. The objective was to create a base knowledge on this type of neural network before investing the efforts purely into 3D CNNs. It is also worth restating that this project was implemented by using the Keras deep learning API. Given the simplicity and flexibility of use, this machine learning platform allows to focus on the parts of the problem that really matter.

3.3.1 The Architecture of CNNs

Among neural network models, one of the most prominent variations is the convolutional neural network. A Convolutional Neural Network, or CNN, is a deep learning technique that was inspired by the biological processes that were observed when studying animal brains, more specifically, and as the name suggests, the neurons and their connection structures. This technique is especially relevant when performing image (or video) related recognition, classification, analysis and other similar tasks, since it does not need any manual feature selection and requires a relatively low amount of preprocessing. This technique has been proved to be a very effective system but it is important to note that, when overused, it may lead to an overfitting problem.

A CNN is a neural network that applies convolutional processes in at least one of its layers. When talking about its architecture, a CNN generally consists of an input layer, an output layer, and a variety of different hidden layers in between them. It is in this input layer that the corresponding data is collected in a tensor object, a format where the dimensions are defined by the images' resolution (height and width), the number of images, and the depth of these images (i.e. an RGB image has a depth of three layers due to its Red, Green and Blue levels). The hidden layers can pose various different forms: convolutional layers, pooling layers, fully connected layers, and normalization layers (Figure 3.10).

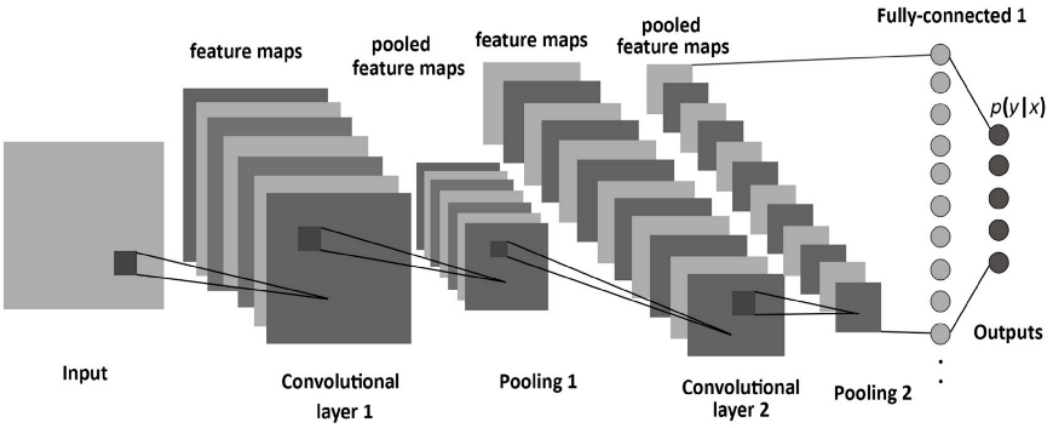


Figure 3.10: A CNN consisting of convolutional, pooling, and fully-connected layers [111]

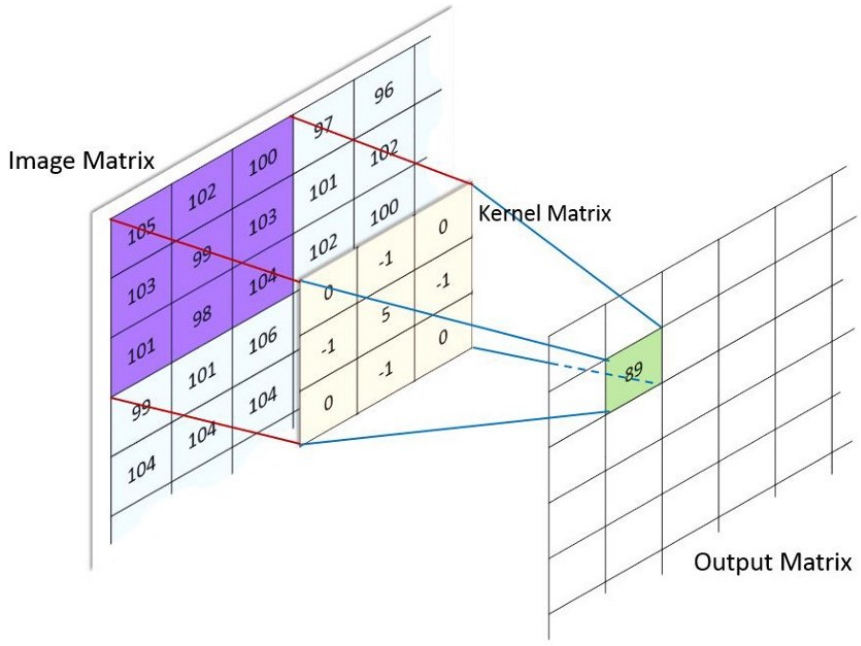


Figure 3.11: Detecting patterns and features through a filter application in a convolutional layer [112]

The first share of hidden layers are also responsible for detecting simple image features, consisting mainly of various different shapes, curves and edges, and progressively establish higher complexity patterns based on those same elements, on the following set of layers. Starting off with the convolutional layers, these are layers that basically apply a specifically sized filter (or kernel) on the input data it receives by preforming a multiplication between both set of values. These are used in order to identify several different basic features present on the input images and from then on evolve into more complex feature combinations, as it was previously mentioned. In this particular case, it is usually visualized as a sliding weighted filter that slowly shifts throughout the full extent of the image (Figure 3.11).

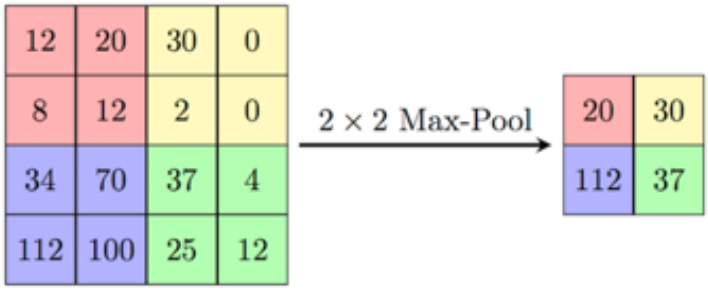


Figure 3.12: Downsampling by taking the maximum value on a given patch of data through a 2x2 max-pooling layer [113]

The pooling layers main purpose is the downsizing of the input image dimensions. This feat is commonly achieved by reducing a specific number of values to the maximum or minimum value of that corresponding selection. These methods are called max-pooling and min-pooling, respectively (Figure 3.12).

The fully connected layers which are simply basic multilayer perceptrons (when each individual neuron found in a particular layer connects to every single neuron found on the following layer), and are usually found more towards the end of the CNN (Figure 3.13). In short, convolutional neural networks aim to develop a deep learning algorithm by attempting to define a consistent correlation between certain features and patterns observed on the input data and its corresponding results at the output. While most of the research and advancements made with CNN-based techniques are related to two-dimensional data, a demand for its three-dimensional counterpart has been steadily growing, raising some difficulties relating to the spatial aspects of the network's operations. As a result, a number of different approaches have been tested in order to achieve the most efficient three-dimensional CNN system, which may vary depending on the type of problem that is being handled in each occasion.

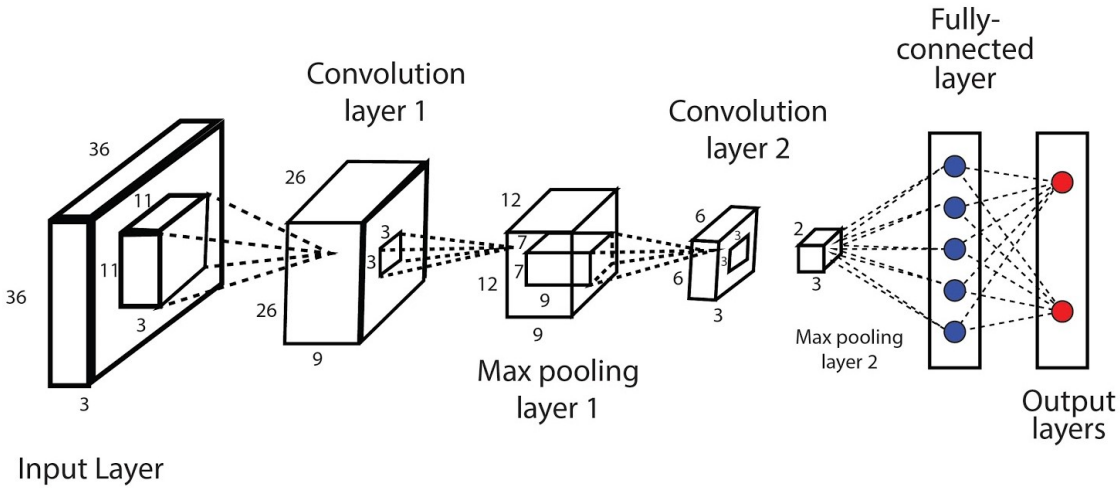


Figure 3.13: Fully connected layer placement on a basic CNN architecture [114]

3.3.2 An Example of a CNN for Binary Classification

As a preliminary study on the use of convolutional networks, it was implemented a deep model for the cats and dogs classification problem based on the Kaggle's dataset titled "Dogs vs. Cats" [115]. This dataset contains roughly 25,000 equally balanced images of cats and dogs (24,946 total images where 12,476 are cat pictures and 12,470 are dog pictures, which corresponds to approximately 50.01% and 49.99%, respectively). These images, although well balanced between both classes, lack consistency when it comes to their dimensions, which became one of the main issues to be tackled on this project since it would also be a problem when dealing with actual medical imaging data.

This particular dataset was comprised of two directories, each one containing various pictures of cats and dogs, separately. As can be observed in Figure 3.14 and Figure 3.15, these pictures vary immensely when it comes to their dimensions, lighting, background or main focus/number of entities present (either additional cats/dogs or some extra unrelated figures like humans or other animals). The script starts by importing all the necessary modules, such as "ImageDataGenerator", which is needed for the dataset preprocessing, and other essential neural network elements like the type of model that's going to be implemented and the different types of neural network layers. Following the imports, some variables were created to define the image's dimensions after preprocessing, which were established as 150 pixels for both height and width, the required dataset directories' paths, and the neural network's parameters.

Regarding the directories, the dataset was split into three separate sections, those being "training", "validation" and "testing". The first step of this distribution was obtained by taking 20% of the full dataset and allocating it to the testing directory, which corresponded to a total of 4,989 images (2,494 dog images and 2,495 cat images). From the remaining 19,957 images, 20% were once again set aside, but this time for the validation directory, corresponding to 3,991 images (1,995 dogs and 1,996 cats). All of the remaining images were assigned to the training directory, corresponding to 15,966 images (7,981 of those being dogs and the remaining 7,985 being cats).

Figure 3.14: Kaggle's dogs vs. cats dataset - Cat pictures [115]

Figure 3.15: Kaggle's dogs vs. cats dataset - Dog pictures [115]

These total numbers of samples per directory were also defined in the script's initial variables alongside the desired number of epochs and number of samples per batch of data which, for this project, were decided to be 50 and 128 respectively. It was also in this section that the network's input format was specified by employing a Keras utility that detects if the RGB channels of the input image are located at the beginning or at the end of the format.

The architecture of the neural network (see Figure 3.16) was implemented into a Keras' sequential model, being comprised of three convolutional layers followed by their respective ReLU activation and Max-Pooling layers. The first two convolutional layers consist of 32 output filters and a 3 by 3 kernel size, while the third convolutional layer had 64 output filters, maintaining the same kernel dimensions. Following that, the data was flattened and submitted to a dense layer (or fully connected layer) of size 64 based on a ReLU activation layer. The final section of the network consisted of a dropout layer, aiming to reduce potential overfitting of the algorithm. The final dense and activation layers based on the sigmoid function.

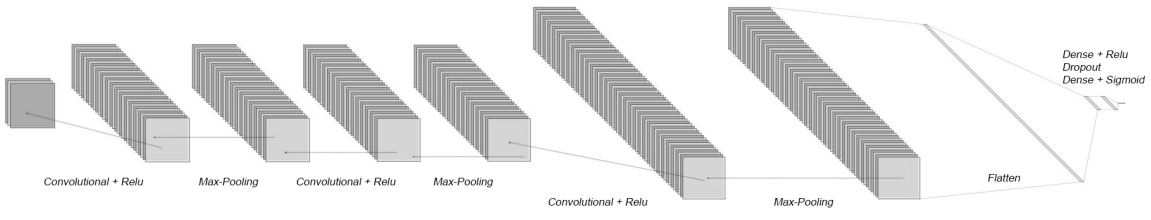


Figure 3.16: Schematics of the CNN model used during the Kaggle's dataset exercise

The model's configuration was also decided in this section of the program. Right after establishing the architecture, the network's compilation is completed by deciding on which optimizer, loss function and evaluation metrics should be used to produce the best accuracy results possible with these settings. After some experimentation with various different optimizers and loss functions, the chosen combination was determined to be an Adam optimizer and a binary cross entropy loss function with the accuracy metrics.

Regarding data preparation, the first algorithm runs were performed with an unaltered dataset, in other words, without using any sort of data augmentation. With that being said, the only data preparation done was a resize to the previously mentioned dimensions (150px by 150px) and a 1/255 rescale of the images in order to transform a potential 0 to 255 pixel value, into a 0 to 1 range. It was also here that the corresponding training, validation and testing directories were assigned.

The results obtained from this first test are represented in Figure 3.17. As can be observed, the 50 epochs that were used on this first experiment were clearly more than necessary for the algorithm to reach its peak validation accuracy. Seeing as the dataset does not suffer any improvement in size or alterations during the training process, it only took around 10 epochs for the program to achieve its maximum potential. For these settings, the maximum validation accuracy obtained was 84.73% while the testing accuracy, obtained from independent data never used in the training process, resulted in a 83.10% score.

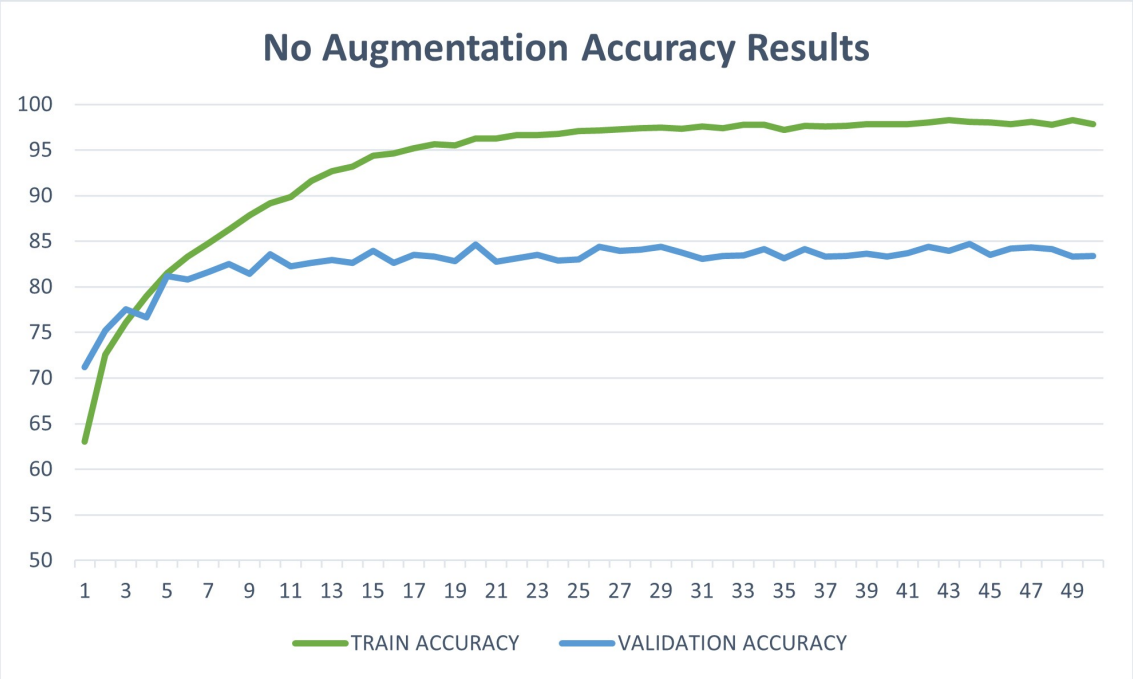


Figure 3.17: Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (no dataset augmentation)

Taking into account the results obtained, an offline dataset augmentation process was considered. This augmentation doubled the size of the training and validation dataset by utilizing an image augmentation library called Augmentor. This library provided the possibility to perform various image transformations at specific rates specified by the user. For this case, the referred datasets were submitted to semi-random slight rotations, zooms, and horizontal flips. After the modifications, the training dataset now had a total of 31,932 images (15,962 dogs and 15,970 cats) and the validation dataset a total of 7,982 images (3,990 dogs and 3,992 cats). The testing directory was not submitted to any changes in order to maintain trustworthy results. The results obtained are depicted in Figure 3.18.

