TÉCNICO LISBOA

Transfer Learning Methods for Alzheimer's Disease Diagnosis

Pedro Miguel Tavares Pereira

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor: Prof. Maria Margarida Campos da Silveira

Examination Committee

Chairperson: Prof. João Fernando Cardoso Silva Sequeira

Supervisor: Prof. Maria Margarida Campos da Silveira

Member of Committee: Prof. Ana Luísa Nobre Fred

January 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgements

I would like to express my gratitude towards Prof. Margarida Silveira, for the guidance and support provided during this thesis. All the comprehension, motivation and knowledge provided made the completion of this thesis possible. I would also like to thank professor Margarida for assigning me this theme. It's a subject that says a lot to me, given that it is present in my family and which now I can understand with more clarity. It's an area in which I would definitely like to work one day and have the opportunity to contribute to scientific progress.

I would also like to thank all my friends and family, who were always there for me, even in the most troubling times, despite the distance caused by the pandemic. To my parents, who made my journey in Técnico possible and provided me with the best conditions so that I could finish the course as soon as possible. To my grandparents and godparents who were always willing to help with everything they could. To my sister Joana, who had to put up with my most absurd moments. To my girlfriend for all the love, support and patience. To all my colleagues and friends from IST, from which I learned a lot during these five years and who were always willing to help me and made this experience enjoyable. To all my friends for all the pleasuring moments and leisure.

Thank you all so much.

Abstract

Alzheimer's disease (AD) is the most common cause of dementia among elderly people and one of the leading causes of death in developed countries, which is manifested by the loss of cognitive functioning and behavioral abilities, caused by progressive neuronal degeneration. Although there is still no practical diagnostic method available, a correct detection of the disease is crucial to slow down it's progression and cognitive decline. In recent years, Deep Learning methods gained popularity in AD detection, especially in dealing with brain scan images. Despite the success of these methods, the volume of medical images available is usually too small, which can easily lead to overfitting. At the same time, multimodal neuroimaging approaches have shown good results in AD diagnosis. In this work, a cross-modal Transfer Learning strategy was adopted using positron emission tomography (PET) and magnetic resonance imaging (MRI) brain scans: Deep Learning models based on Convolutional Neural Networks were pre-trained on one modality and fine-tuned using the other modality. The proposed approach obtained 86.4% accuracy for the classification task of AD vs normal controls (NC), showing improvement of around 2.5% of the classification system's accuracy with Transfer Learning, reducing overfitting, while taking advantage of the information provided by different neuroimaging modalities.

Keywords: Alzheimer's Disease; Transfer Learning; Convolutional Neural Network; Deep Learning; Medical Imaging; Multi-modality

Resumo

A doença de Alzheimer (AD) é a causa mais comum de demência entre idosos e uma das principais causas de morte nos países desenvolvidos, que se manifesta pela perda do funcionamento cognitivo e habilidades comportamentais, causadas pela progressiva degeneração neuronal. Embora ainda não exista um método prático de diagnóstico disponível, uma detecção correta da doença é fundamental para retardar a sua progressão e o declínio cognitivo. Nos últimos anos, os métodos de Aprendizagem Profunda ganharam popularidade na detecção de AD, especialmente através do uso de imagens cerebrais. Apesar do sucesso desses métodos, o volume de imagens médicas disponíveis é geralmente muito pequeno, o que pode facilmente levar a sobreajuste. Ao mesmo tempo, as abordagens de neuroimagem multimodal têm mostrado bons resultados no diagnóstico de AD. Neste trabalho, foi adotada uma estratégia de Aprendizagem de Transferência entre modalidades, usando tomografia por emissão de protões (PET) e imagens de ressonância magnética: modelos de aprendizagem profunda baseados em Redes Neurais Convolucionais foram pré-treinados numa das modalidades e ajustados usando a outra modalidade. A abordagem proposta obteve 86,4% de exatidão para a classificação entre AD vs sujeitos cognitivamente normais (NC), mostrando uma melhoria de cerca de 2.5% da exatidão do sistema de classificação através da Aprendizagem de Transferência, reduzindo o sobreajuste, ao mesmo tempo que aproveita as informações fornecidas por diferentes modalidades de neuroimagem.

Palavras-chave: Doença de Alzheimer; Aprendizagem de Transferência; Redes Neuronais Convolucionais; Aprendizagem Profunda; Imagens Médicas; Multimodalidade

Contents

List of Tables

List of Figures

List of Acronyms

AC	Anterior Commissure
ACC	Accuracy
AD	Alzheimer's Disease
AE	Autoencoder
ANN	Artificial Neural Network
AV-45	Florbetapir
BiLSTM	Bidirectional LSTM
CAE	Convolutional Autoencoder
CDR	Clinical Dementia Rating
CNN	Convolutional Neural Network
CSF	Cerebrospinal Fluid
CV	Cross-validation
DNN	Deep Neural Network
DTI	Diffusion Tensor Imaging
DTL	Deep Transfer Learning
EMCI	Early Mild Cognitive Impairment
FC	Fully Connected
FDG-PET	Fluorodeoxyglucose Positron Emission Tomography
fMRI	functional Magnetic Resonance Imaging
GM	Gray Matter
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
LMCI	Late Mild Cognitive Impairment
LSTM	Long Short-Term Memory
MCI	Mild Cognitive Impairment
MCI-C	Mild Cognitive Impairment - Converters
MCI-NC	Mild Cognitive Impairment - Nonconverters
MD	Mean Diffusivity
MMSE	Mini-Mental State Examination
MRI	Magnetic Resonance Imaging
MTL	Multi-Task Learning
NC	Normal Control
NN	Neural Network
PC	Posterior Commissure
PET	Positron Emission Tomography
ReLU	Rectified Linear Unit
ROI	Region of Interest

RNN	Recurrent Neural Network
SAE	Sparse Autoencoder
SENS	Sensitivity
SD	Standard Deviation
SMC	Significant Memory Concern
SPEC	Specificity
SVM	Support Vector Machine

1 Introduction

1.1 Motivation

Alzheimer's Disease (AD) is the most common cause of dementia among people over the age of 65, affecting more than 5 million Americans [1], as illustrated in figure 1.1. It is defined as the loss of cognitive functioning and behavioral abilities, interfering with a person's daily life and activities, such as difficulty in communication, speaking or walking [6]. Changes in the brain may begin a decade or more before memory and other cognitive problems appear. These changes are related to abnormal deposits of amyloid plaques and tau tangles throughout the brain, killing neuron cells and cutting connections with other neurons in an irreversible way, eventually leading to the person's death.

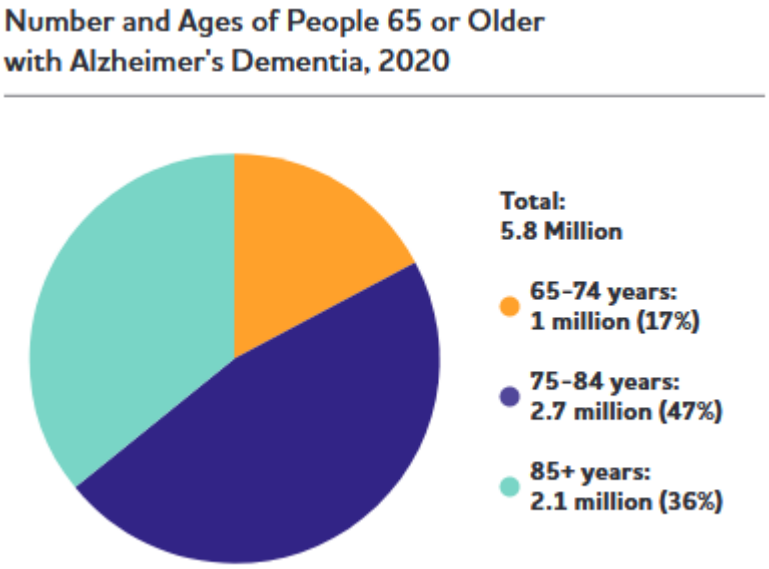


Figure 1.1: Incidence of AD in people over 65 years in the United States of America (adapted from [1]).