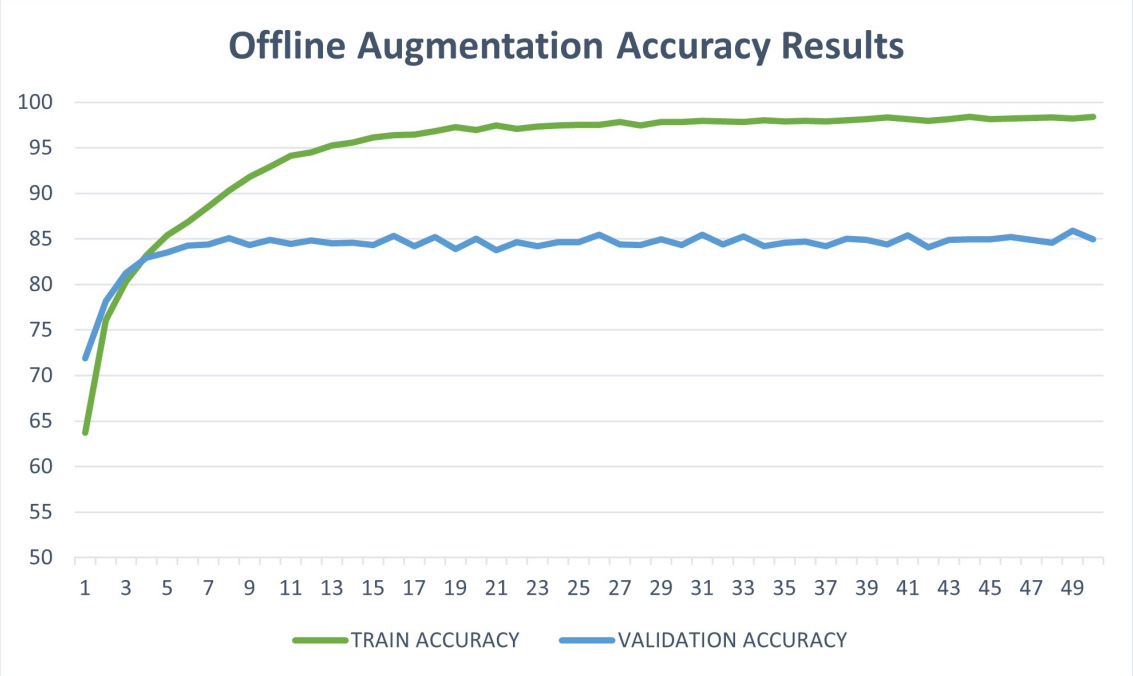


Figure 3.18: Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (offline dataset augmentation)

Even though the dataset suffered a major improvement in size, the number of epochs needed for the program to reach its peak values was still relatively low, not changing much over the previous test run on the base dataset. On the other hand, the accuracy scores did actually sustain a slight improvement over the previous ones: the maximum validation accuracy obtained improved 1.15% (85.88%) and the testing accuracy improved a significant 2.93% (86.03%). Despite the improvements seen from the first to the second trial, the biggest upgrade in results came with this third set of modifications. For this iteration, rather than submitting the base dataset to an offline augmentation process, it was decided to implement an online augmentation operation into the training process.

The type of alterations performed remained the same, but this time they were executed during the training procedure. As result, the training data is unique through each iteration, while not increasing the dataset size itself seeing as though the changes were not permanent. This type of augmentation of course, came with a significant increase in processing and training requirements. The results for this procedure, in terms of learning curves, can be observed in Figure 3.19.

In this case, the increase in the number of epochs needed to achieve peak performance is immediately visible. It can actually be argued that the 50 epochs used might have been slightly less than needed for this achievement. Even with that in mind, the improvement in accuracy results was clear: the maximum validation accuracy obtained was 90.34% (a significant 4.46% improvement from the offline augmentation variation) and the testing accuracy obtained was 89.76% (a 3.73% increase).

The two augmentation variations were then combined in order to assess any potential further enhancements of the accuracy results. Results can be seen below (Figure 3.20). Again, only a small development can be observed on both accuracy scores: 90.68% on validation accuracy (0.3% improvement) and 90.18% testing accuracy score (0.42% improvement).

The knowledge acquired with the implementation of this binary classification problem proved to be useful later during the medical imaging data pre-processing and splitting, as well as in the different development stages of the deep models for classification of Alzheimer's disease.

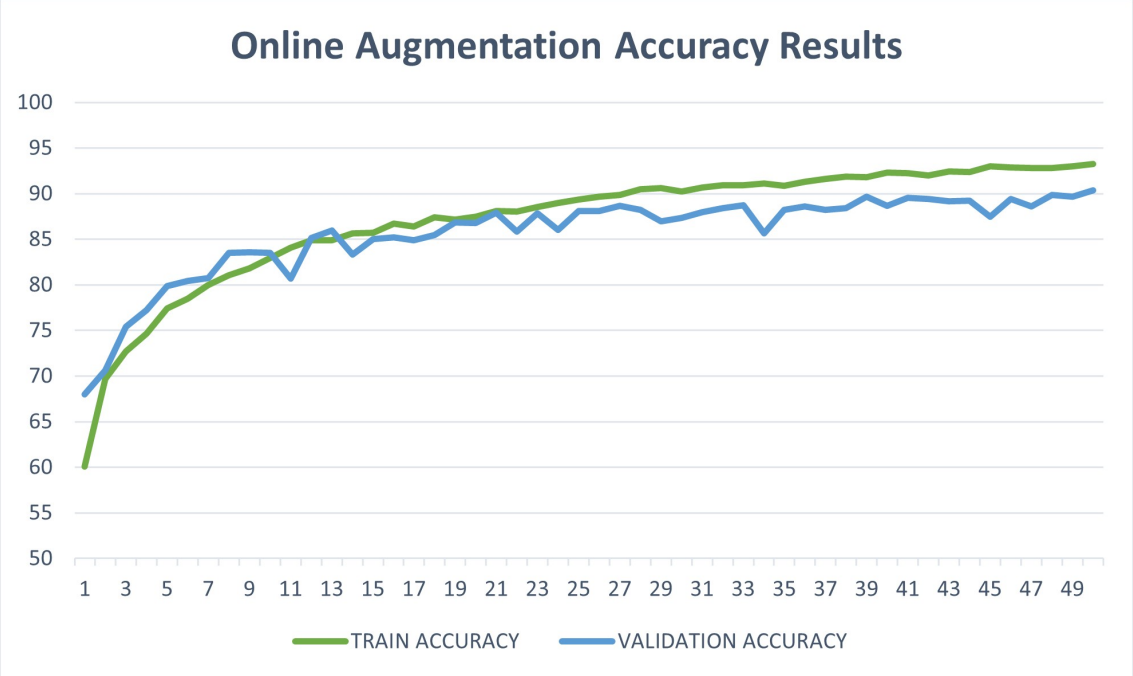


Figure 3.19: Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (online dataset augmentation)

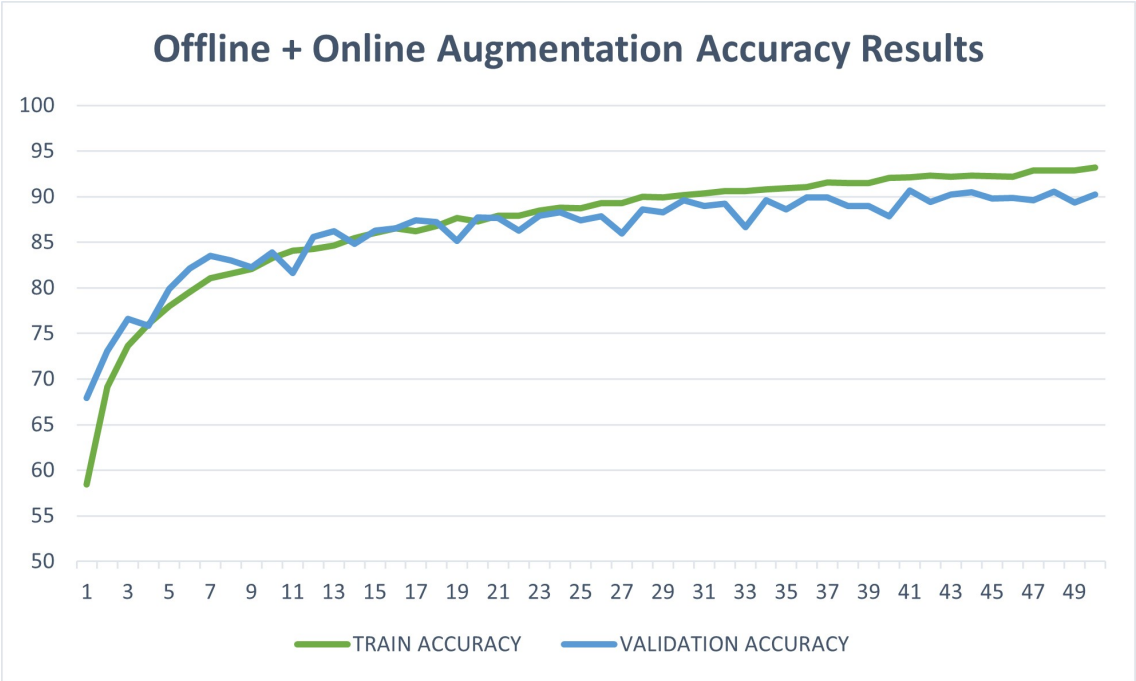


Figure 3.20: Kaggle's dogs vs. cats dataset - Training and validation accuracy results per epoch (offline + online dataset augmentation)

3.3.3 Application of 2D CNNs for Volumetric Data

The approach followed in this study to convert the 3D PET volume into a 2D image is commonly known as 2D slice-level CNN. It consists of a regular two-dimensional convolutional neural network whose input consists of 2D slices extracted from the 3D data (Figure 3.21). The main advantage of this implementation reside in the possibility of using pre-trained models and transfer learning. Since it is a regular 2D CNN, other previously proven successful image classification algorithms can be used in order to improve the model's performance quickly and efficiently.

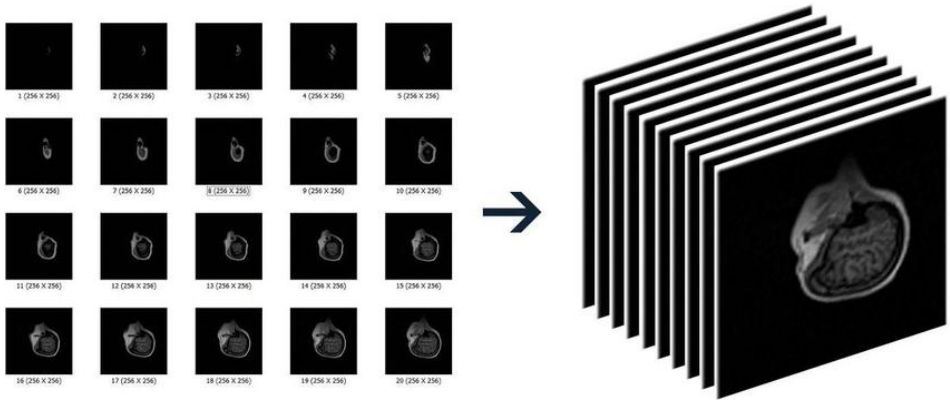


Figure 3.21: 2D slice-level approach of 3D volume processing [116]

This approach also provides the possibility for a significant increase in training data, considering that a large number of 2D slices can be obtained from a single 3D sample. In turn, this increase in the size of the dataset could have positive consequences in the algorithm's robustness. Despite this positive aspects, it also presents some downsides, mainly pertaining to data leakage and the lack of depth in the convolutional operations.

Data leakage occurs when somehow, during the network's training process, the system takes advantage of some information that under regular circumstances would not be available. This may lead to overly promising results if the necessary preventive measures are not correctly applied. For example, this may occur due whether slices provided by the same 3D sample (i.e., the same individual) end up on both the training and the testing dataset. The problem regarding the lack of depth can be understood when considering that by removing the third dimension element, the capability to correlate spatial features and patterns between the three dimensions to the corresponding results is lost in the process.

3.4 Three-Dimensional CNNs

While the main focus of this dissertation is centered around PET scans, there are many other modalities like CT, Ultrasound and MRI using deep learning. In this context, a typical 3D-CNN is very similar to a 2D-CNN in terms of architecture. It takes input data in a 3D format and are based on 3D convolution feature extractors. At the same time, it is important to note that training 3D CNNs may require a rather large labeled dataset in order to accomplish the expected training results. In addition to this difficulty, the complex training may also require a high computational resources that would amount to another possible obstacle to the completion of that task.

When it comes to 3D medical imaging, there are several methods that leverage the 3D information. By following the data gathered by Ebrahimighahnavieh et al. [84], it is possible to analyze the distribution of the different methods being employed in present-day researches in order to process volumetric data with CNN-based architectures (Figure 3.22). The slice-based techniques, referred in Subsection 3.3.3, have been used in 27% of these studies. The next subsections will further detail the patch-based, ROI-based and voxel-based techniques. This methods use the intensity value associated to each 3D data point, or voxel, present in the neuroimaging scan.

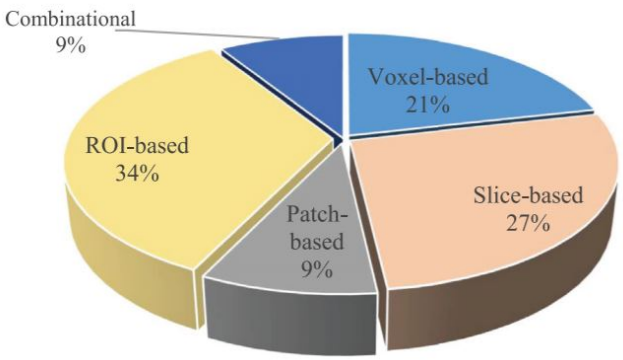


Figure 3.22: Prevalence of each approach to 3D input data management [84]

3.4.1 3D Patch-level CNN

Similar to the 2D slice-level method, the 3D patch-level approach also implements a form of data segmentation. With this technique, instead of relinquishing three-dimensional properties by dividing the 3D volume into 2D slices, the data is partitioned into smaller 3D patches and processed from there. This method would also have the potential to increase the quantity of training data, following the same logic of dividing each 3D sample into smaller ones.

On a case-by-case scenario, the architectural structure may vary depending on the size chosen for each patch and the type of study being tackled. When handling larger patches, there have been cases where multiple CNNs were used, each one corresponding to a certain patch position, and later assembled into a larger CNN. On the other hand, there has also been instances where a combination of both CNNs and clustering techniques were implemented in order to process patches of smaller sizes.

In addition to the training data increase, this method can also be advantageous due to its lower memory usage and the possibility for a lower number of parameters being considered when dealing with only one CNN. Considering that some of the spatial features and patterns can now be taken into consideration with this system, this is still not a foolproof procedure seeing as some of the spatial characteristics would be separated along with the sample's patch divisions.

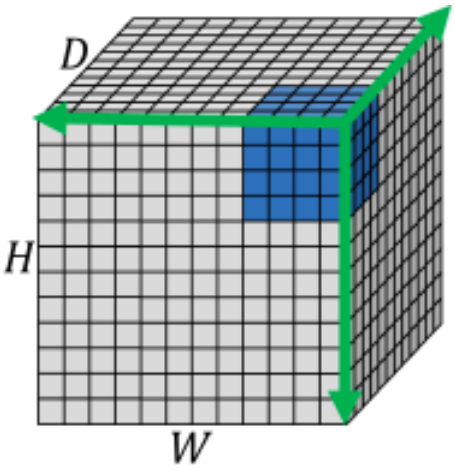


Figure 3.23: 3D patch-level approach to dealing with volumetric input data [117]

3.4.2 ROI-based CNN

While the 3D patch-level CNN method takes the whole 3D sample and subdivides it into smaller 3D patches, a ROI-based CNN only takes into account a given region of interest that vary according the problem to tackle. The main objective of this technique is the removal of any unnecessary information and the exclusive focus on the key data sections of the sample (the specific region of interest). Therefore, it simplifies the process without losing any important data.

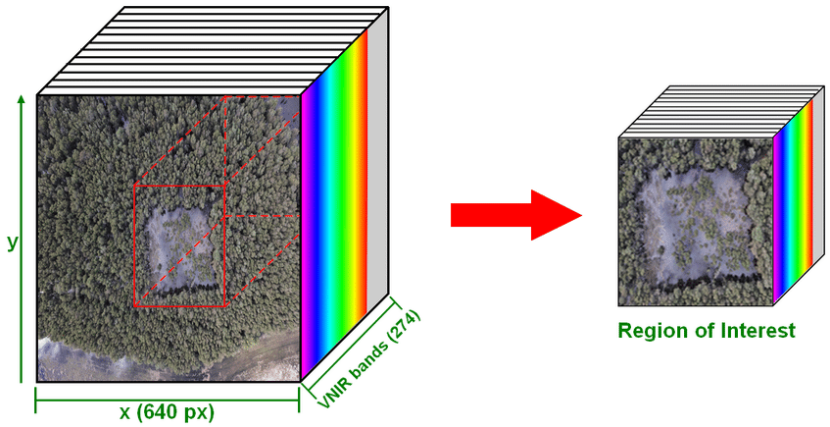


Figure 3.24: Region of interest selection in a ROI-based approach to dealing with volumetric input data [118]

When dealing with brain disorders, this feat is commonly achieved by determining the exact brain region where this condition mainly takes place. Being a seemingly positive evolution of the 3D patch-level alternative, a ROI-based model will only be an appropriate approach to adopt when the respective exercise's key data can be precisely pinpointed to a well defined area of the sample. This would be a difficult feat to achieve when dealing with AD diagnosis.

3.4.3 3D Subject-level CNN

The approach taken in this study, referred as to 3D subject-level CNN model, considers a CNN model that processes all of the available information. Within the scope of this work, the selection of this approach will involve the development of a custom 3D-CNN to take advantage of spatial patterns on the full PET volumes by using 3D filters and 3D pooling layers. The main difficulty that should be considered is the necessity for a high processing power machine in order to be able to run such complex data structures and procedures. Additionally, a major concern will the reduction of training samples since the process is reduced to a sample per subject instead of multiple subdivisions (as it was the case in previous methods).

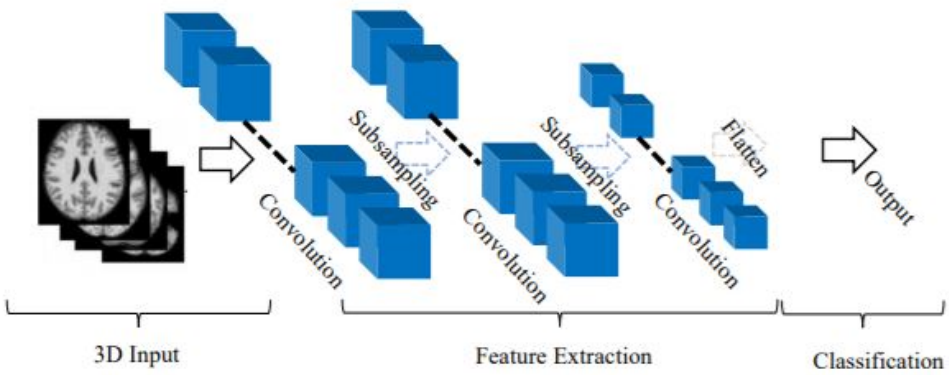


Figure 3.25: 3D subject-level approach to dealing with volumetric input data [119]

Chapter 4

Experiments and Results

This chapter presents the implementation details and the performance evaluation of two CNN-based models for the automatic diagnosis of AD using the ADNI database. Section 4.1 provides an overview of the PET dataset. Section 4.2 focuses on the transfer learning approach based on a pre-trained 2D-CNN model. Section 4.3 addresses the custom 3D-CNN model trained from scratch. The two approaches are compared in a binary classification framework including two classes: Cognitively Normal (CN) and Alzheimer's Disease (AD).

4.1 Dataset Overview

The ADNI dataset consists of PET scans saved in the the most commonly used file formats for neuroimaging data. This includes the NII file format (or NIfTI) which stands for Neuroimaging Informatics Technology Initiative. The images were, subsequently, extracted using the python library NiBabel to facilitates the reading procedure. Since this work is focused only in the CN and AD classes, the final dataset distribution ended being as follows (see Figures 4.1 and 4.2): 866 CN samples (63.91%) to 489 AD samples (36.09%) and 796 male patients (58.75%) to 559 female ones (41.25%). A detailed age distribution is presented in Figure 4.3.

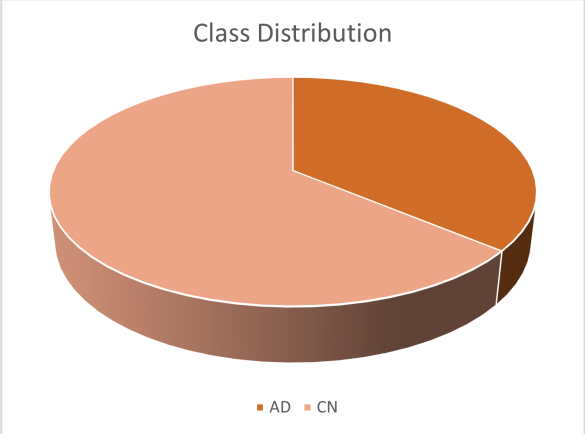


Figure 4.1: Dataset's class distribution

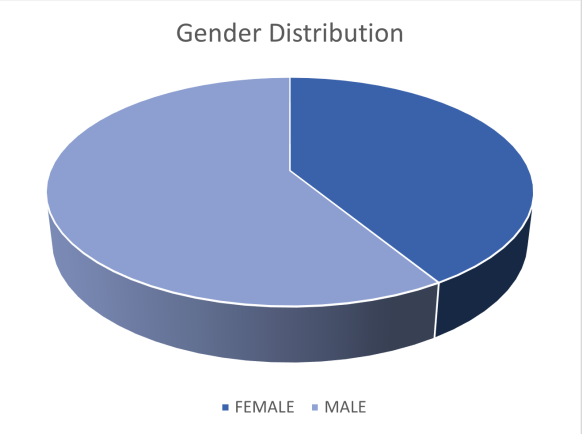


Figure 4.2: Dataset's gender distribution

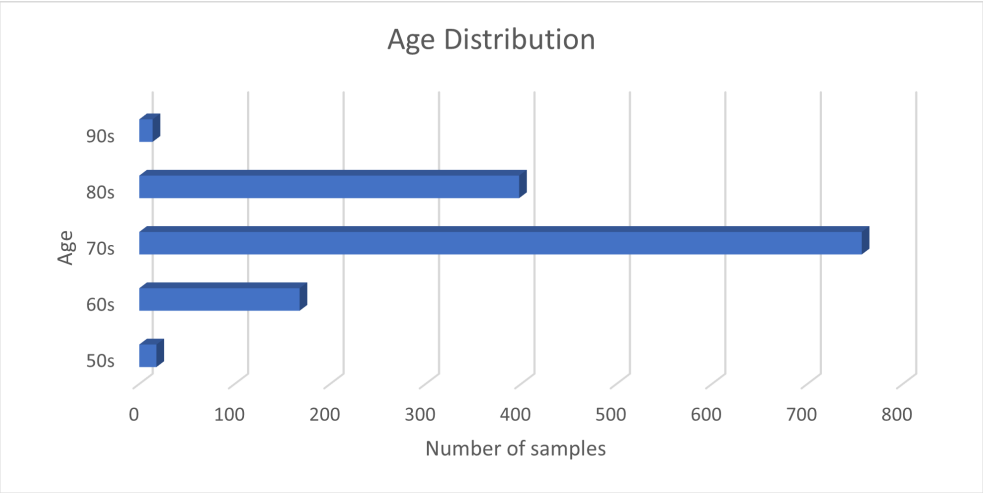


Figure 4.3: Dataset's age distribution

During this task, two different dataset splits were used (A and B). Dataset A was obtained by splitting the 1355 total samples into ten folds, each one consisting of approximately 48 to 49 AD samples and 86 to 87 CN samples. Dataset A was used during the initial model training by implementing a 10-fold cross-validation procedure (method further explained in Subsection 4.2.2).

Dataset B was obtained from the same type of dataset division, but instead of splitting the 10 folds from the initial 1355 total samples, it was done after separating 105 samples out of that total sum (66 from the CN class and 39 from the AD class). The remaining 1250 samples were then subdivided into the 10 folds consisting of 80 CN samples and 45 AD samples for each one. The 105 samples left apart served as a test sub-set independent of the cross-validation performance metrics.

4.2 Transfer Learning Approach (2D Model)

This section organised into several sub-sections detailing the data pre-processing, the CNN model, the trials that were made in order to reach a final model configuration, the final tests and results, and an additional dropout-oriented study. For the data pre-processing, it will first detail how the script finds and converts every NII file into readable numpy arrays. It will then check its dimensions, normalize its values, and extract 16 specific frames in order to assemble a 4 by 4 JPEG collage for each subject. The 2D slice-level model subsection describes the data samples' formats, the cross-validation method, the chosen network architecture and how the dataset was used over the trials. The last two subsections are dedicated to observing and interpreting the final results. The most efficient model configuration was defined and the impact that each studied parameter has on the model's performance was properly examined.

4.2.1 Data Pre-Processing

In the 2D approach every NII file was isolated in the downloaded dataset's directories. This was achieved through the python module "glob" which allows for dynamic manipulation

of directory paths, using pre-determined patterns, in order to retrieve every desired file that fulfills those specified requirements (every averaged NII file in this case).

After locating every file and its corresponding path, these are then saved in an array, and its size (or length) is saved as the number of samples found. Next, a *for* loop is initiated, which will go through each entry of that same path array and load the NII file using the NiBabel python library and its tools.

With the respective data sample loaded into a numpy array, the script proceeds to print its original shape, normalize its values into a 0-255 interval, and save the number of frames of that specific sample into a new count variable. This is necessary seeing as this dataset has PET scans from various different machines, which consequently leads to a constant variation of resolution in terms of number of frames taken from the base to the top of the subject's brain.

At each *for* loop step, 16 frames that approximately correspond to the same brain sections are selected. These scans can have varying resolutions. The selected frames are then assembled (JPEG format) in a 4 by 4 collage (the first frame being at the top left location, and the last one at the bottom right) with the aid of OpenCV packages (see Figure 4.4).

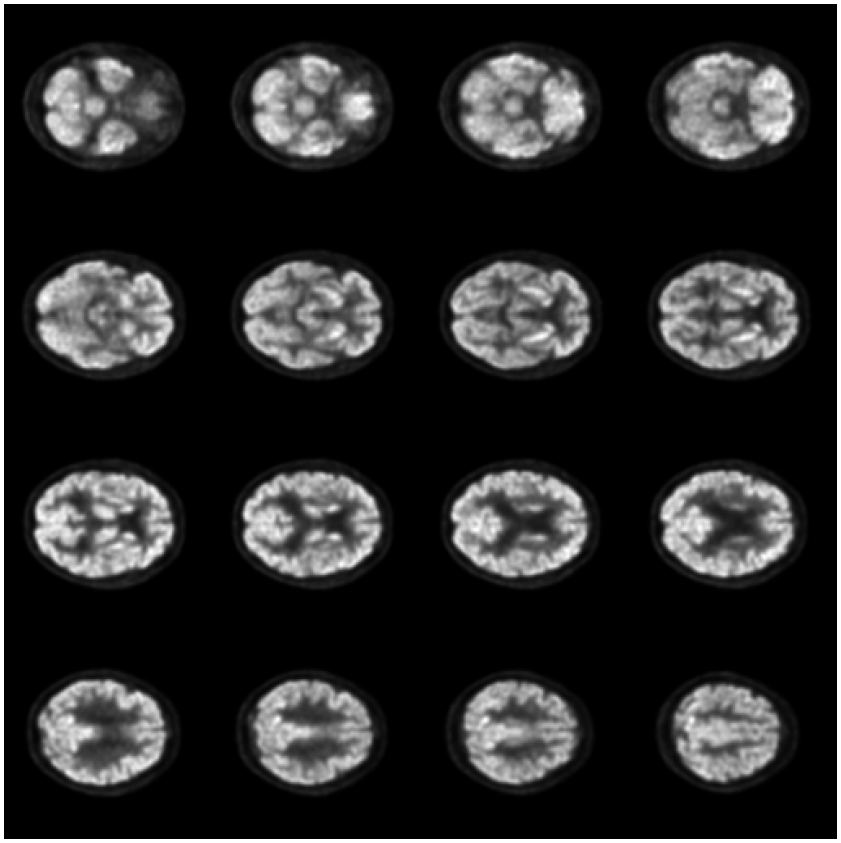


Figure 4.4: Collage of 2D slices extracted from volumetric PET scans

4.2.2 2D Slice-level Model

This approach was inspired by Yiming Ding's [12] 2D collage of a 4 by 4 collection of FDG-PET scan slices. The major challenge of this approach is related to the different formats that

the scans possessed. Starting at 128x128x35 voxels and going up to 400x400x144 voxels, with the average values being around 150x150x70 voxels, it was clear that a consistent format for these collages would be a difficult task to achieve.

After the necessary data pre-processing, the images were divided into the two classes, CN and AD. The K-fold cross-validation (CV) method was applied, where data was divided into K=10 folds (Figure 4.5). For each repetition, 9 folds were used for training and 1 fold for validation. The final validation performance is then quantified as the average of the validation metrics over the K runs.

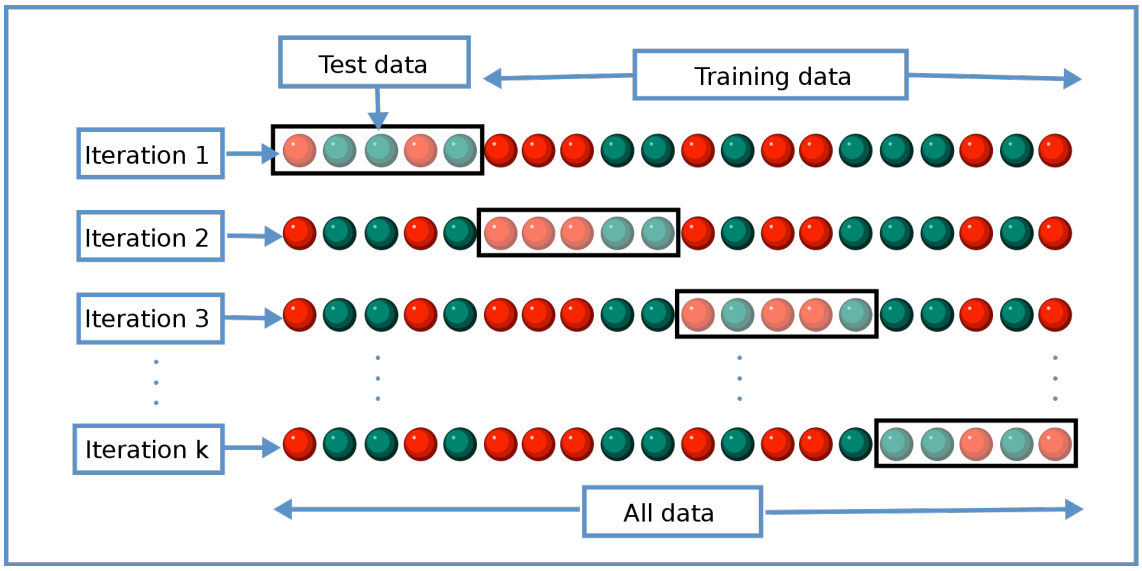


Figure 4.5: Cross-validation method [120]

The python implementation starts with the configuration of the desired GPUs/CPUs, through Keras and TensorFlow tools, in order to customize the hardware where the program will run, the training speed and the memory usage. After the hardware configuration, some variables are initiated such as the desired image height and width for the 4 by 4 input collage (both at 512 pixels in this case), the number of training and testing samples, the number of epochs and batch size.

Google's Inception V3 was chosen as the pre-trained CNN architecture. This widely used model is rather well known around the CNN community due to its impressive accuracy results on the ImageNet dataset, which contains 1000 different classes and around 1.3 million data samples. Another useful trait that this model possesses is its usage of building blocks, and its easy customization (Figure 4.6), which allows the user to fully adapt this architecture to whatever demands the problem requests. The inception blocks are also called "mixed" blocks.

Different Inception V3 architectures were trained, consisting of a different number of sub-networks, trying to optimise the classification accuracy while using a minimal number of parameters. These sub-networks were built by taking the output of the last considered inception block and directly connecting it to the fully connected layers at the end of the network. Figure 4.7 summarizes the Inception V3 models that were studied and their corresponding number of parameters.

Building customized sub-networks allows for a wider range of experimentation with different hyper-parameters, such as the varying input shape. After successfully accessing the

output of the desired "mixed" block, it is then redirected to a Global Max Pooling 2D layer. Then it goes to a fully connected layer with 1024 neurons and a ReLU activation function, followed by a Batch Normalization layer, a 50% dropout layer and ending with a Sigmoid activation function, which is the most suitable output layer activation function when dealing with binary classification. Binary cross-entropy loss function and the accuracy metrics are the arguments for this compilation step.