There are 3 main stages regarding AD: mild AD, moderate AD and severe AD. Preceding those stages, there is a phase called Mild Cognitive Impairment (MCI), in which people have more memory problems than normal for their age, but their symptoms do not interfere with their everyday lives. Some people with MCI may develop AD, but not all of them do. MCI patients that are likely to progress to AD are called MCI converters (MCI-C), while MCI patients that will not convert to AD are called MCI non-converters (MCI-NC). This rate of progression differs for each patient and each person may show different symptoms at each stage, which makes AD a challenging task for diagnosis and prognosis.

As the world population ages, it's estimated that by 2050, 1 in 85 persons will be living with AD [10]. Since current research in pharmacological intervention has not been able to reverse the disease course so far, if interventions could delay both disease onset and progression by 1 year, there would be nearly 9.2 million fewer cases of the disease in 2050, thus showing the importance of the improvement of early detection strategies for AD detection.

With the growth of popularity of machine learning methods and application of such strategies to AD diagnosis, it has been shown that those methods are able not only to achieve better performance in predicting AD than radiologists [37], but also, to increase the speed of the diagnosis without compromising the accuracy, suggesting a general adoption of computer-assisted methods for dementia diagnosis.

Among the several machine learning methods used to tackle this task in the recent years, it has been shown that Deep Learning methods obtain better results [47]. Deep learning is a specific sub field of machine learning, which consists on learning successive layers of increasingly meaningful representations, involving tens or even hundreds of successive layers of representations, all learned automatically, from exposure to training data, via neural networks [18].

The success of Deep Learning was boosted, recently, with the accessibility of affordable parallel computing resources via graphics processing units (GPUs) for computational acceleration. Deep learning architectures have been turned into advanced learning algorithms that extract high-level features directly from the images, without the engagement of human experts (computer aided feature extraction). The most popular deep learning model used for image analysis is the Convolutional Neural Network (CNN). When CNNs are trained on images, the first layers tend to learn generic features, such as edges, colors and textures, regardless of the cost function or dataset, as more deep layers tend to learn more abstract features related to particular dataset and task (specific). One of the major benefits of the CNN in comparison with other deep learning methods is that just as in a clinical context, where studies are based on visual analysis, the CNN is also motivated by the visual perception of humans. Since CNNs learn representations based on visual concepts, which can be interpreted by humans, they are able to contradict the notion of the 'black box' algorithm [11], that has grown in the biomedical field with the solutions proposed by Deep Learning. These black box AI systems can classify features and often do it with high accuracy, but without explaining the network's decisions associated with such classification.

Transfer Learning methods focus on solving one problem in the base domain and transferring the knowledge gained to a different but related problem, the target task. This process will tend to work if the learned features are general in the base domain. Usually, in Deep Learning, the method is based on training a base network and copying the first n layers to the first n layers of a target network. Some of the layers of the previous base task can then be fine-tuned to the new task or left frozen [62]. Fine-tuning works by unfreezing a few of the top layers of the frozen previously trained model and jointly training these layers with the top layers of the target network, slightly adjusting the more abstract representations of the model being reused to make them more relevant for the problem at hand. The use of a pre-trained CNN with adequate fine-tuning can outperform a CNN trained from scratch, using less amount of training data [56].

Most researchers believe that within the next 15 years, most of the medical diagnosis will be using deep learning based applications. Despite the growth of this research area, penetration of deep learning in healthcare is still quite slow, due to challenges related with the available datasets, legal issues, dedicated medical experts, non standard data, machine learning methods, etc [50]. One particular prob-

lem is that the volume of medical images available is usually too small to train the entire deep learning structure, which can easily lead to overfitting. This problem can be solved using Transfer Learning.

The combination of distinct biomarkers can even increase possibilities of a more in depth study of Transfer Learning methods: the data from a single modality can be used to train a CNN, as the base domain, and then transferring it to a target domain, containing images from a different modality. Transfer Learning therefore is a way to bypass the problem of having a small sample size for a deep learning architecture and to take full advantage of having several neuroimaging modalities at our disposal.

The identification of biomarkers for AD and it's combination with deep learning techniques that are able to identify patterns, features and hidden representations, contribute to the early detection of the disease and may accelerate the development of new therapies that can slow down the disease progression and cognitive decline, which can have huge impacts in patient's and caregivers' life quality.

1.2 Objectives

The goal of this thesis is to investigate Transfer Learning methods within a Deep Learning context for AD diagnosis, using neuroimaging data, with the aim to verify the several advantages of Transfer Learning enumerated in section 1.1, when compared to building a network from scratch. By doing this we study the effects of fine-tuning on initial and deep layers.

Additionally, the benefits of using multimodal data are explored through a cross-modal Transfer Learning approach in which two modalities of data are combined to classify subjects with AD from healthy NCs. This approach is compared to other more common approaches used to merge two types of neuroimaging data.

This method is implemented using two types of deep convolutional networks and conclusions are taken based on the performance metrics of the applied methods and the visual representations of the features learned by the models developed, making possible to see into the ”black box” of a deep learning model.

To date, from the information collected during this thesis, there are no studies that use cross-modal Transfer Learning for AD detection using PET and MRI modalities or using the deep learning networks that were implemented in this thesis, which means that this field hasn't been highly explored, whereby this work might lead to a better understanding in this area.

1.3 Thesis Outline

The rest of the paper is organized as follows. A literature review is presented in section 2. Section 3 shows the proposed methodology, while section 4 presents the obtained results for the proposed

Transfer Learning methods and establishes a comparison with other known methods, followed by the conclusions drawn from this approach in section 5.

2 State of the Art

The aim of this section is to frame the work developed in this thesis among existing solutions that try to apply neuroimaging to the problem of Alzheimer's Disease diagnosis using particularly Transfer Learning and multimodal machine learning approaches, accordingly to the objectives listed in section 1.2.

2.1 Scientific framework

This section discusses different aspects that machine learning methods that tackle AD classification encompass.

2.1.1 Classification Task

Deep learning architectures can be dimensioned to distinguish between two or more stages of AD. Researchers have considered multiple binary (usually normal controls (NC) vs pathological) or multiclass classification problems in order to distinguish between AD stages.

2.1.2 Temporal follow-up of subjects

Longitudinal studies follow each subject during a certain period of time, while cross-sectional studies evaluate each subject at a specific point in time. Usually longitudinal studies can obtain more accurate results, since they are more sensitive to early changes in the brain [21, 22, 27].

2.1.3 Imaging Modalities

In AD and related dementias, the most widely used biomarkers measure changes in the size and function of the brain and its parts, as well as levels of certain proteins seen on brain scans and in cerebrospinal fluid (CSF) and blood, being magnetic resonance imaging (MRI), the most popular imaging method. Several other methods have been used together with structural MRI, such as functional MRI (fMRI) [52], Diffusion Tensor Imaging (DTI) [9], Positron Emission Tomography (PET) images [63, 13, 16, 22, 24, 48, 60] and CSF [13, 14, 22, 41, 63].