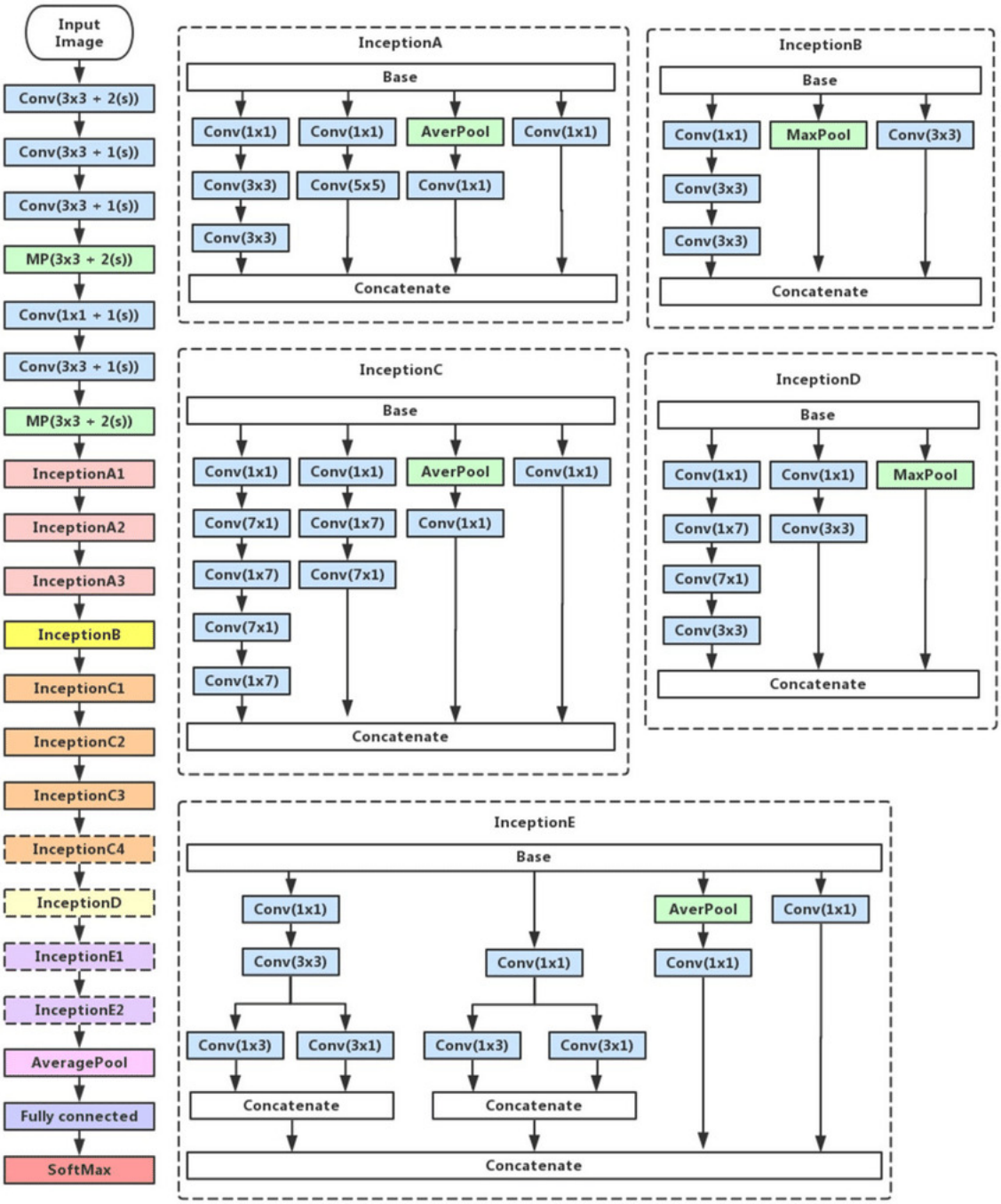


Figure 4.6: Inception V3 architecture and corresponding modules [121]

Inception-v3	Total	Trainable	Non-Trainable
Mixed6	6,834,468	6,819,492	14,976
Mixed7	8,978,340	8,959,524	18,816
Mixed8	10,679,972	10,658,596	21,376
Mixed9	15,730,916	15,703,012	27,904
Mixed10	21,810,980	21,776,548	34,432

Figure 4.7: Number of parameters per Inception V3's sub-networks [121]

The training and testing dataset were defined by making use of the ImageDataGenerator tool from the Keras Image Pre-Processing package. This allows for the user to easily build up these datasets directly from their directories, while simultaneously normalizing, resizing, and encoding their respective labels (in a binary class mode for this specific model).

In this first set of experiments, only a portion of the full dataset was used in order to maintain a balanced amount of samples for both classes (Dataset A-Balanced). In the subsequent trials the full dataset was used (Dataset A-Imbalanced). Table 4.1 summarises the CV average accuracy and standard deviation over the 10 experiments for 4 different Inception V3 models and 4 different optimizers. The results show that the "mixed8" Inception V3 variation outperforms the other models. Further to that, the SGD (Stochastic Gradient Descent) optimizer maintained consistently better results when compared to the other optimizer options. Note that, the results presented in Table 4.1 were obtained after having initialized the Inception V3 models with pre-trained weights using the ImageNet dataset. Though the ImageNet dataset has nothing to do with medical scans, using a pre-trained model was favourable as a starting point for fine tuning only the final (custom) layer.

As previously mentioned, an imbalanced dataset was used from this point forward, including Table 4.2 and its corresponding results, in order to increase the already significantly limited amount of available data. Here, only the "mixed8" Inception V3 model was used, being the most favourable model from the previous set of experiments. The optimizer and the number of training epochs were the hyper-parameters to be analyzed. The top 3 optimizers from the previous set of experiments were compared: SGD, Adadelta and Adamax (a variant of the Adam optimizer). As expected, the performance of the models, fine tuned on more even unbalanced data, is still overall better than the results in Table 4.1.

Architecture	Trainable Parameters	Epochs	Optimizer	Cross-Validation Average Accuracy	Standard Deviation
InceptionV3 (mixed7)	9,746,977	15	Adamax	75.16%	6.35%
InceptionV3 (mixed7)	9,746,977	15	Adadelta	77.11%	11.30%
InceptionV3 (mixed7)	9,746,977	15	SGD	77.52%	7.85%
InceptionV3 (mixed8)	11,968,289	15	Adamax	78.20%	10.41%
InceptionV3 (mixed8)	11,968,289	15	Adadelta	77.68%	9.98%
InceptionV3 (mixed8)	11,968,289	15	SGD	81.50%	7.61%
InceptionV3 (mixed8)	11,968,289	15	Adam	75.89%	11.69%
InceptionV3 (mixed9)	17,796,065	15	Adamax	71.27%	8.32%
InceptionV3 (mixed9)	17,796,065	15	SGD	76.69%	4.37%
InceptionV3 (mixed10)	23,869,601	15	SGD	74.75%	7.36%
Note: Dataset A-Balanced, 978 total samples (489 for each AD and CN) 512x512 pixels per input collage (128x128 pixels per each of the 16 frames)					

Table 4.1: 2D Binary Classification - Balanced Dataset

Architecture	Trainable Parameters	Epochs	Optimizer	Cross-Validation Average Accuracy	Standard Deviation
InceptionV3 (mixed8)	11,968,289	10	SGD	67.87%	15.34%
InceptionV3 (mixed8)	11,968,289	15	SGD	83.62%	2.18%
InceptionV3 (mixed8)	11,968,289	20	SGD	81.40%	7.39%
InceptionV3 (mixed8)	11,968,289	10	Adamax	76.19%	15.00%
InceptionV3 (mixed8)	11,968,289	15	Adamax	80.60%	6.61%
InceptionV3 (mixed8)	11,968,289	20	Adamax	77.12%	6.17%
InceptionV3 (mixed8)	11,968,289	10	Adadelta	77.81%	9.58%
InceptionV3 (mixed8)	11,968,289	15	Adadelta	78.60%	12.38%
InceptionV3 (mixed8)	11,968,289	20	Adadelta	78.21%	11.44%
Note: Dataset A-Imbalanced, 1355 total samples (489 for AD and 866 for CN) 512x512 pixels per input collage (128x128 pixels per each of the 16 frames)					

Table 4.2: 2D Binary Classification - Imbalanced Dataset

The "mixed8" Inception V3 model, proposed in this work. was compared with other common CNN architectures, namely LeNet5 model (Figure 4.8) and two variations of the ResNet50 architecture. The results are summarized in Table 4.3. These trials were also executed without ImageNet weight initialization, since the Keras API does not possess LeNet5 as an available pre-trained model, and therefore the option to load the corresponding ImageNet weights was not available. Since LeNet5 is a simpler deep neural network (DNN), higher batch size was possible to implement, such as 32. In contrast, ResNet50 is a more complex DNN so, in order to make the training feasible, the batch size was reduced to 8. The Inception model outperformed the ResNet50, however, LeNet5 achieved competitive performance. This result can be explained with the relatively small data set, for which simpler architectures are more favourable.

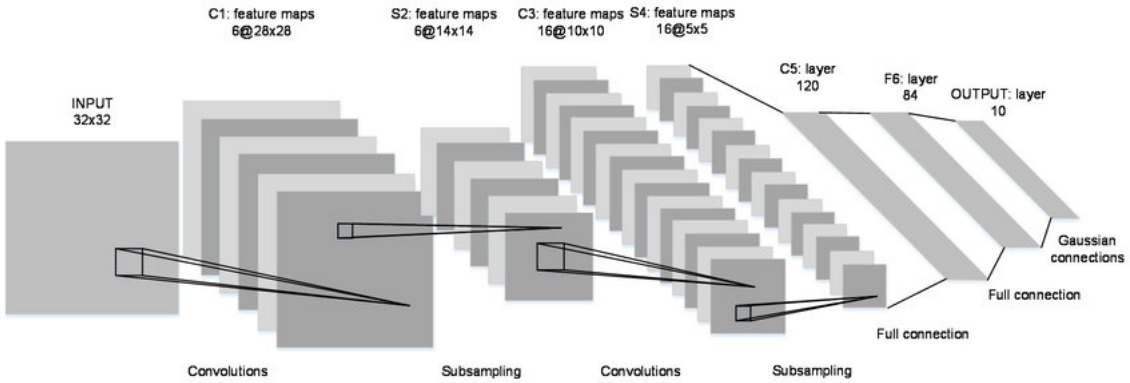


Figure 4.8: LeNet5 architecture [122]

Architecture	Trainable Parameters	Epochs	Batch Size	Cross-Validation Average Accuracy	Standard Deviation
InceptionV3 (mixed8)	11,968,289	20	16	72.78%	8.92%
InceptionV3 (mixed8)	11,968,289	30	16	78.32%	5.83%
InceptionV3 (mixed8)	11,968,289	40	16	74.48%	14.83%
LeNet5	30,493,337	20	16	75.51%	4.98%
LeNet5	30,493,337	25	16	78.53%	3.89%
LeNet5	30,493,337	30	16	76.33%	5.31%
LeNet5	30,493,337	35	16	77.43%	3.84%
LeNet5	30,493,337	35	32	78.02%	5.60%
LeNet5	30,493,337	40	32	74.47%	4.16%
LeNet5	30,493,337	45	32	77.35%	3.71%
ResNet50 (bn5c)	23,536,641	10	8	72.17%	10.83%
ResNet50 (bn4f)	8,559,617	10	8	72.79%	14.50%
Note: Dataset A-Imbalanced, 1355 total samples (489 for AD and 866 for CN) 512x512 pixels per input collage (128x128 pixels per each of the 16 frames) No ImageNet weights initialization					

Table 4.3: 2D Binary Classification - Imbalanced Dataset (LeNet5 & ResNet50)

4.2.3 Train-CV-Test Results

This section presents, in detail, the results of the training, cross validation (CV) and testing stages of the compared DNN models. Note that, the testing accuracy is obtained by evaluating the model, after loading the weights from the best performing cross-validation fold. The test data are 105 independent samples never seen during the training or CV process. With the exception of the LeNet5 model, the other alternatives were initially pre-trained on ImageNet data.

After having determined what was generally the best framework combination for this approach, it was now time to assess the impact that each one of the adjustments had on this model's optimization process.

The following conclusions can be taken from Figure 4.9:

- All models suffer from overfitting issues, which is certainly related to the small dataset. The training accuracy in all models is significantly higher when compared to the CV and testing accuracy. Further to that, better testing accuracy than CV accuracy can be also explained due to the very small CV and test sub sets.
- The Inception V3 (mixed 8 sub-network) model exhibits the best performance with respect to new (test) data.
- The LeNet5 model was trained from scratch without the advantage of the pre-trained weights initialization like the other three models. It can be speculated that the LeNet5 model may outperform the other alternatives if it has been previously trained on a large database such as ImageNet.

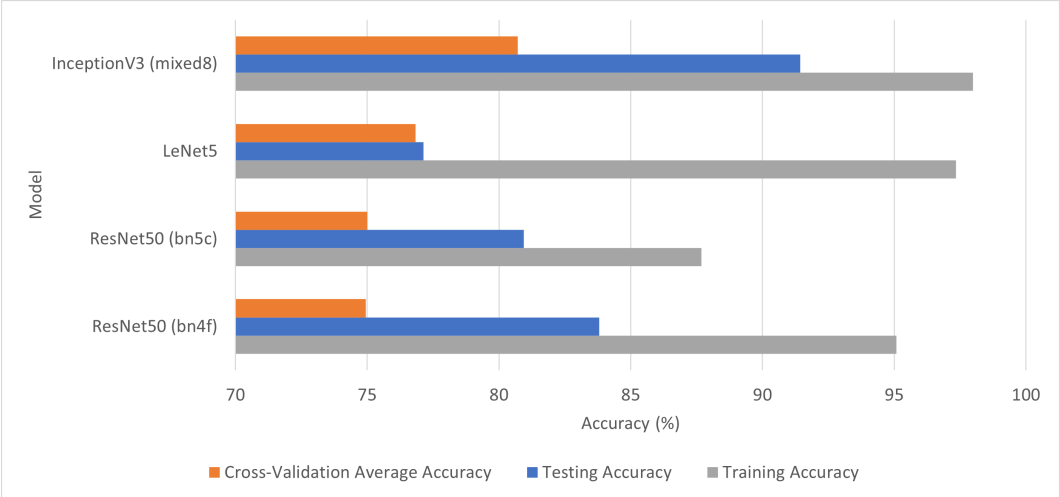


Figure 4.9: 2D binary classification - Network model comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; pre-trained ImageNet weights initialization, with the exception of LeNet5)

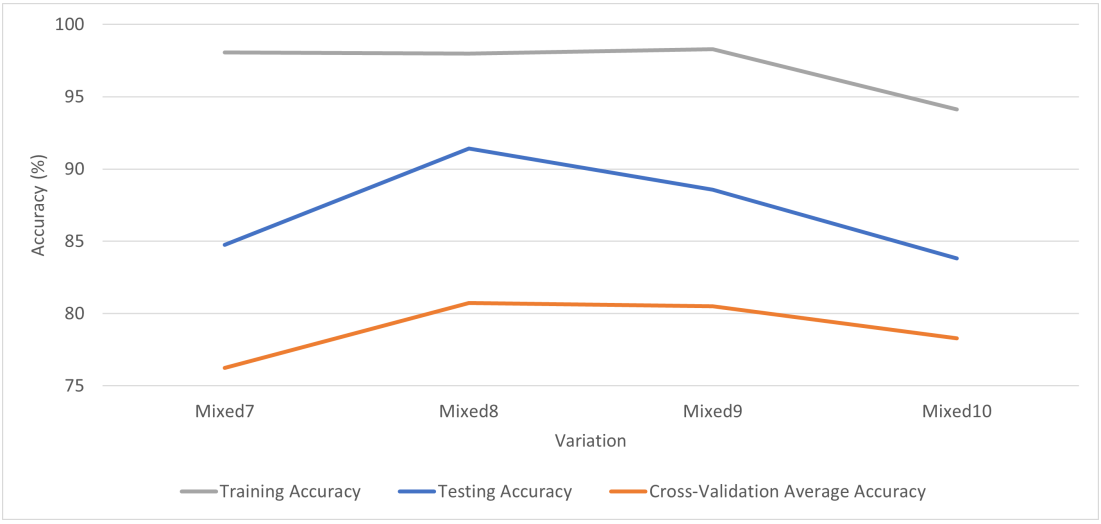


Figure 4.10: 2D binary classification - Inception V3 sub-networks comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; pre-trained ImageNet weights initialization)

The results with respect to the Inception V3 sub-networks comparison are summarized in Figure 4.10. Similarly to the previous conclusions, the models suffer from overfitting problems. Both, in CV and test stages, the mixed 8 DNN model exhibits the best performance. More complex models means more trainable parameters which is not suitable for this particular case study. The outcome of the study with respect to which is the most favourable optimization method is presented in Figure 4.11. Experiments were conducted with seven optimizers, Adadelta, Adamax and SGD were elected as the best options.

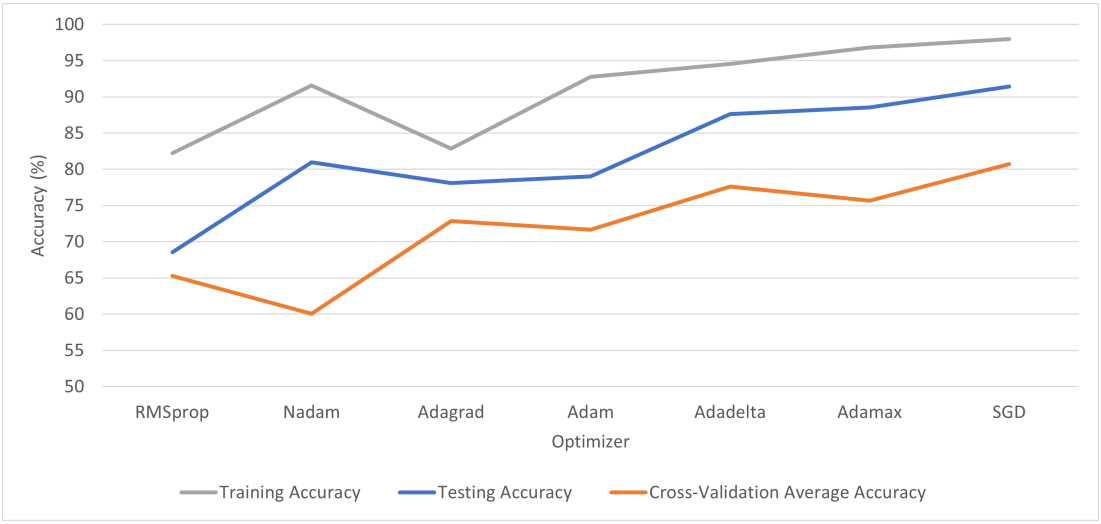


Figure 4.11: 2D binary classification - Optimizer comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization)

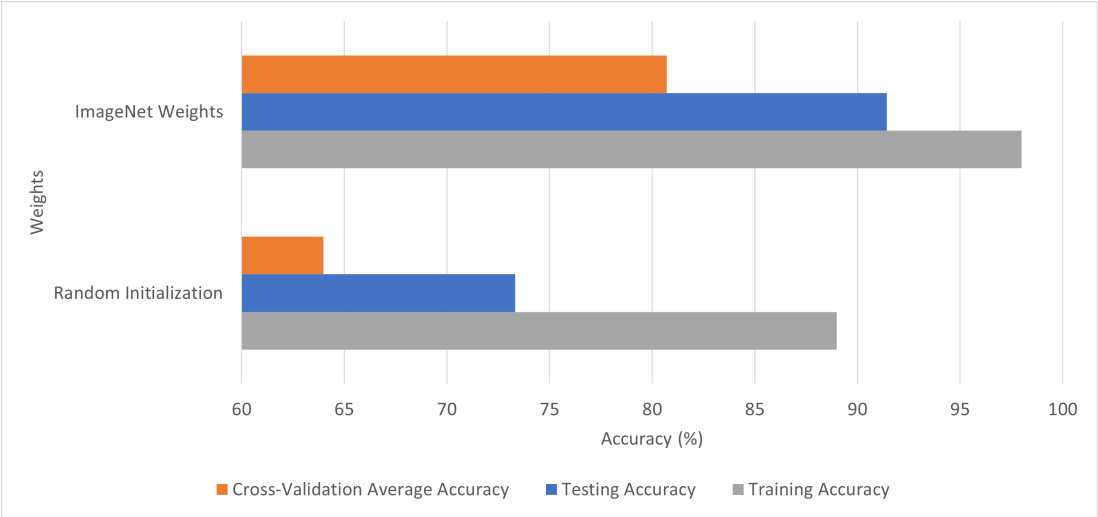


Figure 4.12: 2D binary classification - Weights initialization impact (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; SGD optimizer)

The last study was to analyse the impact of training the models with random weights initialization vs. pre-trained weights initialization with ImageNet. The results depicted in Figure 4.12 clearly show the positive impact of pre-trained weights initialization.

Finally, the best DNN model for the present 2D binary classification problem was the Inception V3 architecture (mixed8 sub-network, Figure 4.9 and Figure 4.10). This model was trained with the full dataset available, with the SGD optimizer (Figure 4.11), with pre-trained ImageNet weights initialization instead of a random weights initialization (Figure 4.12). The model reached a 10-fold CV average accuracy of around 83.62% (Dataset A-Imbalanced) and a testing accuracy of 91.43% (Dataset B).

4.2.4 Dropout Analysis

The overfitting problem was tackled by adding a dropout layer. Figure 4.13 provides direct performance comparison between different dropout rates. Note that the 50% dropout value provided the best outcome. Figure 4.14 and Figure 4.15 show the evolution of the training accuracy and loss (cost) function respectively, during the training epochs.

These graphs provide a smooth and simple direct visualization of the effects that a given dropout layer percentage causes on a model's training and performance. The smaller the dropout percentage is, the more susceptible it is to overfitting, and the less time it will take for the system to converge. On the other hand, if the percentage is too high, it will take a lot more time for the convergence to take place, and the system will become a lot more unstable.

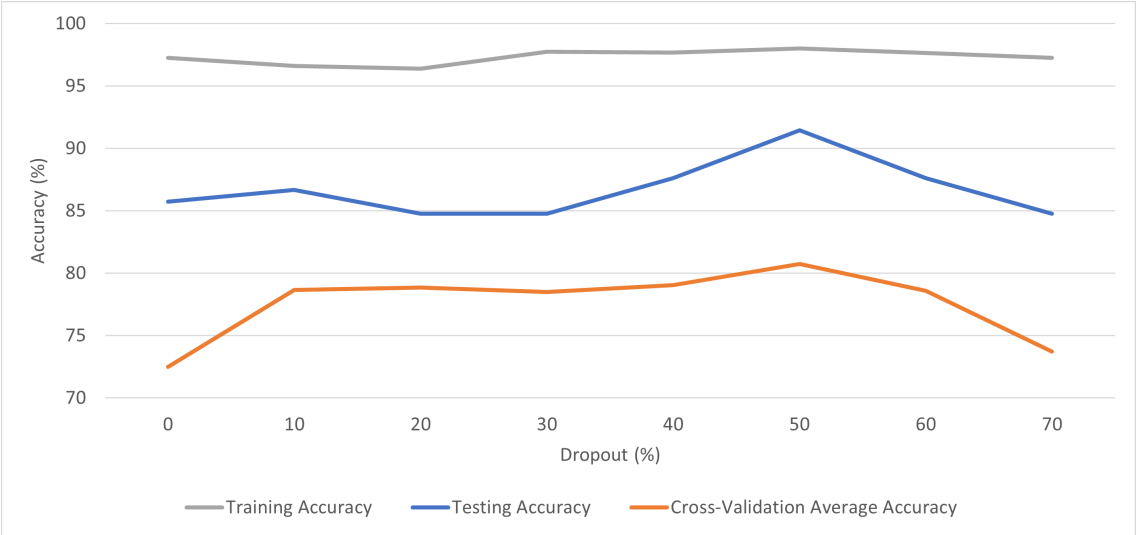


Figure 4.13: 2D binary classification - Dropout rates comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer)

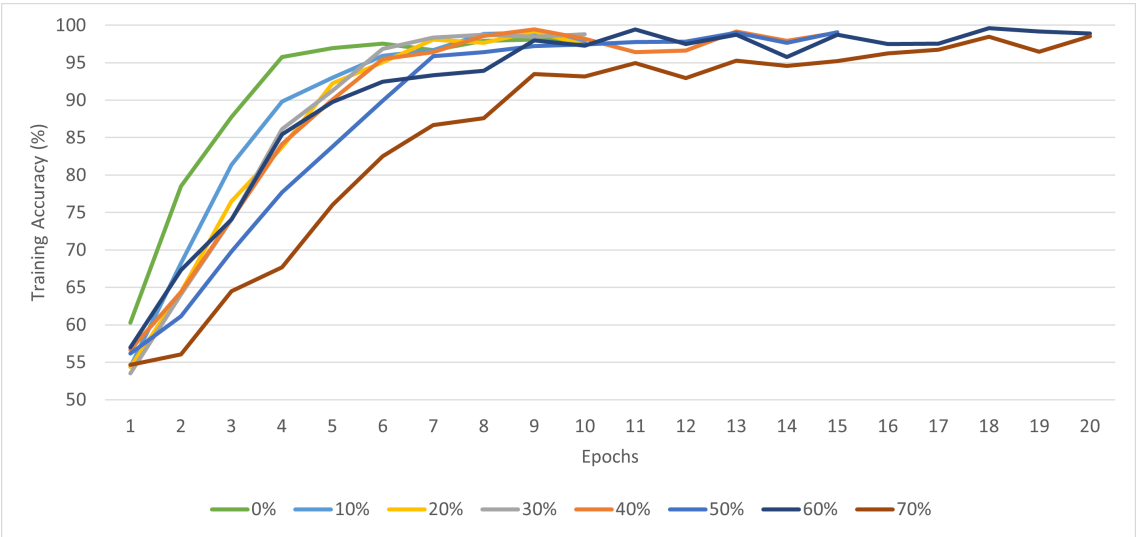


Figure 4.14: 2D binary classification - Training accuracy per epoch (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer)

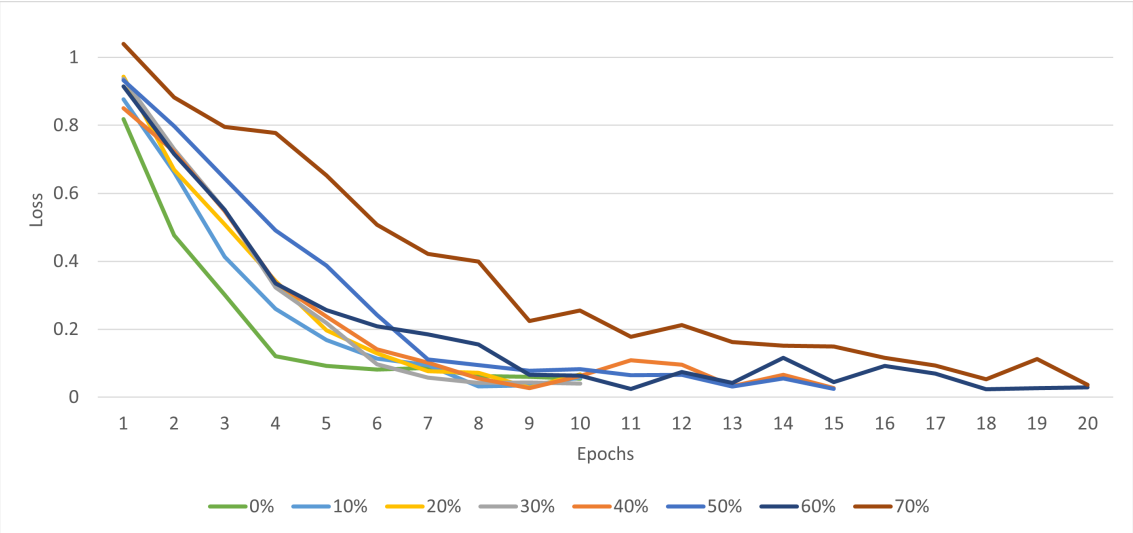


Figure 4.15: 2D binary classification - Loss per epoch during training (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; Inception V3 [mixed8]; pre-trained ImageNet weights initialization; SGD optimizer)

4.3 Custom Architecture Approach (3D Model)

Similar to Section 4.2, this section will also go over the same concepts: data pre-processing, CNN model configurations and fitting, final results and dropout study. In contrast to the previous transformation of the raw data into 2D image dimension, here the original data has to be first pre-processed into 3D tensors and then loaded into the network. How to define the best DNN configuration, the learning rate adjustment and multiple sample resolutions are the topics discussed in the second subsection. The last two subsections are focused into comparison of train-CV-test stages of performance and dropout analysis.

4.3.1 Data Pre-Processing

When dealing with 3D data, there are different techniques that need to be implemented when compared to its 2D counterpart. The base script is essentially the same from the creation of the files' location path array to the *for* loop and its data normalization process after loading the respective NII file.

From that point, new procedures were implemented, starting with an interpolation tool named "zoom", from the "Multidimensional Image Processing" SciPy package. This tool allows 3D data to be "reshaped" into certain dimensions pre-defined by the user. Similar to a resolution change in 2D imaging as it reduces, or increases, the total amount of pixels on a certain image while still "preserving" the original picture. After being reshaped into the desired dimensions (Figure 4.16), the samples are then appended into a numpy array and subsequently saved into a numpy file (*.npy*) in order to be more easily loaded in the future.

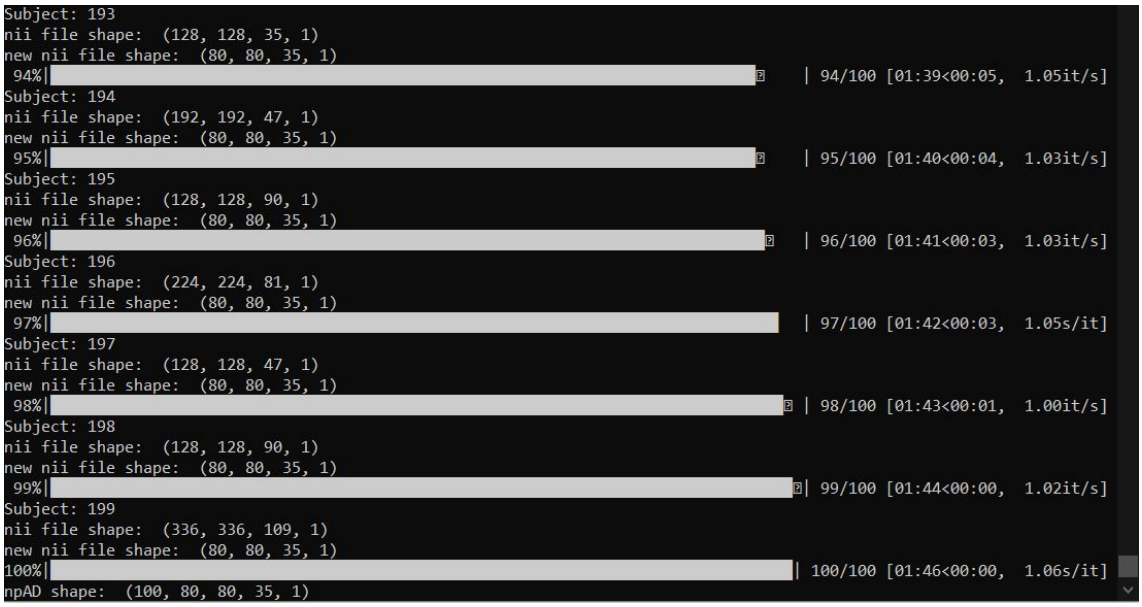


Figure 4.16: 3D binary classification - 3D pre-processing terminal output example

4.3.2 3D Subject-level Model

When it comes to the three-dimensional approach to this problem, the python script in itself has a very similar beginning to the two-dimensional one. Starting off by also configuring the computing hardware, this variation also needs to load the 3D samples that were saved into numpy files during the pre-processing procedure. These numpy files have to be loaded into their own separate arrays in 100 sample batches due to their high memory demands, and subsequently concatenated into a single joint array. This process is repeated for both classes and their corresponding shapes are printed afterwards so that to indicate a successful loading process. After that, each class array is completely shuffled.

With the data properly prepared, the cross-validation testing/training dataset division is carried out and their shapes printed. This process is done separately for each class and, once again, concatenated afterwards. Binary label arrays are also created in order to correctly assign the corresponding classes to their data samples. Afterwards, the zip tool is used so that to "connect" each sample position to its corresponding label and subsequently shuffled, that way maintaining each correct class pairing even after the randomization process (Figure 4.17).

The next point that deserves attention is the development of the network's architecture. Unlike the 2D approach where there are pre-trained models, when dealing with 3D data there are no previous architectures available. Therefore, the architecture used for this experiment was designed from scratch and progressively optimized. The base structure of this prototype consisted of two groups of two 3D Conv layers, a 3D max-pooling layer followed by a Batch Normalization and Flatten layer. At the end of these block, three progressively neuron decreasing fully-connected layers, interchanged with dropout layers in between them, are considered. The model was trained with the binary cross-entropy loss function (log loss) and the SGD (Stochastic Gradient Descent) optimizer.

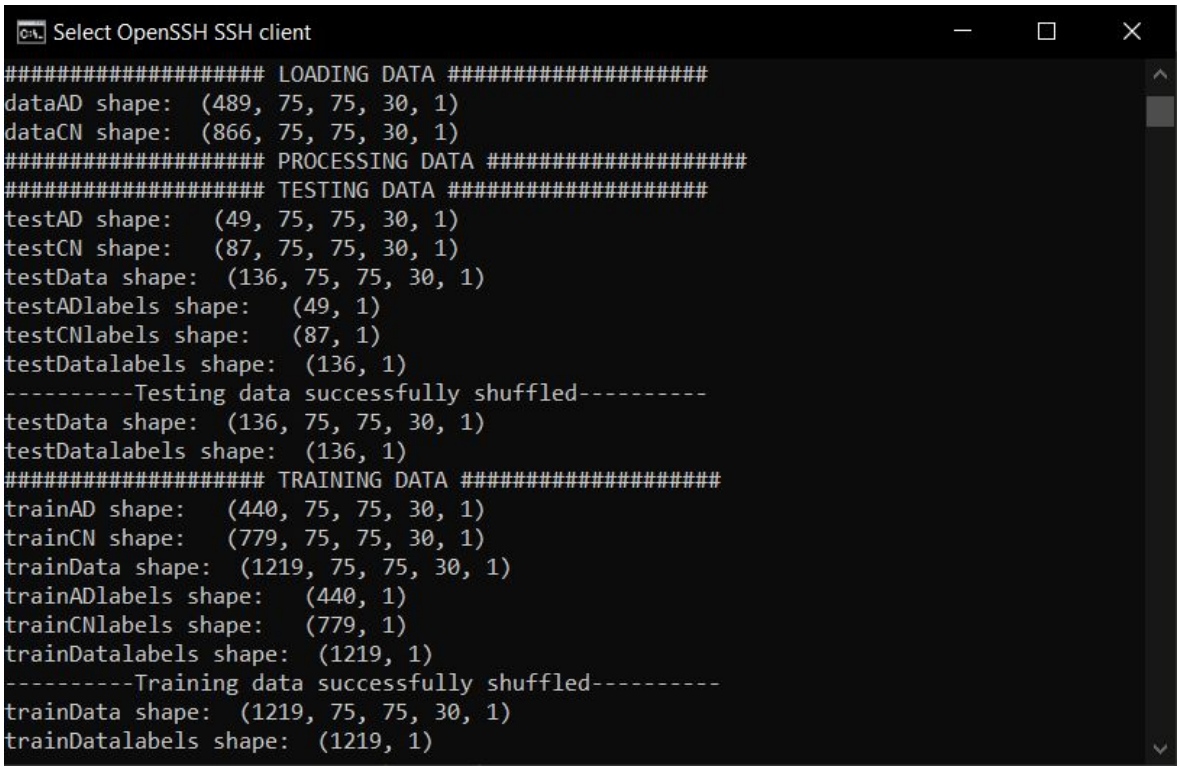


Figure 4.17: 3D binary classification - 3D data preparation terminal output example

As in the previous section, the first set of experiments were with the balanced dataset. As expected, the accuracy results increased significantly when changing into the imbalanced dataset, once again. CV average accuracy of 79.78% was obtained.

Next, the impact of the learning rate, the batch size and the training epochs were studied. The intuition behind the reduction of the default value 0.01 of the learning rate to the lower value of 0.001 was made after the observation that the lowest scoring iterations from the previous k-fold experiments seemed to get stuck on local optimas. Some of the more relevant results obtained from this trial can be observed in Table 4.4. As seen in the table, decreasing the learning rate had a very positive impact on the standard deviation. The reduction of the standard deviation means less variation in the CV accuracy and subsequently improved the average CV accuracy from 79.78% (for learning rate of 0.01) to around 82.43% (for earning rate of 0.001).