In clinical practice it's expensive and time consuming to collect several biomarkers from subjects, and hence, the size of collected complete multimodal biomarker dataset is often small. This is why most research only focuses on a single modality of neuroimaging, such as MRI. However, in section 2.2.2 it is shown that the use of different biomarkers can provide complementary information for diagnosis of AD

and MCI. In a systematic literature review of over 100 recent articles related to deep learning methods applied to Alzheimer's Disease from neuroimaging, [57] concluded that multi-modality studies generally outperform single-modality, referring that combining different modalities will reflect different metabolic or structural aspects of AD, presenting a more accurate model of the disease, which is helpful especially for it's early detection.

2.1.4 Pre-Processing of Brain Scans

Pre-processing of brain scans strongly impacts in the performance of an AD detection system. Different machine learning methods have different pre-processing requirements, and in deep learning, some pre-processing steps became less critical. Most frequent and essential pre-processing techniques applied to raw images include intensity normalization and registration [21], but other techniques might be applied such as tissue segmentation, skull stripping and motion correction.

2.1.5 Data augmentation

One other method that addresses this lack of data related to medical images and multimodal datasets is data augmentation. This method expands the training dataset by adding modified copies of the data already existing in the training set or creating new samples from existing data. When applied to AD, deep learning methods usually perform data augmentation techniques, such as reflection, rotation, translation, noise injection, blurring, cropping, scaling and gamma correction. Other techniques focus on adding brain scans of subjects at different time points provided by longitudinal datasets, in a time-independent way [43, 21].

2.1.6 Input data management

Depending on the type of extracted features and information available after the feature extraction step in a computer aided classification system, studies can be grouped into voxel-based, slice-based, patch-based or ROI-based studies.

Depending on the type of features extracted, machine learning structures can perform feature dimension reduction, by prioritizing the most relevant information for the classification problem at hand and reduce high feature dimensionality.

2.1.7 Learning algorithm

A wide range of machine learning and deep learning methods have been used to classify AD and it's early stages. Earlier studies used machine learning methods like SVMs or semi-supervised SVMs,

however needing to perform feature extraction and classification in separate steps.

Nowadays most research methods focus on supervised deep learning such as CNNs or DNNs. Unsupervised deep learning algorithms such as Autoencoders can also be used as a good initialization step for deep neural networks, or extract high-level features in an unsupervised way.

Many deep learning algorithms follow the logical step of letting computers learn the features that optimally represent the data for the problem at hand, which is the case of CNNs. 2D CNNs are widely used for Transfer Learning purposes, since it is common to use well known architectures pre-trained on the popular ImageNet dataset [51]. Among these architectures are LeNet [40], AlexNet [38], CaffeNet [59], VGGNet [53], GoogLeNet [55], ResNet [28], Inception [54], and SqueezeNet [32]. This is possible due to the fact that lower CNN layers are able to learn general features, which can benefit many classification tasks.

One issue is that 2D CNNs are not able to capture volumetric information from the 3D imaging modalities. This issue can be addressed with 3D CNNs or with the combination of 2D CNNs and Recurrent Neural Networks (RNNs) in order to capture spatial information. Several methods used different ways to combine 2D CNNs and RNNs to deal with this problem [20, 15].

The main competition at this time seems to be between 3D CNNs and 2D CNNs (with or without RNNs) [21].

2.2 State of the art review

In this section, different applications of Transfer Learning for AD detection within the last 10 years are summarized. In addition, several methods in which multimodal neuroimaging data has been combined are described.

2.2.1 Transfer Learning methods for AD detection

Transfer Learning methods have been proven to be robust even for very dissimilar domains, such as networks trained on a dataset containing natural images used with medical images [30, 61, 49, 27, 44, 45, 20], which is the most common application of TL in AD detection, but there are several other ways to apply Transfer Learning to Alzheimer's early detection problem, as detailed in this section. Generally, the use of TL speeds up training and improves performance even when reusing a pre-trained network from a distant task, such as a generic image-classification task.

In a deep learning architecture, the number of layers from the original model to be fine-tuned and the number of top replaced layers depends from case to case. In the case of CNNs, it is usual to replace the densely connected classifier and fine-tune some layers of the convolutional base [18], once

the representations learned by the convolutional base are frequently more generic and therefore more reusable. Despite the current trend in Transfer Learning, an in-depth study on the reusability of the representations of each layer and on the number of fine-tuned layers that benefit performance appears to be missing for AD detection.

Research reports very distinct methods: [13] uses a Support Vector Machine (SVM) for classification of MCI-C vs MCI-NC patients, based on a related auxiliary domain, given by the task of classifying AD vs NC patients. The same authors extend their TL approach to use multiple auxiliary domains - Multi-Domain Transfer Learning [12, 14]. The classification was made by SVMs and obtained measures of performance for several tasks. Each task used the others as auxiliary domains, for example, in [12] the task of classifying MCI-C vs MCI-NC patients was based on the AD vs NC and MCI vs NC tasks, and [14] the target task MCI-C vs MCI-NC used also AD vs MCI in the auxiliary domain.

Filipovych $\&$ Davatzikos [25] had already been using, in 2011, AD vs NC domain to target the MCI-C vs MCI-NC classification, using a semi-supervised SVM to classify MCI subjects in the absence of certain diagnostic information for some patients in the ADNI database. In their work, an analysis about volumetric differences in gray matter (GM) structures classified for AD and NC was made.

Because these methods used SVMs, there was a necessity of performing feature extraction manually. Regions of interest (ROIs) were labeled from each image and for each ROI the volume of GM tissue was computed as a feature [12, 14, 25]. [13] also used average pixel intensity of each ROI for the PET images as features and three CSF biomarkers: CSF $A\beta_{42}$, CSF t-tau, CSF p-tau. [14] also used CSF $A\beta_{42}$, CSF t-tau, CSF p-tau as features.

Other methods used CNNs as learning algorithm, and therefore, no manual feature extraction was needed, since CNNs can learn features from the training data on their own. [30] used MRI data from the OASIS dataset [7], using VGG16 and Inception V4 architectures, with pre-trained weights from ImageNet and fine-tuning. An accuracy of 96.25% was obtained for AD vs NC classification with the Inception architecture, also showing that this type of Transfer Learning achieves better results than only training the same model from scratch, when the training size is small, which can result in overfitting. Wu et al. [61] also compared the performance of GoogleNet and CaffeNet architectures using Transfer Learning from pre-trained ImageNet (and fine-tuning), obtaining accuracy measures of 87.78% for the three way classification of NC vs MCI-C vs MCI-NC for the CaffeNet architecture and 83.23% for GoogleNet. In their work, a novel data augmentation strategy was also used, which selected 3 slices from the 3D volume MRI data with a certain interval to combine a RGB color image, used as input to the network. Transfer Learning using pre-trained ResNet architectures and fMRI images from ADNI was used in [49] which achieved an average accuracy of 97.92% in the multi-class classification of AD, NC, significant memory concern (SMC) and three MCI stages, including early MCI (EMCI), MCI, and late MCI (LMCI). The authors compared this architecture with the AlexNet architecture for Transfer Learning, concluding that the use of residual learning (ResNet) and Transfer Learning both improved the performance. In [27], a pre-trained ResNet model was fine-tuned from MRI slices, extracting slice-level features and a Long

Short-Term Memory (LSTM) layer is then used to learn longitudinal-level features for each subject.

Lu et al. [44] also addressed this type of Transfer Learning for classification between individuals with brain pathology (AD among them) and NC. Using the AlexNet architecture, pre-trained on ImageNet, with fine-tuning, they achieved 100% accuracy for this task. [45] also fine-tuned a pre-trained AlexNet architecture to classify segmented GM, WM and CSF images and unsegmented MRI images from the OASIS dataset, in which the unsegmented images led to better results for the multi-class classification of the multiple stages of AD (in this case, the classified stages were NC, very mild AD, mild AD and moderate AD). Both of these studies modified the original AlexNet network for the target task, by replacing the last three layers with new layers with randomly initialized weights to learn class specific features in the target domain. Then, the weights of the remaining layers were adjusted during training jointly with the replaced layers in the case of [44], or remained fixed in [45] and only the replaced top layers had their weights updated.