Until this point, all experiments were performed with 100x100x40 voxels resized data, obtained through the 3D pre-processing mentioned in the previous section. The next important hyper parameter to vary was the 3D image resolution. We have tested smaller (75x75x30 voxels) and larger (125x125x50 voxels) resolutions. After successfully generating the new sized datasets, the model was trained and its performance assessed with the corresponding results being compared to the ones produced by the previous 100x100x40 voxels resolution. Table 4.5 displays the results.

Architecture	Learning Rate	Epochs	Batch Size	Cross-Validation Average Accuracy	Standard Deviation
Custom 0	0.01 (Default)	30	4	78.62%	7.07%
Custom 0	0.01 (Default)	25	8	79.78%	3.80%
Custom 0	0.01 (Default)	25	16	78.34%	5.72%
Custom 1	0.001	25	4	81.98%	2.30%
Custom 1	0.001	30	8	82.43%	3.01%
Custom 1	0.001	35	4	82.39%	4.55%
Note: Dataset A-Imbalanced, 1355 total samples (489 for AD and 866 for CN) 100x100x40 voxels, SGD optimizer					

Table 4.4: 3D Binary Classification - Learning Rate Comparison

Data Resolution (Voxels)	Trainable Parameters	Epochs	Batch Size	Cross-Validation Average Accuracy	Standard Deviation
75x75x30	3,763,569	30	2	84.44%	3.92%
75x75x30	3,763,569	35	2	85.01%	3.53%
75x75x30	3,763,569	35	4	84.29%	2.75%
100x100x40	13,954,417	25	4	81.98%	2.30%
100x100x40	13,954,417	30	8	82.43%	3.01%
100x100x40	13,954,417	35	4	82.39%	4.55%
125x125x50	28,978,545	20	8	79.11%	4.71%
125x125x50	28,978,545	25	2	78.54%	3.42%
125x125x50	28,978,545	25	4	79.85%	2.99%
Note: Dataset A-Imbalanced, 1355 total samples (489 for AD and 866 for CN) SGD optimizer, 0.001 learning rate					

Table 4.5: 3D Binary Classification - Resolution Comparison

By analyzing these results, it is clear that the 75x75x30 voxels variation outperformed its larger counterparts. It seems a bit surprising that the resolution that brings more information is in fact less favourable for the classification task. However, it seems that the important information can still be correctly discerned in the smaller variations while the irrelevant data is significantly diminished, giving more emphasis to the more essential data sections. Reducing the resolution to 50x50x20 voxels did not bring improvements, therefore this option was not further explored. Since the 75x75x30 voxels resolution improved the average CV accuracy from 82.43% to approximately 85.01%, it was considered as the most efficient and was used in the next experiments.

After performed adjustments to the dataset balance, learning rate and data shape, now the 3D architecture will be optimised from scratch. Up until this point, the base custom architecture (coined Custom 1) consists of: 2 pairs of 3D Conv layers (each pair with 16 filters on the first one and 8 filters on the second one) and a 3x3x3 kernel size, two 3D max-pooling layers with 2x2x2 pool size after each Conv layer pair, one batch normalization layer, one flattening layer, 3 Fully Connected (FC) layers with 2 dropout layers in between them with a 40% dropout rate. The first FC layer consists of 512 neurons, the second has 128 neurons and the last one has one neuron with sigmoid activation function, to reflect the binary classification problem. The code used to implement the Custom 1 model can be seen in Figure 4.18.

```
print("#################### NETWORK ARCHITECTURE ####################")

input_format = Input((75, 75, 30, 1))

conv1 = Conv3D(filters=16, kernel_size=(3, 3, 3), activation='relu')(input_format)
conv2 = Conv3D(filters=8, kernel_size=(3, 3, 3), activation='relu')(conv1)

max_pool1 = MaxPool3D(pool_size=(2, 2, 2))(conv2)

conv3 = Conv3D(filters=16, kernel_size=(3, 3, 3), activation='relu')(max_pool1)
conv4 = Conv3D(filters=8, kernel_size=(3, 3, 3), activation='relu')(conv3)

max_pool2 = MaxPool3D(pool_size=(2, 2, 2))(conv4)

norm1 = BatchNormalization()(max_pool2)
flat1 = Flatten()(norm1)

fc1 = Dense(units=512, activation='relu')(flat1)

drop1 = Dropout(0.4)(fc1)

fc2 = Dense(units=128, activation='relu')(drop1)

drop2 = Dropout(0.4)(fc2)

fc3_out = Dense(units=1, activation='sigmoid')(drop2)
```

Figure 4.18: 3D binary classification - Base custom 3D-CNN architecture

Table 4.6 shows the results obtained from 3 additional custom architectures and compares them to the Custom 1 model. The variations basically consist in changing the number of Conv filters (8 or 16 filters in the first or second pair of Conv layers). After analyzing these results, it is possible to verify that some improvements have been achieved, although the number of trainable parameters increased significantly (almost doubled in some cases). The maximum cross-validation average accuracy score obtained had now gone from the previous 85.01%, with the Custom 1 architecture, to 86.80% with the custom 4 architecture.

Architecture	Trainable Parameters	Epochs	Batch Size	Cross-Validation Average Accuracy	Standard Deviation
Custom 1 (16/8/16/8)	3,763,569	30	2	84.44%	3.92%
Custom 1 (16/8/16/8)	3,763,569	35	2	85.01%	3.53%
Custom 1 (16/8/16/8)	3,763,569	35	4	84.29%	2.75%
Custom 2 (8/16/8/16)	7,449,769	35	2	85.91%	1.48%
Custom 2 (8/16/8/16)	7,449,769	40	4	85.67%	2.63%
Custom 2 (8/16/8/16)	7,449,769	40	8	84.65%	3.18%
Custom 3 (16/16/8/8)	3,765,297	35	2	86.18%	3.93%
Custom 3 (16/16/8/8)	3,765,297	40	2	85.08%	2.85%
Custom 3 (16/16/8/8)	3,765,297	40	4	84.81%	3.98%
Custom 4 (8/8/16/16)	7,451,497	25	2	84.87%	2.71%
Custom 4 (8/8/16/16)	7,451,497	35	2	86.80%	4.70%
Custom 4 (8/8/16/16)	7,451,497	40	4	85.98%	3.32%
Note: Dataset A-Imbalanced, 1355 total samples (489 for AD and 866 for CN) 75x75x30 voxels, SGD optimizer, 0.001 learning rate					

Table 4.6: 3D Binary Classification - Custom Architecture Comparison

4.3.3 Train-CV-Test Results

In order to create a viable comparison between the 2D and 3D approaches, the train-CV-test performance was evaluated again with the same 105 independent testing samples as in the 2D approach. The dropout rate was also adjusted from 40% to 50%, as this value was the most efficient one during the 2D architecture study. The four Custom 3D architectures are compared in Figure 4.19. As seen in the figure, the Custom 4 model outperforms the other models and it will be used in the next experiments.

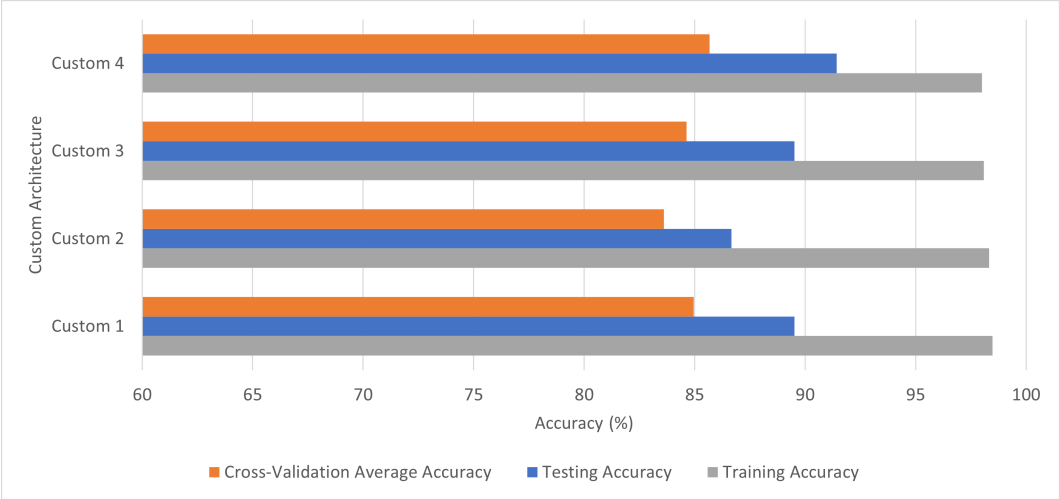


Figure 4.19: 3D binary classification - Custom variations comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; 50% dropout rate; batch size = 2)

Figure 4.20 depicts the train-CV-test accuracy results for varying learning rates with respect to the Custom 4 model. The results obtained confirmed that the 0.001 learning rate provided the best scores out of the four options tested, even though the 0.0005 option had basically the same level of results.

The performance of the Custom 4 model when the resolution of the input 3D image varies is plotted in Figure 4.21. Reducing the sample's dimension can be interpreted as noise reduction and therefore the performance has improved.

Besides these studies, an experiment was also conducted in order to assess the impact that the batch size might incur on the classification accuracy (Figure 4.22). The results obtained seem to go along with the majority of the top results that were previously achieved, with the most effective being a batch size of 2, even though the change in the cross-validation average accuracy is not that significantly increased.

In conclusion, the best results obtained with the 3D binary classification model were provided by a custom architecture built from scratch. The model was trained on the full dataset that was available (Figure 4.19), with the SGD optimizer, learning rate of 0.001 (Figure 4.20) and 75x75x30 voxels sample (Figure 4.21). The model achieved CV average accuracy of around 86.80% (Dataset A-Imbalanced) and a testing accuracy of 91.43% (Dataset B).

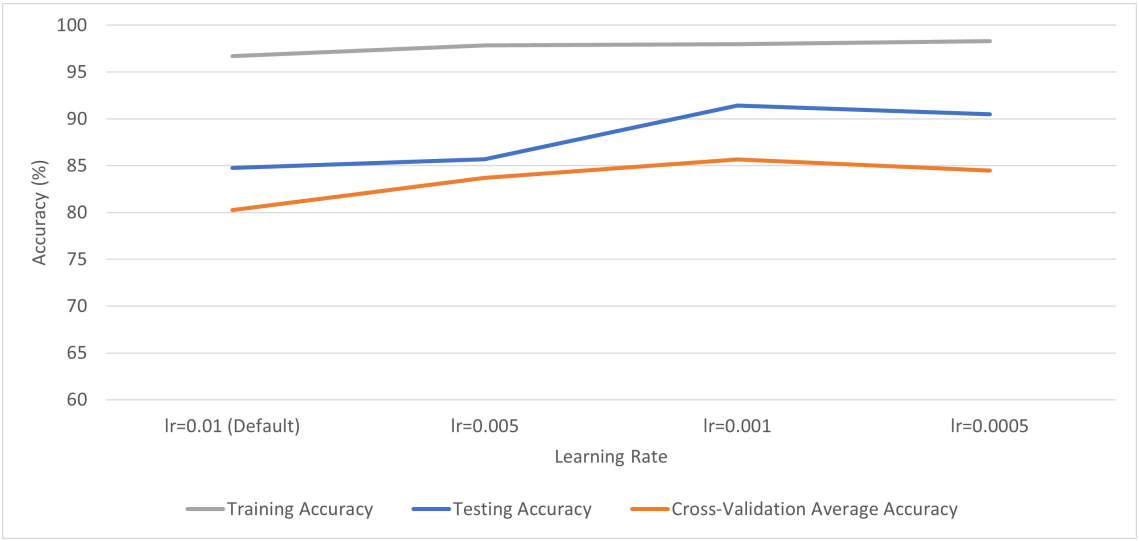


Figure 4.20: 3D binary classification - Learning rate comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; custom 4 architecture; 50% dropout rate; batch size = 2)

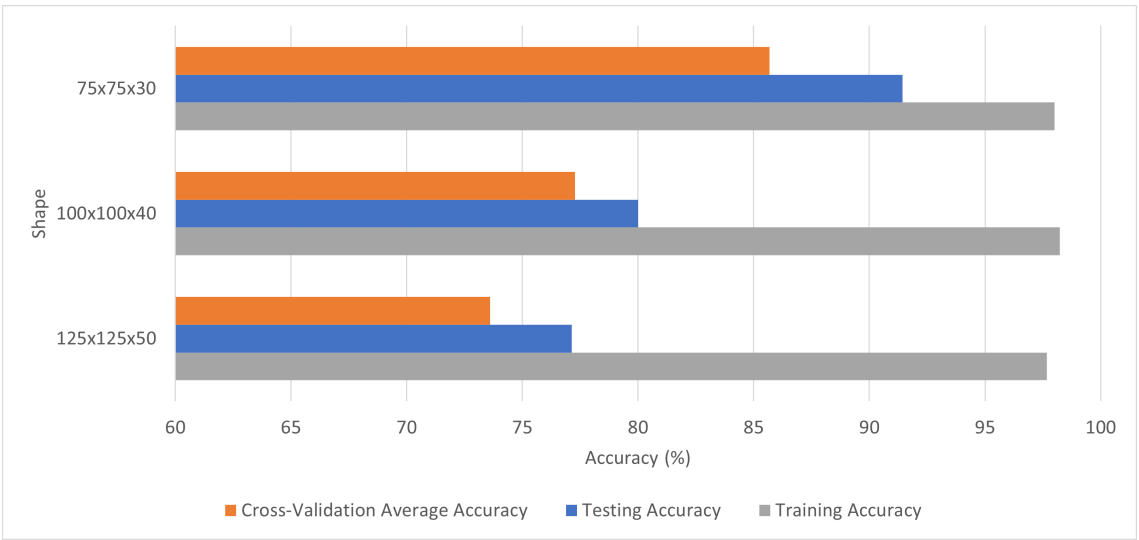


Figure 4.21: 3D binary classification - Resolution comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; SGD optimizer; 0.001 learning rate; custom 4 architecture; 50% dropout rate; batch size = 2)

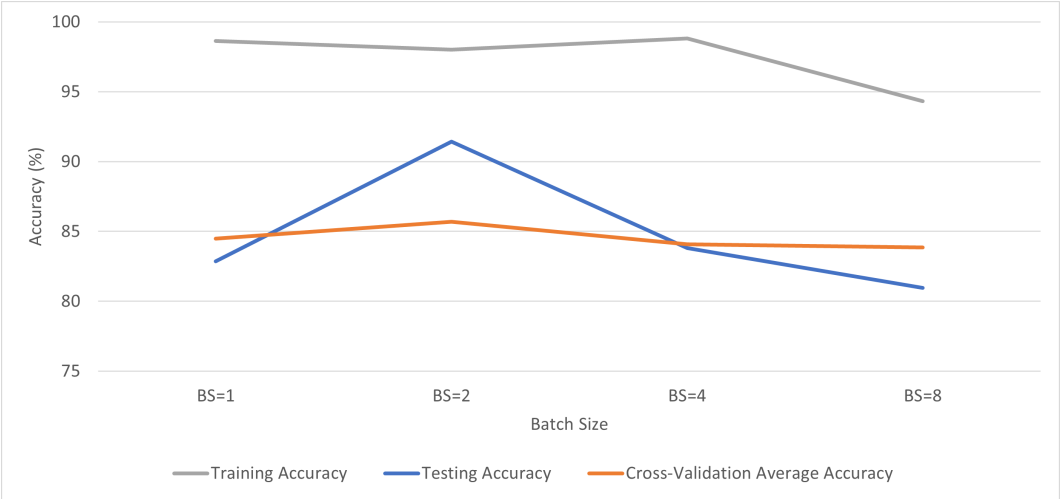


Figure 4.22: 3D binary classification - Batch size comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; 50% dropout rate)

4.3.4 Dropout Analysis

The overfitting issue was tackled through the variation of the dropout rate. As with the 2D approach, the 50% dropout rate provided better overall results. It did not make substantial impact on the CV average accuracy metric but the testing accuracy score had a significantly greater peak (Figure 4.23).

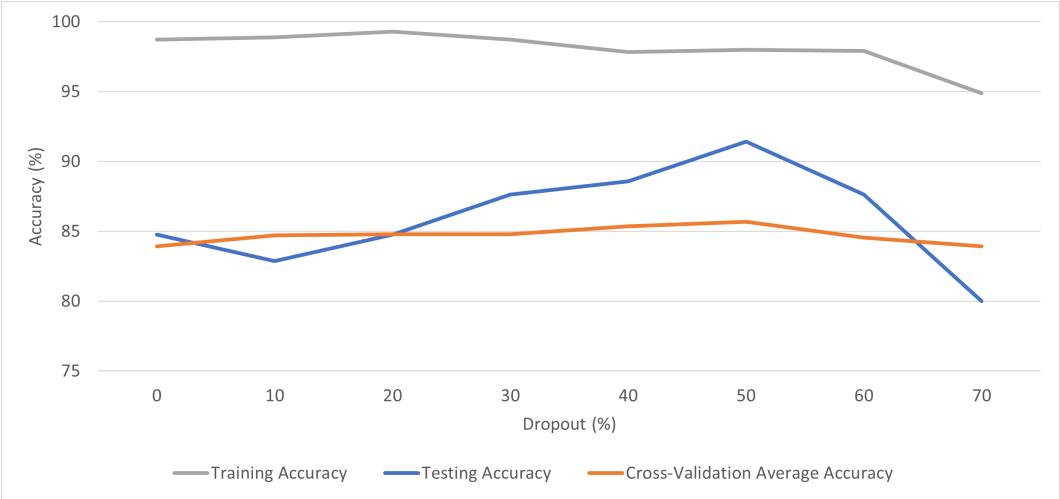


Figure 4.23: 3D binary classification - Dropout rates comparison (Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2)

The data processed by this 3D approach was more complex in terms of its structure, compared to the images that were processed by the 2D slice-level model. Due to this reason, the convergence of the training accuracy and the loss function are less stable as shown in Figure 4.24 and Figure 4.25.