Other studies analysed the current trend of using Transfer Learning from natural images to AD classification, specifically, using networks pre-trained on ImageNet, such as [20], which implemented several well-known 2D CNNs to extract discriminative features from MRI slices and a LSTM to incorporate spatial information across slices in the classification, showing better results when using the pre-trained SqueezeNet model.

A different Transfer Learning method was used by Hosseini-Asl et al. [31], in which a 3D CNN was built upon a stacked 3D convolutional autoencoder (CAE) network, which was pre-trained on CAD-Dementia dataset [3] (base domain) in order to capture anatomical shape variations in structural MRI scans. The 3D CNN's layers were initialized by encoding the 3D-CAE weights and the upper layers were then fine-tuned for the specific task using data from the ADNI dataset (target domain), achieving 100% accuracy in AD vs MCI classification and more than 90% accuracy in the other evaluated tasks.

Similarly, Payan et al. [47] used a 3D CNN for AD diagnosis based on pre-training by a 3D sparse autoencoder (SAE). This pre-training was performed by randomly selecting small 3D patches of MRI scans. The trained weights of the SAE are then used for pre-training of convolutional filters of 3D CNN. The fully connected layers of the 3D CNN are then fine-tuned. This method was improved by Vu et al. [60], to combine MRI and fluorodeoxyglucose positron emission tomography (FDG-PET).

Focusing on the Hippocampal region and in two distinct imaging modalities: DTI. in particular Mean Diffusivity (MD) density maps derived from DTI and sMRI, [9] proposed a cross-modal Transfer Learning method for classification between AD, NC and MCI, in which a 2D CNN model is trained first on the sMRI dataset and then fine-tuned on the target MD dataset, with a limited amount of data, for each projection (Sagittal, Axial and Coronal). The results from each projection were fused using Majority Vote. This cross-modal Transfer Learning method showed a reduction of overfitting and improvement of learning performance, and encouraged a new perspective in Transfer Learning for Alzheimer's disease detection, in which each domain is represented by a different neuroimaging modality.

2.2.2 Combination of neuroimaging modalities

There are several ways to fuse multimodal data, particularly neuroimaging data. Imaging data can be also combined with other available information such as cognitive measures or demographic information, and selecting the best modality combination and fusion method is a task that's been studied in-depth.

Multimodal classification was compared with the case of using only one biomarker by [63]. They combined MRI, PET and CSF biomarkers using a kernel combination method to train an SVM and obtaining an accuracy of 93.2% for the classification of AD vs NC and 76.4% for MCI vs NC. These results were better than using only one biomarker, emphasizing the benefits of having multimodal data. Similar methods based on SVMs that were mentioned in the previous section, reported better results when combining MRI and PET images and CSF biomarkers [13] and MRI and CSF biomarkers [14].

An early fusion method was used in [16], which combined two types of PET images, FDG-PET and ^{18}F-florbetapir (AV-45) PET, to train a 3D CNN for the AD vs NC task, exploring both glucose metabolism and amyloid deposit in the patient's brain at the same time. This network was then applied to predict between MCI-C and MCI-NC subjects, showing better results when both modalities were used simultaneously to train the model.

Although this method achieved good performance and can be easily implemented, combining modalities in the inputs of the same deep learning model is unusual. Instead, it is more common to use a different network to learn features from each modality and combine the networks at a later stage. In this sense, to predict conversion of MCI to AD, [41] used a multimodal gated recurrent unit (GRU) network, which integrated subject's demographic information, longitudinal CSF biomarkers, longitudinal cognitive performance and cross-sectional MRI images obtained from ADNI. This required two steps: the training of a single GRU separately for each modality of data and merging the four networks into one. In the first step, the GRU makes it possible to transform longitudinal data into a fixed-length vector, exploiting temporal patterns in a data sequence, which is then integrated with the data from several modalities through concatenation in the second step. With the incorporation of several modalities into one prediction model, while using longitudinal data, the accuracy improved from 75% to 81% for the classification between MCI-C and MCI-NC. [22] also combined time series neuroimaging data from MRI and PET and subject's cognitive scores from 15 time steps and static background knowledge from the patients' first visits, (such as age, gender, CSF, symptoms, etc.) to predict disease progression and four cognitive scores at the time of progression in a multitask and multi-class deep learning framework which achieved 92.62% accuracy for the CN vs MCI-NC vs MCI-C vs AD task. Deep features extracted from each time series modality and fed into a separate stacked CNN-Bidirectional Long-short term memory (BiLSTM) pipeline and the learned representations are fused together with a set of dense layers. In a second fusing step, the common features from these modalities are fused with the baseline background data features and a final set of dense layers are used to learn task specific features.

Using a similar concatenation method, [48] designed a fusion model that obtained 92.34% accuracy

for the AD vs NC classification problem using MRI and florbetapir PET images from ADNI. In this work, the authors built a 3D CNN for each modality with three convolutional layers and three fully connected layers. Then, to perform fusion, the output layer of both networks is replaced by a concatenation layer, which fuses the information from both modalities before the final classification is made. The improvements in performance due to fusion of both modalities indicates that the two modalities share complementary information useful in this task, although the authors revealed amyloid (AV-45) PET to be more discriminative in comparison to MRI in the first study that fused and compared these two modalities. A more common choice of modalities is MRI and FDG-PET which was fused in [60] and the authors reported improvements in comparison to the classification of each modality separately for the AD vs NC problem, reaching 91.14%. This approach fused the outputs of two 3D CNNs trained for each modality through a 3-layer fully connected layer neural network. This boosting of the performance is not only due to fusion, but also the pre-training of the CNN using a SAE trained on random 3D patches from the scans, similarly to [47]. [43] achieved an accuracy of 82.93% for discrimination between MCI-NC and MCI-NC subjects, by concatenating the representations learned from six Deep Neural Networks (DNNs), which corresponded to three different patch scales from FDG-PET and GM and using another DNN to fuse these representations.

Instead of using fully connected layers to share information between modalities, [24] achieved better results using a Bidirectional-RNN, which took as inputs the features extracted from a 3D CNN trained on PET images and GM density maps segmented from anatomical MRI images.

Transfer Learning was also used as a method to combine imaging modalities which showed promising results in [9], described in section 2.2.1.

Apart from combining modalities, different views from brain scans (Sagittal, Axial and Coronal) from the same modality can be combined to achieved a global prediction score, which is done frequently using Majority Vote, as in [20] and [9].

2.2.3 Summary

A summary of the Transfer Learning methods referred previously in section 2.2.1 is presented in table 2.1 regarding Transfer Learning and in table 2.2 for the combination of different neuroimaging modalities, without Transfer Learning. In each table, performance information is provided in terms of accuracy, sensitivity and specificity for each method, as well as the database from which the data was download and number of participants, to highlight differences between each study that might it harder to compare the different approaches.

Table 2.1: Performance of different AD classification systems which combine different imaging modalities, apart from Transfer Learning. Results for different tasks are in regard to: a - AD vs NC; b - MCI vs NC; c - AD vs MCI; d. MCI-C vs MCI-NC; e. MCI-C vs NC f. MCI-NC vs NC; g. NC vs MCI-C vs MCI-NC; h. NC vs SMCI vs EMCI vs MCI vs LMCI vs AD; i. NC vs mild AD vs very mild AD vs moderate AD; j. AD vs MCI-NC vs MCI-C vs NC; Abbreviations: Accuracy (ACC); Sensitivity (SENS); Specificity (SPEC).