It is also possible to observe that for higher dropout rates, the model had a really hard time converging. In some cases, like the 70% dropout rate, it still did not converge after 90 epochs. This might be due to the fact that this network's architecture was assembled from scratch, and therefore was not fully optimized for the particular problem.

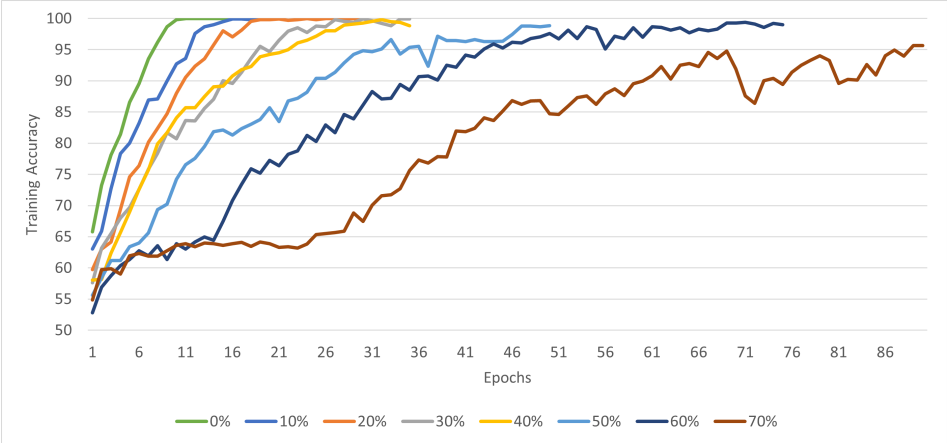


Figure 4.24: 3D binary classification - Training accuracy per epoch (Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2)

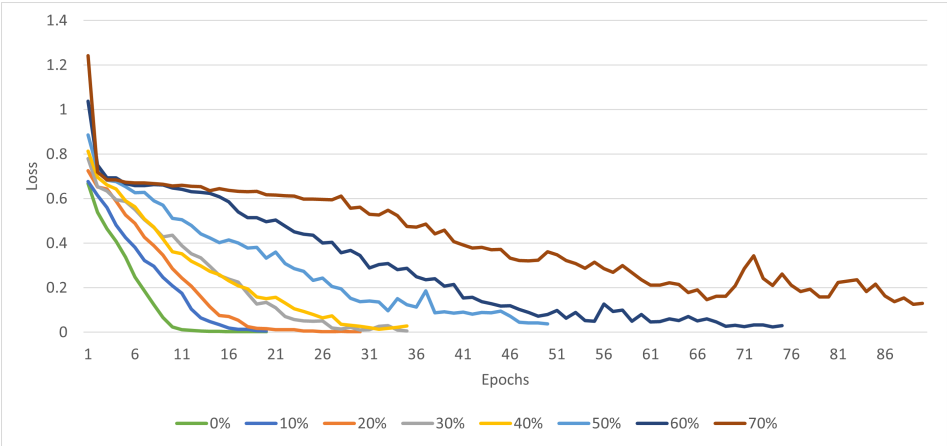


Figure 4.25: 3D binary classification - Loss per epoch during training (Dropout trial - Dropout trial - Dataset B: 125 samples per cross-validation fold [80 CN/45 AD] and 105 samples for testing [66 CN/39 AD]; 75x75x30 voxels; SGD optimizer; 0.001 learning rate; custom 4 architecture; batch size = 2)

Chapter 5

Conclusions

This dissertation addressed the problem of diagnosing the Alzheimer's disease based on deep learning techniques. The primary objective was to study the potential of ^{18}F-FDG PET neuroimaging as a AD biomarker for classifying healthy versus AD patients. The comparison between a 2D model based on transfer learning and a custom developed 3D-CNN trained from scratch has become the central focus of this work. In addition, the study evaluates the effectiveness of different techniques implemented in order to prevent the occurrence of overfitting.

A custom Inception V3 model was evaluated by making use of the 2D Inception V3 model and a 2D PET scan slice collage technique. In this case, for a neural network with approximately 12 million trainable parameters, the best cross-validation average accuracy obtained was around 83.62%, and 91.43% for the best testing accuracy (105 subjects). This approach confirmed that the transfer learning drastically increases the model's performance, even when the dataset differs in many aspects from the original one. The final results still benefited form other factors, such as the implementation of dropout and the reduction on the model's complexity through the use of the Inception V3's sub-networks. Together, these were the aspects allowing a considerable decrease in the model's overfitting.

For the 3D alternative, a custom 3D-CNN architecture was developed in the context of a subject-level approach. This approach reached a maximum cross-validation average accuracy of 86.80% (a 3.18% increase over the 2D version) and the same 91.43% best testing accuracy. This means that on both best-case scenarios, the models predicted correctly 96 cases, out of those 105 total, which led to the same 91.43%. From a comparative point of view, it is relevant to highlight the 7.5 million trainable parameters of the 3D, which is a significant decrease when compared to the 12 million used in the 2D approach. Furthermore, it is worth mention that other custom variant models reached similar results (86.18% - custom 3), while only using around 3.75 million trainable parameters. Techniques such as dropout and complexity reduction were also used in order to reduce the effects of overfitting. Additionally, complexity reduction was achieved through the modification of the samples' resolution.

Overall, this thesis has allowed for a considerable analysis of CNN performance through a comparison between 2D and 3D models. The results are promising and show the usefulness of using ^{18}F-FDG PET scans as a viable medical imaging modality in the realm of deep learning applications. Despite the fact that the data available for this experiment was quite limited, which inevitably leads to some amount of overfitting, the 3D-CNN presented slightly higher results on the same dataset.

Listed below are some possible future endeavors that could serve as a continuation of this dissertation's work:

- `Other three-dimensional approaches`: As was previously detailed on chapter 2, there are several different approaches that could have been implemented instead of the three-dimensional subject-level or the two-dimensional slice-level approach taken during this project. Maybe something like a ROI-based approach would provide some interesting results provided a reasonably efficient region of interest could be delineated on the corresponding subject's brain scans.

- `Additional subject information`: Some additional types of data could also be implemented into the model in order to try and improve the network's performance. Information like the subject's age, gender, previous medical records, and so on, could turn out to be useful tools when it comes to improving the classifier's accuracy.

- `Different classification`: Although an AD vs CN binary classification might seem like the most clear cut classification option for this problem, there are some other class pairs that would provide extremely valuable information if they could be successfully predicted. For example, classifying subjects between sMCI (Stable Mild Cognitive Impairment) and pMCI (Progressive Mild Cognitive Impairment), the latter which would eventually develop into AD, would be a great medical support tool if properly developed. Following that same logic, there is also the option to create a multi-class classifier instead of a binary one. CN vs MCI (Mild Cognitive Impairment) vs AD could be a useful variation for instance.

- `Ensemble model`: CNN ensemble methods could also be applied to this problem. Simply put, ensemble methods allow the user to combine multiple CNNs, and its corresponding outputs, in order to improve the prediction's accuracy [123]. So, for example, an ensemble model that combines the predictions from both models developed during this thesis would theoretically increase the accuracy of the prediction when compared to the ones obtained from just a single one of those models. An ensemble model could also be implemented in order to combine two types of the same model, but with the input being two different medical imaging modalities (e.g., an ^{18}F-FDG PET scan on one model, and a MRI scan on the other).

References

[1] EU Joint Programme Neurodegenerative Disease Research. Why choose neurodegenerative diseases? https://www.neurodegenerationresearch.eu/why/, 2019.

[2] Zewen Li, Wenjie Yang, Shouheng Peng, and Fan Liu. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, 2020.

[3] Cryer D Jonathan;Chan Sik Kung. Time series analysis With Apllication in R. 2008.

[4] Milind Rane, Aseem Patil, and Bhushan Barse. Real Object Detection Using TensorFlow. In Lecture Notes in Electrical Engineering, 2020.

[5] Michael Nielsen. Neural Networks and Deep Learning. Artificial Intelligence, pages 389–411, 2018.

[6] Aseem Patil and Milind Rane. Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition. In Smart Innovation, Systems and Technologies, 2021.

[7] Jue Zhang, Kun Chen, Di Wang, Fei Gao, Yijia Zheng, and Mei Yang. Editorial: Advances of Neuroimaging and Data Analysis, 2020.

[8] Charles Marcus, Esther Mena, and Rathan M. Subramaniam. Brain PET in the diagnosis of Alzheimer's disease, 2014.

[9] Philip Scheltens, Kaj Blennow, Monique M.B. Breteler, Bart de Strooper, Giovanni B. Frisoni, Stephen Salloway, and Wiesje Maria Van der Flier. Alzheimer's disease, 2016.

[10] Alzheimer's Association. Research And Progress: Earlier Diagnosis. https://www.alz.org/alzheimers-dementia/research_progress/earlier-diagnosis, 2021.

[11] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis, 2017.

[12] Yiming Ding, Jae Ho Sohn, Michael G. Kawczynski, Hari Trivedi, Roy Harnish, Nathaniel W. Jenkins, Dmytro Lituiev, Timothy P. Copeland, Mariam S. Aboian, Carina Mari Aparici, Spencer C. Behr, Robert R. Flavell, Shih Ying Huang, Kelly A. Zalocusky, Lorenzo Nardo, Youngho Seo, Randall A. Hawkins, Miguel Hernandez Pampaloni, Dexter Hadley, and Benjamin L. Franc. A deep learning model to predict a diagnosis of Alzheimer disease by using 18 F-FDG PET of the brain. Radiology, 2019.

[13] Donghuan Lu, Karteek Popuri, Gavin Weiguang Ding, Rakesh Balachandar, and Mirza Faisal Beg. Multiscale deep neural network based analysis of FDG-PET images for the early diagnosis of Alzheimer's disease. Medical Image Analysis, 2018.

[14] Yechong Huang, Jiahang Xu, Yuncheng Zhou, Tong Tong, and Xiahai Zhuang. Diagnosis of Alzheimer's disease via multi-modality 3D convolutional neural network. Frontiers in Neuroscience, 2019.

[15] Manhua Liu, Danni Cheng, Kundong Wang, and Yaping Wang. Multi-Modality Cascaded Convolutional Neural Networks for Alzheimer's Disease Diagnosis. Neuroinformatics, 2018.

[16] Jun Shi, Xiao Zheng, Yan Li, Qi Zhang, and Shihui Ying. Multimodal Neuroimaging Feature Learning with Multimodal Stacked Deep Polynomial Networks for Diagnosis of Alzheimer's Disease. IEEE Journal of Biomedical and Health Informatics, 2018.

[17] Serge Przedborski, Miquel Vila, and Vernice Jackson-Lewis. Series Introduction: Neurodegeneration: What is it and where are we? Journal of Clinical Investigation, 2003.

[18] William Dauer and Serge Przedborski. Parkinson's disease: Mechanisms and models, 2003.

[19] David S. Geldmacher and Peter J. Whitehouse. Evaluation of Dementia. New England Journal of Medicine, 1996.

[20] 2020 Alzheimer's disease facts and figures. Alzheimer's and Dementia, 2020.

[21] World Health Organization. Dementia: number of people affected to triple in next 30 years. `https://www.who.int/news/item/07-12-2017-dementia-number-of-people-affected-to-triple-in-next-30-years`, 2017.

[22] Jose Viña and Ana Lloret. Why women have more Alzheimer's disease than men: Gender and mitochondrial toxicity of amyloid-β peptide, 2010.

[23] Alireza Atri. Current and Future Treatments in Alzheimer's Disease. Seminars in Neurology, 2019.

[24] Chris Fox, Louise Lafortune, Malaz Boustani, and Carol Brayne. Debate & Analysis The pros and cons of early diagnosis in dementia, 2013.

[25] J. Wang, T. Yang, P. Thompson, and J. Ye. Sparse models for imaging genetics. In Machine Learning and Medical Imaging. 2016.

[26] Jessica Wilson. Alzheimer's Disease. `https://www.sciencephoto.com/media/447399/view/alzheimer-s-disease`, 2020.

[27] Jeffrey R. Petrella, R. Edward Coleman, and P. Murali Doraiswamy. Neuroimaging and early diagnosis of alzheimer disease: A look to the future, 2003.

[28] Sascha Gill, Pauline Mouches, Sophie Hu, Deepthi Rajashekar, Frank P. MacMaster, Eric E. Smith, Nils D. Forkert, and Zahinoor Ismail. Using Machine Learning to Predict Dementia from Neuropsychiatric Symptom and Neuroimaging Data. Journal of Alzheimer's Disease, 2020.

[29] Stefan Klöppel, Cynthia M. Stonnington, Carlton Chu, Bogdan Draganski, Rachael I. Scahill, Jonathan D. Rohrer, Nick C. Fox, Clifford R. Jack, John Ashburner, and Richard S.J. Frackowiak. Automatic classification of MR scans in Alzheimer's disease. Brain, 2008.

[30] Mohammad R. Arbabshirani, Sergey Plis, Jing Sui, and Vince D. Calhoun. Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls. NeuroImage, 2017.

[31] Felp Roza. End-to-end learning, the (almost) every purpose ML method. `https://towardsdatascience.com/e2e-the-every-purpose-ml-method-5d4f20dafee4`, 2019.

[32] ADNI. Alzheimer's Disease Neuroimaging Initiative. `http://adni.loni.usc.edu/`, 2017.

[33] Ben Schmand, Piet Eikelenboom, and Willem A. Van Gool. Value of neuropsychological tests, neuroimaging, and biomarkers for diagnosing Alzheimer's disease in younger and older age cohorts. Journal of the American Geriatrics Society, 2011.

[34] Gil D. Rabinovici. Late-onset Alzheimer disease, 2019.

[35] Ellis Niemantsverdriet, Sara Valckx, Maria Bjerke, and Sebastiaan Engelborghs. Alzheimer's disease CSF biomarkers: clinical indications and rational use, 2017.

[36] Jaeho Kim, Yuhyun Park, Seongbeom Park, Hyemin Jang, Hee Jin Kim, Duk L. Na, Hyejoo Lee, and Sang Won Seo. Prediction of tau accumulation in prodromal Alzheimer's disease using an ensemble machine learning approach. Scientific Reports, 2021.

[37] Juan Alvarez-Linera. 3 T MRI: Advances in brain imaging. European Journal of Radiology, 2008.

[38] Valentina Berti, Alberto Pupi, and Lisa Mosconi. PET/CT in diagnosis of dementia. Annals of the New York Academy of Sciences, 2011.

[39] Peter Lam. What to know about MRI scans. `https://www.medicalnewstoday.com/articles/146309`, 2018.

[40] Gonzalo Miranda Benítez. Técnico Superior en Radioterapia Y Dosimetría. `https://docplayer.es/66550603-Tecnico-superior-en-radioterapia-y-dosimetria.html`, 2018.

[41] Ronald C. Petersen. Aging, mild cognitive impairment, and Alzheimer's disease. Neurologic Clinics, 2000.

[42] P. N. Wang, J. F. Lirng, K. N. Lin, F. C. Chang, and H. C. Liu. Prediction of Alzheimer's disease in mild cognitive impairment: A prospective study in Taiwan. Neurobiology of Aging, 2006.

[43] Jason P. Lerch, Jens Pruessner, Alex P. Zijdenbos, D. Louis Collins, Stefan J. Teipel, Harald Hampel, and Alan C. Evans. Automated cortical thickness measurements from MRI can accurately separate Alzheimer's patients from normal elderly controls. Neurobiology of Aging, 2008.

[44] European Society of Radiology. Brainwatch: Detecting and Diagnosing Brain Diseases with Medical Imaging. `https://www.esnr.org/en/news/download-the-book-on-brain-imaging-08-11-2014/`, 2014.

[45] Daniela Perani, Pasquale Anthony Della Rosa, Chiara Cerami, Francesca Gallivanone, Federico Fallanca, Emilia Giovanna Vanoli, Andrea Panzacchi, Flavio Nobili, Sabina Pappatà, Alessandra Marcone, Valentina Garibotto, Isabella Castiglioni, Giuseppe Magnani, Stefano F. Cappa, Luigi Gianolli, Alexander Drzezga, Robert Perneczky, Mira Didic, Eric Guedj, Bart N. Van Berckel, Rik Ossenkoppele, Silvia Morbelli, Giovanni Frisoni, and Anna Caroli. Validation of an optimized SPM procedure for FDG-PET in dementia diagnosis in a clinical setting. NeuroImage: Clinical, 2014.

[46] Gaël Chételat, Javier Arbizu, Henryk Barthel, Valentina Garibotto, Ian Law, Silvia Morbelli, Elsmarieke van de Giessen, Federica Agosta, Frederik Barkhof, David J. Brooks, Maria C. Carrillo, Bruno Dubois, Anders M. Fjell, Giovanni B. Frisoni, Oskar Hansson, Karl Herholz, Brian F. Hutton, Clifford R. Jack, Adriaan A. Lammertsma, Susan M. Landau, Satoshi Minoshima, Flavio Nobili, Agneta Nordberg, Rik Ossenkoppele, Wim J.G. Oyen, Daniela Perani, Gil D. Rabinovici, Philip Scheltens, Victor L. Villemagne, Henrik Zetterberg, and Alexander Drzezga. Amyloid-PET and 18F-FDG-PET in the diagnostic investigation of Alzheimer's disease and other dementias, 2020.

[47] Yu Kyeong Kim, Dong Soo Lee, Sang Kun Lee, Chun Kee Chung, June Key Chung, and Myung Chul Lee. 18F-FDG PET in localization of frontal lobe epilepsy: Comparison of visual and SPM analysis. Journal of Nuclear Medicine, 2002.

[48] Kyum Yil Kwon, Choong G. Choi, Jae S. Kim, Myoung C. Lee, and Sun Ju Chung. Comparison of brain MRI and 18F-FDG PET in the differential diagnosis of multiple system atrophy from Parkinson's disease. Movement Disorders, 2007.

[49] Tim Van den Wyngaert, Stijn De Schepper, and Laurens Carp. Quality Assessment in FDG-PET/CT Imaging of Head-and-Neck Cancer: One Home Run Is Better Than Two Doubles, 2020.

[50] Dimitrios Kapogiannis and Mark P. Mattson. Disrupted energy metabolism and neuronal circuit dysfunction in cognitive impairment and Alzheimer's disease, 2011.

[51] Richard K.J. Brown, Nicolaas I. Bohnen, Ka Kit Wong, Satoshi Minoshima, and Kirk A. Frey. Brain PET in suspected dementia: Patterns of altered FDG metabolism. Radiographics, 2014.

[52] Juergen Dukart, Ferath Kherif, Karsten Mueller, Stanislaw Adaszewski, Matthias L. Schroeter, Richard S.J. Frackowiak, and Bogdan Draganski. Generative FDG-PET and MRI Model of Aging and Disease Progression in Alzheimer's Disease. PLoS Computational Biology, 2013.

[53] Ciprian Catana, Alexander R. Guimaraes, and Bruce R. Rosen. PET and MR imaging: The odd couple or a match made in heaven?, 2013.

[54] Hossein Jadvar and Patrick M. Colletti. Competitive advantage of PET/MRI, 2014.

[55] Hayit Greenspan, Bram Van Ginneken, and Ronald M. Summers. Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique, 2016.

[56] Zilong Hu, Jinshan Tang, Ziming Wang, Kai Zhang, Lin Zhang, and Qingling Sun. Deep learning for image-based cancer detection and diagnosisA survey. Pattern Recognition, 2018.

[57] Xinggang Wang, Wei Yang, Jeffrey Weinreb, Juan Han, Qiubai Li, Xiangchuang Kong, Yongluan Yan, Zan Ke, Bo Luo, Tao Liu, and Liang Wang. Searching for prostate cancer by fully automated magnetic resonance imaging classification: Deep learning versus non-deep learning. Scientific Reports, 2017.

[58] Kenny H. Cha, Lubomir Hadjiiski, Heang Ping Chan, Alon Z. Weizer, Ajjai Alva, Richard H. Cohan, Elaine M. Caoili, Chintana Paramagul, and Ravi K. Samala. Bladder Cancer Treatment Response Assessment in CT using Radiomics with Deep-Learning. Scientific Reports, 2017.

[59] Koichiro Yasaka, Hiroyuki Akai, Osamu Abe, and Shigeru Kiryu. Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: A preliminary study. Radiology, 2018.

[60] Thijs Kooi, Bram van Ginneken, Nico Karssemeijer, and Ard den Heeten. Discriminating solitary cysts from soft tissue lesions in mammography using a pretrained deep convolutional neural network. Medical physics, 2017.

[61] Seokmin Han, Ho Kyung Kang, Ja Yeon Jeong, Moon Ho Park, Wonsik Kim, Won Chul Bang, and Yeong Kyeong Seong. A deep learning framework for supporting the classification of breast lesions in ultrasound images. Physics in Medicine and Biology, 2017.

[62] Hongkai Wang, Zongwei Zhou, Yingci Li, Zhonghua Chen, Peiou Lu, Wenzhi Wang, Wanyu Liu, and Lijuan Yu. Comparison of machine learning methods for classifying mediastinal lymph node metastasis of non-small cell lung cancer from 18F-FDG PET/CT images. EJNMMI Research, 2017.

[63] Paras Lakhani and Baskaran Sundaram. Deep learning at chest radiography: Automated classification of pulmonary tuberculosis by using convolutional neural networks. Radiology, 2017.

[64] U. K. Lopes and J. F. Valiati. Pre-trained convolutional neural networks as feature extractors for tuberculosis detection. Computers in Biology and Medicine, 2017.

[65] Huafeng Wang, Tingting Zhao, Lihong Connie Li, Haixia Pan, Wanquan Liu, Haoqi Gao, Fangfang Han, Yuehai Wang, Yifan Qi, and Zhengrong Liang. A hybrid CNN feature model for pulmonary nodule malignancy risk differentiation. Journal of X-Ray Science and Technology, 2018.

[66] Francesco Ciompi, Kaman Chung, Sarah J. Van Riel, Arnaud Arindra Adiyoso Setio, Paul K. Gerke, Colin Jacobs, Ernst Th Scholten, Cornelia Schaefer-Prokop, Mathilde M.W. Wille, Alfonso Marchianò, Ugo Pastorino, Mathias Prokop, and Bram Van Ginneken. Towards automatic pulmonary nodule management in lung cancer screening with deep learning. Scientific Reports, 2017.

[67] Qing Zeng Song, Lei Zhao, Xing Ke Luo, and Xue Chen Dou. Using Deep Learning for Classification of Lung Nodules on Computed Tomography Images. Journal of Healthcare Engineering, 2017.

[68] Jakub Olczak, Niklas Fahlberg, Atsuto Maki, Ali Sharif Razavian, Anthony Jilert, André Stark, Olof Sköldenberg, and Max Gordon. Artificial intelligence for analyzing orthopedic trauma radiographs: Deep learning algorithms—are they on par with humans for diagnosing fractures? Acta Orthopaedica, 2017.

[69] Yanping Xue, Rongguo Zhang, Yufeng Deng, Kuan Chen, and Tao Jiang. A preliminary examination of the diagnostic value of deep learning in hip osteoarthritis. PLoS ONE, 2017.

[70] David B. Larson, Matthew C. Chen, Matthew P. Lungren, Safwan S. Halabi, Nicholas V. Stence, and Curtis P. Langlotz. Performance of a deep-learning neural network model in assessing skeletal maturity on pediatric hand radiographs. Radiology, 2018.

[71] Hyunkwang Lee, Shahein Tajmir, Jenny Lee, Maurice Zissen, Bethel Ayele Yeshiwas, Tarik K. Alkasab, Garry Choy, and Synho Do. Fully Automated Deep Learning System for Bone Age Assessment. Journal of Digital Imaging, 2017.

[72] C. Spampinato, S. Palazzo, D. Giordano, M. Aldinucci, and R. Leonardi. Deep learning for automated skeletal bone age assessment in X-ray images. Medical Image Analysis, 2017.

[73] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. Nature, 2017.

[74] Justin Ker, Lipo Wang, Jai Rao, and Tchoyoson Lim. Deep Learning Applications in Medical Image Analysis. IEEE Access, 2017.

[75] Dinggang Shen, Guorong Wu, and Heung Il Suk. Deep Learning in Medical Image Analysis. Annual Review of Biomedical Engineering, 2017.

[76] Shelly Soffer, Avi Ben-Cohen, Orit Shimon, Michal Marianne Amitai, Hayit Greenspan, and Eyal Klang. Convolutional Neural Networks for Radiologic Images: A Radiologist's Guide, 2019.

[77] Kenji Suzuki. Overview of deep learning in medical imaging, 2017.

[78] Noah Stier, Nicholas Vincent, David Liebeskind, and Fabien Scalzo. Deep learning of tissue fate features in acute ischemic stroke. In Proceedings - 2015 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2015, 2015.

[79] Hongyoon Choi, Seunggyun Ha, Hyung Jun Im, Sun Ha Paek, and Dong Soo Lee. Refining diagnosis of Parkinson's disease with deep learning-based interpretation of dopamine transporter imaging. NeuroImage: Clinical, 2017.

[80] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with Deep Neural Networks. Medical Image Analysis, 2017.

[81] Konstantinos Kamnitsas, Christian Ledig, Virginia F.J. Newcombe, Joanna P. Simpson, Andrew D. Kane, David K. Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Medical Image Analysis, 2017.

[82] James H. Cole, Rudra P.K. Poudel, Dimosthenis Tsagkrasoulis, Matthan W.A. 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. NeuroImage, 2017.

[83] Shui Hua Wang, Yi Ding Lv, Yuxiu Sui, Shuai Liu, Su Jing Wang, and Yu Dong Zhang. Alcoholism Detection by Data Augmentation and Convolutional Neural Network with Stochastic Pooling. Journal of Medical Systems, 2018.

[84] Mr Amir Ebrahimighahnavieh, Suhuai Luo, and Raymond Chiong. Deep learning to detect Alzheimer's disease from neuroimaging: A systematic literature review. Computer Methods and Programs in Biomedicine, 2020.

[85] Arianna Sala, Camilla Caprioglio, Roberto Santangelo, Emilia Giovanna Vanoli, Sandro Iannaccone, Giuseppe Magnani, and Daniela Perani. Brain metabolic signatures across the Alzheimer's disease spectrum. European Journal of Nuclear Medicine and Molecular Imaging, 2020.

[86] Hongyoon Choi and Kyong Hwan Jin. Predicting cognitive decline with deep learning of brain metabolism and amyloid imaging. Behavioural Brain Research, 2018.

[87] Arjun Punjabi, Adam Martersteck, Yanran Wang, Todd B. Parrish, and Aggelos K. Katsaggelos. Neuroimaging modality fusion in Alzheimer's classification using convolutional neural networks. PLoS ONE, 2019.

[88] Junhao Wen, Elina Thibeau-Sutre, Mauricio Diaz-Melo, Jorge Samper-González, Alexandre Routier, Simona Bottani, Didier Dormont, Stanley Durrleman, Ninon Burgos, and Olivier Colliot. Convolutional neural networks for classification of Alzheimer's disease: Overview and reproducible evaluation. Medical Image Analysis, 2020.

[89] Shui Hua Wang, Preetha Phillips, Yuxiu Sui, Bin Liu, Ming Yang, and Hong Cheng. Classification of Alzheimer's Disease Based on Eight-Layer Convolutional Neural Network with Leaky Rectified Linear Unit and Max Pooling. Journal of medical systems, 2018.

[90] Arwa Mohammed Taqi, Ahmed Awad, Fadwa Al-Azzo, and Mariofanna Milanova. The Impact of Multi-Optimizers and Data Augmentation on TensorFlow Convolutional Neural Network Performance. In Proceedings - IEEE 1st Conference on Multimedia Information Processing and Retrieval, MIPR 2018, 2018.

[91] Shangran Qiu, Gary H. Chang, Marcello Panagia, Deepa M. Gopal, Rhoda Au, and Vijaya B. Kolachalama. Fusion of deep learning models of MRI scans, Mini–Mental State Examination, and logical memory test enhances diagnosis of mild cognitive impairment. Alzheimer's and Dementia: Diagnosis, Assessment and Disease Monitoring, 2018.

[92] Aly Valliani and Ameet Soni. Deep residual nets for improved Alzheimer's diagnosis. In ACM-BCB 2017 - Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, 2017.

[93] Mingxia Liu, Jun Zhang, Ehsan Adeli, and Dinggang Shen. Landmark-based deep multi-instance learning for brain disease diagnosis. Medical Image Analysis, 2018.

[94] Fan Li and Manhua Liu. Alzheimer's disease diagnosis based on multiple cluster dense convolutional networks. Computerized Medical Imaging and Graphics, 2018.

[95] Chunfeng Lian, Mingxia Liu, Jun Zhang, and Dinggang Shen. Hierarchical fully convolutional network for joint atrophy localization and Alzheimer's disease diagnosis using structural MRI. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.

[96] Karim Aderghal, Jenny Benois-Pineau, Karim Afdel, and Catheline Gwenaëlle. FuseMe: Classification of sMRI images by fusion of deep CNNs in 2D+e projections. In ACM International Conference Proceeding Series, 2017.

[97] Weiming Lin, Tong Tong, Qinquan Gao, Di Guo, Xiaofeng Du, Yonggui Yang, Gang Guo, Min Xiao, Min Du, and Xiaobo Qu. Convolutional neural networks-based MRI image analysis for the Alzheimer's disease prediction from mild cognitive impairment. Frontiers in Neuroscience, 2018.

[98] Karl Backstrom, Mahmood Nazari, Irene Yu Hua Gu, and Asgeir Store Jakola. An efficient 3D deep convolutional network for Alzheimer's disease diagnosis using MR images. In Proceedings - International Symposium on Biomedical Imaging, 2018.

[99] Ehsan Hosseini Asl, Mohammed Ghazal, Ali Mahmoud, Ali Aslantas, Ahmed Shalaby, Manual Casanova, Gregory Barnes, Georgy Gimel'farb, Robert Keynton, and Ayman El Baz. Alzheimer's disease diagnostics by a 3D deeply supervised adaptable convolutional network. Frontiers in Bioscience - Landmark, 2018.

[100] Upul Senanayake, Arcot Sowmya, and Laughlin Dawes. Deep fusion pipeline for mild cognitive impairment diagnosis. In Proceedings - International Symposium on Biomedical Imaging, 2018.

[101] Yaroslav Shmulev and Mikhail Belyaev. Predicting conversion of mild cognitive impairments to alzheimer's disease and exploring impact of neuroimaging. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018.

[102] Hongyoon Choi. Deep Learning in Nuclear Medicine and Molecular Imaging: Current Perspectives and Future Directions, 2018.

[103] Dana Smith. Artificial intelligence can detect Alzheimer's disease in brain scans 6 years before a diagnosis. `https://www.universityofcalifornia.edu/news/`

artificial-intelligence-can-detect-alzheimer-s-disease-brain-scans-6-years-diagnosis, 2019.

[104] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning, 2017.

[105] Rahil Shaikh. Cross Validation Explained: Evaluating estimator performance. https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85, 2018.

[106] David Chuan-En Lin. 8 Simple Techniques to Prevent Overfitting. https://towardsdatascience.com/8-simple-techniques-to-prevent-overfitting-4d443da2ef7d, 2020.

[107] DataScienceGyan. Artificial Intelligence Vs Machine Learning Vs Deep Learning. https://datasciencegyan.com/artificial-intelligence-vs-machine-learning-vs-deep-learning/, 2018.

[108] CHM. Deep Learning, the basics and more! https://laptrinhx.com/deep-learning-the-basics-and-more-847480669/, 2020.

[109] Jason Brownlee. How to Choose an Activation Function for Deep Learning. https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/, 2021.

[110] Christoph Roser. Local Optima Global Optimum. https://www.allaboutlean.com/polca-pros-and-cons/local-global-optimum/, 2018.

[111] Saleh Albelwi and Ausif Mahmood. A framework for designing the architectures of deep Convolutional Neural Networks. Entropy, 2017.

[112] Nameer Hirschkind, Saruque Mollick, Jyo Pari, and Jimin Khim. Convolutional Neural Network. https://brilliant.org/wiki/convolutional-neural-network/, 2021.

[113] Christoph Lampert. Deep Learning with Tensorflow, 2021.

[114] Tanesh Balodi. Convolutional Neural Network (CNN): Graphical Visualization with Python Code Explanation. https://www.analyticssteps.com/blogs/convolutional-neural-network-cnn-graphical-visualization-code-explanation, 2019.

[115] Kaggle. Dogs vs. Cats: Create an algorithm to distinguish dogs from cats. https://www.kaggle.com/c/dogs-vs-cats/, 2013.

[116] Esmitt Ramirez and Fernando Álvarez. Vicot: Virtual collaboration tool to render images on the web. Acta científica venezolana, 67:26–43, 10 2016.

[117] Qihang Yu, Yingda Xia, Lingxi Xie, Elliot K. Fishman, and Alan L. Yuille. Thickened 2d networks for efficient 3d medical image segmentation, 2019.

[118] Juan Sandino, Geoff Pegg, Felipe Gonzalez, and Grant Smith. Aerial mapping of forests affected by pathogens using UAVs, hyperspectral sensors, and artificial intelligence. Sensors (Switzerland), 2018.

[119] Satya P. Singh, Lipo Wang, Sukrit Gupta, Haveesh Goli, Parasuraman Padmanabhan, and Balázs Gulyás. 3d deep learning on medical images: A review, 2020.

[120] Kurtis Pykes. Cross-Validation: Validating your Machine Learning Model Performance. https://towardsdatascience.com/cross-validation-c4fae714f1c5, 2020.

[121] Qingge Ji, Jie Huang, Wenjie He, and Yankui Sun. Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images. Algorithms, 2019.

[122] Viet Tra, Jaeyoung Kim, Sheraz Ali Khan, and Jong Myon Kim. Bearing fault diagnosis under variable speed using convolutional neural networks and the stochastic diagonal levenberg-marquardt algorithm. Sensors (Switzerland), 2017.

[123] Jason Brownlee. Ensemble Learning Methods for Deep Learning Neural Networks. https://machinelearningmastery.com/ensemble-methods-for-deep-learning-neural-networks/, 2018.