Author(s)	Year	Biomarker(s)	Learning Algorithm	Subjects	Database	Transfer Learning Type	ACC (%)	SENS (%)	SPEC (%)
Cheng et al. [13]	2015	MRI, FDG-PET, CSF	SVM	51 AD, 43 MCI-C, 56 MCI-NC, 52 NC	ADNI	MCI-C vs MCI-NC based on AD vs NC auxiliary domain	79.4^{d}	84.5^{d}	72.7^{d}
Cheng et al. [12]	2017	MRI	SVM	186 AD, 395 MCI, 226 NC	ADNI	MCI-C vs MCI-NC based on multiple auxiliary domains (^{a,b})	94.7^{a} 73.8^{d}	94.1^{a} 69.0^{d}	94.8^{a} 77.4^{d}
Cheng et al. [14]	2019	MRI, CSF	SVM	102 AD, 192 MCI, 112 NC	ADNI	MCI-C vs MCI-NC based on multiple auxiliary domains (^{a,b,c})	95.2^{a} 76.3^{d}	95.2^{a} 73.4^{d}	95.3^{a} 81.8^{d}
Filipovych et al. [25]	2011	MRI	Semi-supervised SVM	54 AD, 68 MCI-C, 174 MCI-NC, 63 NC	ADNI	MCI-C vs MCI-NC based on AD vs NC auxiliary domain	82.91^{a}	79.63^{a}	85.71^{a}
Hon et al. [30]	2017	MRI	2D CNN	100 AD, 100 NC	OASIS	Pre-trained Inception V4 network and fine-tuning	96.25^{a}	-	-
Wu et al. [61]	2018	MRI	2D CNN	150 MCI-NC, 157 MCI-C, 150 NC	ADNI	Pre-trained CaffeNet and fine-tuning	87.78^{g}	-	-
Ramzan et al. [49]	2020	fMRI	2D CNN	25 NC, 25 SMC, 25 EMCI, 25 LMCI, 13 MCI, 25 AD	ADNI	Pre-trained ResNet	97.92^{h}	97.92^{h}	-
Gao et al. [27]	2018	MRI	2D CNN + LSTM	111 AD, 150 MCI, 154 NC	ADNI	Pre-trained ResNet and fine-tuning	89.5^{a} 81.7^{b}	-	-
Lu et al. [44]	2019	MRI	2D CNN	177 pathological, 38 NC	Harvard Medical School Website	Pre-trained AlexNet and fine-tuning	100^{a}	100^{a}	100^{a}

Table 2.1: *Continued from previous page*

Author(s)	Year	Biomarker(s)	Learning Algorithm	Subjects	Database	Transfer Learning Type	ACC (%)	SENS (%)	SPEC (%)
Maqsood et al. [45]	2019	MRI	2D CNN	167 NC, 87 very mild AD, 105 mild AD, 23 moderate AD	OASIS	Pre-trained AlexNet	89.66^{a} 92.85^{i}	100^{a} 92.85^{i}	82.0^{a} 74.27^{i}
Ebrahimi-Ghahnavieh et al. [20]	2019	MRI	2D CNN + LSTM	132 AD, 132 NC	ADNI	Pre-trained SqueezeNet	90.62^{a}	-	-
Hosseini-Asl et al. [31]	2016	MRI	3D CNN	70 AD, 70 MCI, 70 NC	ADNI	stacked 3D CAE network pre-trained on CADDementia dataset	99.3^{a} 94.2^{b} 100^{c}	98.6^{a} 100^{b} 100^{c}	97.2^{a} 100^{b} 98.6^{c}
Payan et al. [47]	2015	MRI	3D CNN	755 AD, 755 MCI, 755 NC	ADNI	SAE pre-trained with random 3D patches from the scans	95.39^{a} 92.11^{b} 86.84^{c}	-	-
Vu et al. [60]	2017	MRI, FDG-PET	3D CNN	145 AD, 172 NC	ADNI	SAE pre-trained with random patches from the scans	91.14^{a}	-	-
Aderghal et al. [9]	2018	MRI, DTI	2D CNN	188 AD, 399 MCI, 228 NC	ADNI	Model trained on the MRI dataset and fine-tuned on the DTI dataset	92.5^{a} 80.0^{b} 85.0^{c}	94.7^{a} 92.8^{b} 93.7^{c}	90.4^{a} 73.0^{b} 79.1^{c}

Table 2.2: Performance of different AD classification systems which combine different imaging modalities, apart from Transfer Learning. Results for different tasks are in regard to: a - AD vs NC; b - MCI vs NC; c - AD vs MCI; d. MCI-C vs MCI-NC; e. MCI-C vs NC f. MCI-NC vs NC; g. NC vs MCI-C vs MCI-NC; h. NC vs SMCI vs EMCI vs MCI vs LMCI vs AD; i. NC vs mild AD vs very mild AD vs moderate AD; j. AD vs MCI-NC vs MCI-C vs NC; Abbreviations: Accuracy (ACC); Sensitivity (SENS); Specificity (SPEC).

Author(s)	Year	Biomarker(s)	Learning Algorithm	Subjects	Database	Combination type	ACC (%)	SENS (%)	SPEC (%)
Zhang et al. [63]	2011	MRI, PET, CSF	SVM	51 AD, 43 MCI-C, 56 MCI-NC, 52 NC	ADNI	Kernel combination	93.2^{a} 76.4^{b}	93.0^{a} 81.8^{b}	93.3^{b} 66.0^{b}
Choi et al. [16]	2018	FDG-PET, AV-45 PET	3D CNN	139 AD, 171 MCI, 182 NC	ADNI	Joint training of both modalities in the same network	96.0^{a} 84.2^{d}	93.5^{a} 81.0^{d}	97.8^{a} 87.0^{d}
Lee et al. [41]	2019	MRI, CSF, demographic information, cognitive performance	GRU	338 AD, 307 MCI-C, 558 MCI-NC, 415 NC	ADNI	Concatenation-based	81.0^{d}	84.0^{d}	80.0^{d}
EL-Sappagh et al. [22]	2020	MRI, FDG-PET, cognitive scores, static background knowledge	1D CNN + LSTM	339 AD, 473 MCI-NC, 305 MCI-C, 419 CN	ADNI	Concatenation-based	92.62^{j}	-	-
Punjabi et al. [48]	2019	MRI, AV-45 PET	3D CNN	723 total	ADNI	Concatenation-based	92.34^{a}	-	-
Lu et al. [43]	2018	MRI, FDG-PET	DNN	409 MCI-NC, 217 MCI-C	ADNI	Concatenation-based	82.93^{d}	79.69^{d}	83.84^{d}
Feng et al. [24]	2018	MRI, FDG-PET	3D CNN + BiRNN	91 AD, 76 MCI-C, 128 MCI-NC, 100 NC	ADNI	Concatenation-based	94.29^{a} 84.66^{e} 64.47^{f}	96.59^{a} 83.56^{e} 70.43^{f}	92.38^{a} 89.63^{e} 67.14^{f}

3 Methods

This chapter explains the fundamental concepts related to the deep learning framework and Transfer Learning methods explored and implemented during this thesis, in section 3.1. Section 3.2 details how these concepts were implemented in practice for AD diagnosis.

3.1 Theoretical Background

3.1.1 Data

3.1.1.1 Neuroimaging modalities

For the problem of AD detection, two imaging modalities were used: MRI-based images and PET images. The MRI-based images used consist in the GM tissue component of the MRI volume, whereas the PET images consist in the full brain image. Both of these modalities present a measure of neurodegeneration: the progressive loss of neurons or their connections and corresponding impairment in neuronal function [34].

Magnetic resonance imaging (MRI) is a medical imaging technique used in radiology to form pictures of the anatomy of the brain or other body organs through the use of strong magnetic fields, magnetic field gradients, and radio waves. MRI is based on the polarization of protons in a magnetic field. A pulse of radiofrequency alters the energy state of protons, which emit a radiofrequency signal as they return to their energy state when the pulse is turned off [35]. Different combinations of gradients and pulse sequences can be designed to explore different tissue characteristics.

Structural imaging based on MRI is an integral part of the clinical diagnosis of patients with suspected AD, since this disease is shown to be associated with the abnormal deposits of amyloid plaques and tau tangles in the brain, which lead to progressive brain tissue damage in characteristically vulnerable brain regions. Several studies show that structural MRI can estimate tissue damage or loss in these vulnerable regions [26, 35] and therefore can estimate cognitive impairment. Brain atrophy is first manifest in the medial temporal lobe, particularly in the entorhinal cortex, followed by other structures such as the hippocampus, amygdala and parahippocampus, also affecting other structures within the limbic lobe, such as the posterior cingulate. Attrophy is then spread to involve the temporal neocortex and neocortical association areas. MRI brain scans can be segmented into the main types of brain tissue: GM, CSF and white matter (WM). Segmentation is the method that identifies the set of voxels which make up either the contour or the interior of a given object of interest, which allows the reduction of search area in an image. Previous studies confirmed that GM is highly related to AD in comparison with WM or CSF [24].

Fluorodeoxyglucose positron emission tomography (FDG-PET) is another important modality of biomarkers for AD and MCI detection. This functional imaging technique is used to observe metabolic processes in the body, through the intravenous injection of a short-lived radioactive tracer isotope into the subject. This radiotracers are the aggregation of carrier molecules which are bonded to a radioactive isotope. The carrier molecule FDG is an indicator of glucose consumption in the tissues and can be labeled with the isotope Fluorine-18 (fluorodeoxyglucose F18 (FDG)). These radiotracers emit amounts of energy reaching 511keV for F18, which are measured by a scanner, producing a 3D image of the distribution of FDG in the body. Recent studies have reported the reduction of glucose metabolism in parietal, posterior cingulated, and temporal brain regions for AD patients [19].

Challenges related with neuroimaging biomarkers, like the ones used, are related to the probable co-occurrence of other brain related pathologies in one subject at the same time, particularly in elderly individuals in which AD is more frequent, such as cerebrovascular disease, syrucleinopathy and hippocampal sclerosis. In this scenario, an individual can have two or more co-occurring pathophysiological processes present, one of which being AD, or an individual can have a predominant non-AD pathophysiological process [34].

3.1.1.2 Subject evaluation

The subjects are evaluated over time, in a longitudinal way. Longitudinal datasets provide several brain scans per subject at different time points, and can be used to investigate disease progression or in a time-independent way.

In this sense, it was taken advantage of the longitudinal dataset, which provides several brain scans per subject, at each different time periods for each modality, in order to reduce overfitting when training the model and generalizing to new data. However, it has to be taken into account that scanning sessions from the same patient should not be used in both training and test sets, which would result in an ”information leak” and the algorithm would overfit to the patient’s identity rather than learning the disease pattern, causing overoptimistic test results. The data partitioning procedure is further detailed in section 3.2.3.

3.1.1.3 Feature type

The features used for image classification for either one of the modalities consist in the raw voxel intensities, which form 3D volumes, composed by slices of 2D images, that are going to be taken as input by the deep learning model.

3.1.2 Pre-processing

The pre-processing of brain scans accounts for the first stage in the deep learning pipeline and plays a key role in the performance of an AD detection system [21]. A feature normalization method was explored, apart from the methods that had been previously applied to the data, mentioned in section 3.2.4.

3.1.2.1 Feature Normalization

Feature normalization corresponds to the mapping of intensities of the volumes' voxels to a reference scale. Given the different range of intensities in both modalities, feature normalization is required, which speeds up model training by avoiding extra iterations that are required when features' values occupy very different ranges.

The approach taken was the normalization of features to the range [-1, 1], which was calculated applying the equation

$$\hat{x}_k^i = \frac{x_k^i}{max(|x_k|)}, \qquad k = 1, 2, \quad i = 1, 2, ..., N_k, \tag{3.1}$$

where $\hat{x}_k^i$ corresponds to the normalized voxel value from modality k, N_k is the total number of features corresponding to modality k and $max(|x_k|)$ is the maximum of the modulus from features corresponding to modality k in the training set.

3.1.3 Supervised learning

The goal of supervised learning is to find a decision function $f(x)$ that correctly predicts the output given a pair of input-output:

$$\hat{y} = f(x), \tag{3.2}$$

assuming there is a training set $T = (x_i, y_i), \quad i = 1, ..., n$ with $x_i \in \mathbb{R}^p$, where p corresponds to the number of input features and y_i to the class label and n is the total number of training patterns.

3.1.4 Neural Networks

An artificial neural network (ANN) is an interconnected group of artificial neurons inspired by biological processes in the brain, in which the relationship between the neurons' (nodes) inputs and outputs in a given layer can be written as

$$o^k = g((x^k)^T \cdot w^k - b^k), \tag{3.3}$$

where $o^k \in \mathbb{R}^n$ is the output vector of the k^{th} layer of an ANN, $x_k \in \mathbb{R}^n$ is the input vector, n is the total number of nodes in layer k, $w^k \in \mathbb{R}^{mxn}$ the weight matrix, which relates information from the previous layer (with m nodes), and $b^k \in \mathbb{R}^n$ a bias vector. The function g is called the activation function. In a fully connected (FC) neural network, all the outputs from a given layer are connected to all the units of the next layer, being the inputs of this layer represented by o^{k-1}.

Activation functions add non-linearity to the network so that the network can learn complex patterns that benefit from adding multiple layers. The ReLU function is the current recommended activation function and softmax or sigmoid layers are common in final layer's activation functions.

The training of a NN is done by estimating all the weights in an iterative way, given a set of training patterns T defined in section 3.1.3. This can be achieved by obtaining predictions for each input training pattern and minimizing the empirical risk, defined as

$$\mathcal{R} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y^i, \hat{y}^i), \tag{3.4}$$

where n is the number of classes in the classification problem and $\mathcal{L}$ corresponds to the loss function - a measure of the mismatch between the prediction ($\hat{y}^i$) for a given input feature and the corresponding target y^i. The training loop is followed by the update of all the weights of the network, according to the measures obtained from equation 3.4, reducing training loss in each set (batch) of inputs.

A common loss function is the categorical cross-entropy loss, which is given by the expression

$$\mathcal{L} = -\sum_{i=1}^{M} y^i log(p^i(x)), \tag{3.5}$$

where M is the number of classes in a given classification problem, y^i the target corresponding to a given class i and $p^i(x)$, the prediction probability of class i given an input x, achieved by the last layer activation function. In the case of a binary classification problem, the categorical cross-entropy loss is reduced to

$$\mathcal{L} = -(ylog(p(x)) + (1-y)log(1-p(x))), \tag{3.6}$$

where y is the class label (0 or 1) and $p(x)$ the prediction probability for class 1 given an input x.

The minimization of $\mathcal{R}$ can be achieved by using the Backpropagation algorithm [18], in which the NN's weights are updated in order to minimize the loss function, which means finding the combination

of weights values that yields the smallest possible loss function by computing the gradient of the loss function with regard to the network's parameters and moving the parameters in the opposite direction from the gradient. This can be done given a training input at a time (online mode), all inputs at a time (batch mode), or using a small group of training samples (mini-batch) at each time. The computation of the gradients and update of parameters can be done by several optimization methods, such as stochastic gradient descent (SGD), or Adaptive Momentum Estimator (Adam) [36].

3.1.5 Deep Learning models

Two supervised deep learning architectures based on CNNs were explored: CNN-LSTM and 3D CNN. CNNs are the most successful deep model for image analysis. These models took as input GM and PET volumes for the classification task of AD vs NC. The results and conclusions about the application of these two models to the task are shown in section 4.

3.1.5.1 Convolutional Neural Networks

The 2D CNN was applied successfully in several imaging classification problems, such as handwritten digit recognition [39]. Applications in AD classification are described in section 2. This type of networks is composed of neurons having learnable weights and biases, forming the convolutional layer. This layer computes the output of neurons that are connected to local regions in the input (feature maps). The convolution operation extracts patches from its input feature map and applies the same transformation to all of these patches, producing an output feature map by computing the dot product between the kernel weight and the input feature map. The output depth is a parameter of the layer. Different channels in that depth axis stand for different filters that encode specific aspects of the input data.

Mathematically, the convolution operation performed in a two-dimensional image can be denoted as

$$s(x,y) = (f * g)[x,y] = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} f(n_1,n_2) \cdot g(x-n_1, y-n_2), \tag{3.7}$$

where x and y represent the pixel positions for a given 2D data, s the output feature map resulting from the convolution between the input image f and a convolution window (kernel) g, which slides through the input image. Every location in the output feature map corresponds to the same location in the input feature map.

The convolutional layer accepts volumes of size $W \times H \times D$ and requires four hyperparameters: K, the number of filters; F, size of the patches extracted, or receptive field; S, the size of the stride; P, the amount of zero padding. The convolutional layer then returns a volume of size $W_2 \times H_2 \times D_2$, as

described in equations 3.8a, 3.8b and 3.8c.

$$W_2 = (W - F)/S + 1 \tag{3.8a}$$

$$H_2 = (H - F)/S + 1 \tag{3.8b}$$

$$D_2 = D \tag{3.8c}$$

Zero padding consists on adding zeros on the border of the input feature map, in order to get an output feature map with the same dimensions as the input. Stride is the distance between two successive convolution windows. This parameter can be used for downsampling. For example, using stride 2 means the width and height of the feature map are downsampled by a factor of 2.

The max-pooling layers, also take the role of downsampling feature maps and consist of extracting windows from the input feature maps and outputting the max value of each channel. It is common to use max-pooling layers which perform downsampling by a factor of two.

3.1.5.2 Long Short-Term Memory

LSTM (Long Short-Term Memory) [29] is an artificial recurrent neural network (RNN) architecture, which can process sequences of data, such as video. In an RNN, sequences are processed by iterating through the sequence elements and maintaining a state containing information relative to past computations. In practice, RNNs can't learn properly information about inputs seen many timesteps before, due to the vanishing gradient problem: the gradient's value shrinks exponentially as it propagates through each timestep, as explained in [29]. The LSTM attempts to solve this problem, adding a way to carry information across many timesteps, which can be reinjected at a later time.

In figure 3.1 it's described how the LSTM layer includes an additional data flow that carries information across timesteps (c_t). The update of c_t is done by three gates that compose the LSTM cell, the input gate, the output gate that provide information about the present and a forget gate, which is a way to forget irrelevant information in the carry dataflow.

3.1.5.3 CNN-LSTM model

A combination of 2D CNNs and LSTM can be designed to include spatial relations among 2D slices of neuroimaging volumes. Different combinations of these two types of layers were explored in the literature review in section 2 for the problem of AD detection. Usually these methods divide each volume into several groups of slices and for each group, a different CNN is trained and an RNN is used to group the CNN's outputs in an ordered way.

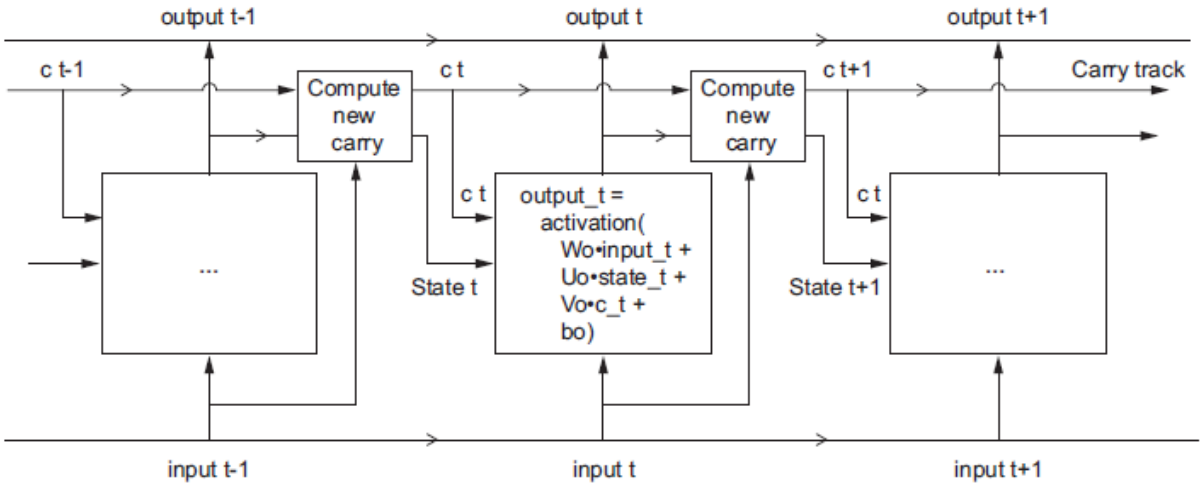


Figure 3.1: Anatomy of an LSTM (adapted from [18]).

A combination of time-distributed CNNs and RNNs have been used in video classification, but still hasn't been applied to AD detection. This method uses the Keras TimeDistributed wrapper, which allows the distribution of CNN layers across every slice of a 3D input. In this sense, 3D neuroimaging volumes can be seen as classification of video sequences, where the third dimension is spatial, instead of temporal.

This is illustrated in figure 3.2, where the TimeDistributed wrapper applies the same instance of the convolution layer to every slice in the input volume, using the same set of weights for every convolution and thus applying the same transformation for a list of input data, which is the main difference relatively to other applied methods that combine CNNs and LSTM. The LSTM layer can then be used to process images in a given order, detecting the relationship between slices, followed by a set of fully connected layers to compute the final prediction.

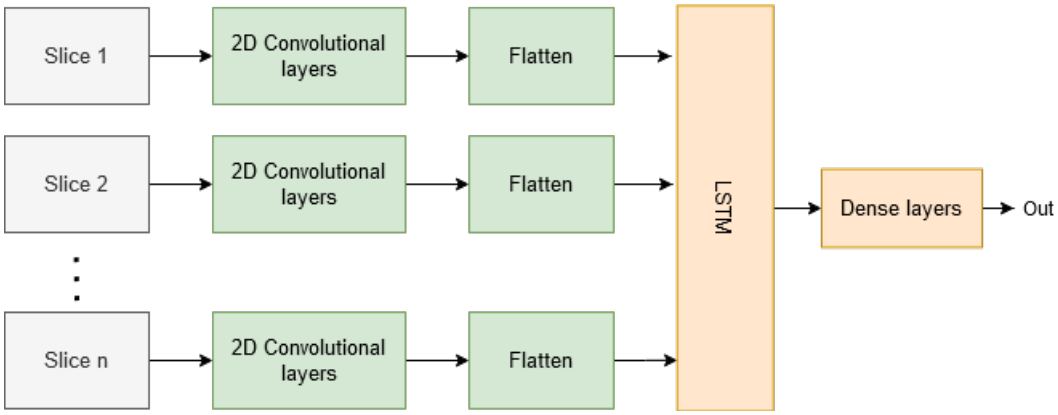


Figure 3.2: Representation of the CNN-LSTM model with TimeDistributed CNN layers.

3.1.5.4 3D CNN model

In 3D CNNs, convolutions are applied on 3D feature maps, as in figure 3.3. These convolutions are performed by adding a third spatial dimension to equation 3.7. Instead of three dimensions in the case of 2D CNNs, in this case there are four dimensions: two image dimensions, the time/height dimension and the channels dimension. Since the filters move in three dimensions, 3D CNN layers are more expensive in terms of computational resources.

3D convolutions have been applied in medical imaging in problems such as AD detection (section 2) or lung anomaly detection [23] and have shown good results in modeling spatiotemporal features.

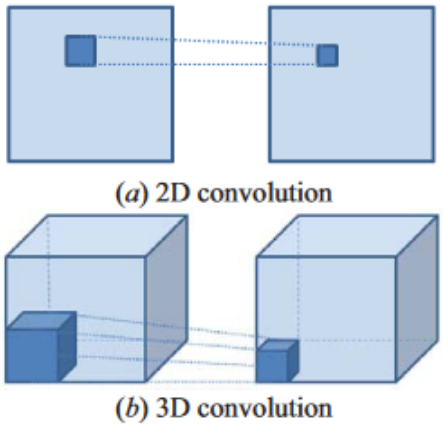


Figure 3.3: Comparison between (a) 2D convolution and (b) 3D convolution (adapted from [23]).

3.1.6 Transfer Learning

A cross-modal Transfer Learning strategy was developed and investigated, taking advantage of the multimodal data refered in section 3.1.1 using fine-tuning, ie, the Transfer Learning strategy was designed from one modality to another.

Transfer Learning as introduced in section 1.1, aims to solve one problem in the base domain and transferring the knowledge gained to a different but related task, in the target domain, being fine-tuning it's most common application in deep learning.

The deep Transfer Learning structure used in this work is exemplified schematically in figure 3.4.

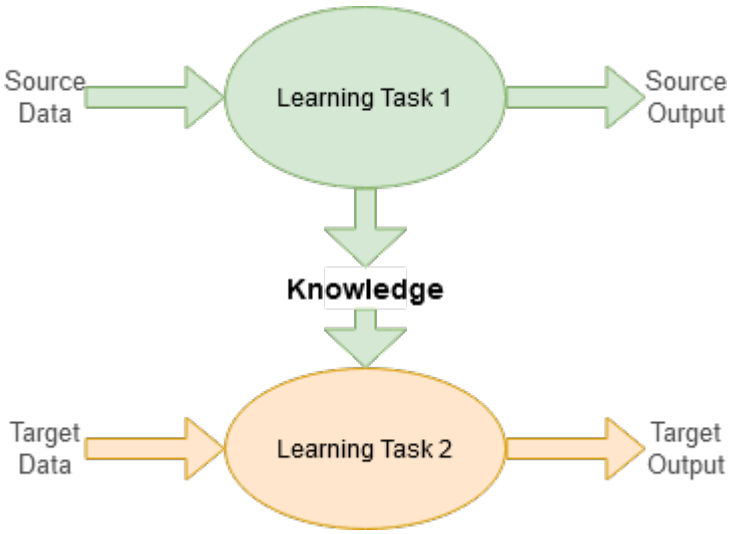


Figure 3.4: Representation of the learning process of Transfer Learning in which the source data is composed of brain scans from one modality and the target data is composed of brain scans from another modality.

The transfer of knowledge between modalities is a method that attempts to explore complementary information between modalities. The performance of this model increases if the base task and target task are similar [62]. Thus, using cross-modal Transfer Learning is expected to yield good results, given the similarity between tasks.

In terms of notation, according to [57] a domain can be represented by $\mathcal{D} = \{\chi, P(X)\}$, in which χ represents the feature space and $P(X)$ the marginal probability distribution where $X = \{x_1, ..., x_n\} \in \chi$. A task can be represented by $\mathcal{T} = \{y, f(x)\}$ in which y represents the labels and $f(x)$ the prediction function.

Transfer Learning can be defined as given a certain learning task $\mathcal{T}_t$ based on target domain $\mathcal{D}_t$, with the help from a source a domain $\mathcal{D}_s$ for the corresponding learning task $\mathcal{T}_s$, with the objective to improve the performance of predictive function $f_{\mathcal{T}}(\cdot)$ for learning task $\mathcal{T}_t$ by discover and transfer latent knowledge from $\mathcal{D}_s$ and $\mathcal{T}_s$, where $\mathcal{D}_s \neq \mathcal{D}_t$ and/or $\mathcal{T}_s \neq \mathcal{T}_t$. In most of the cases, the size of $\mathcal{D}_s$ is much

larger than the size of $\mathcal{D}_t$, $\mathcal{N}_s \gg \mathcal{N}_t$.

Deep Transfer Learning (DTL) is then defined as a Transfer Learning task within a deep neural network. Deep Transfer Learning can be classified into several categories [57], one of them being Network-based DTL. Network-based DTL is the Transfer Learning category explored and described in section 1.1 and refers to partially reuse a network pre-trained in the source domain, including its network structure and weights, and transfer it to a deep neural network in the target domain. The transferred sub-network can be updated using a fine-tuning strategy. This process is illustrated in figure 3.5.

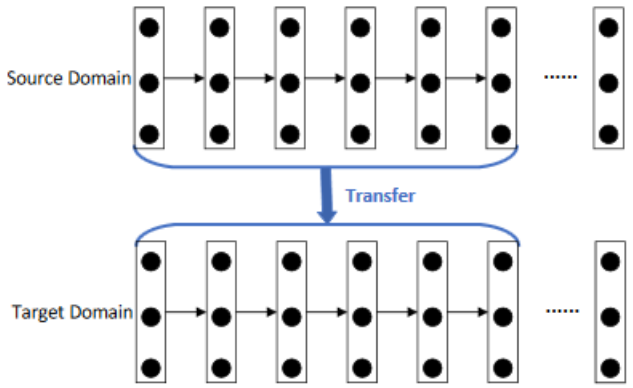


Figure 3.5: Illustration of a network-based DTL process (adapted from [57]).

3.1.7 Multi-Task Learning

Besides Transfer Learning, two other common methods were explored to combine two neuroimaging modalities. These two approaches are based on the subfield of multi-task learning (MTL) and try to explore commonalities and differences between GM and PET data to improve learning performance when compared to training the models separately, while dealing with lack of data, in the same way as Transfer Learning, where the learning of one task can as well benefit from the learning of other tasks.

MTL [58] is a machine learning method that aims to solve multiple tasks at the same time, utilizing correlated information among different tasks to improve the learning of each task.

Although MTL and TL can be related, in the sense that both aim to perform a new task by exploiting knowledge acquired when solving previous tasks, in what is called the learning-to-learn problem, in TL the generalization of the main task is improved with the extra information provided by the learning of an auxiliary task, whereby in MTL, all tasks are treated equally.

3.1.7.1 Joint training of both modalities in the same deep learning network

Using the GM and PET modalities, the feature space was composed by volumes of both modalities, which were taken as input together for training a single model, thus concatenating modalities in an early fusion mode. Classification of data from each modality is treated as a task, which means that the model learns the tasks jointly and share information among different tasks, which can lead to improvements in the learning performance.

In this case, as defined in [42] if the single model is represented by h then, the final prediction can be written as

$$p = h([v_1, ..., v_m]), \tag{3.9}$$

where p represents the model's prediction probability vector and v_m is a vector that represents the input data related with modalities $m = 1, 2, ..., M$, which in the case of having two modalities will lead to $M = 2$.

This method is one of the simplest methods to implement to combining modalities, involving only one model, and allows the use of all available data for analysis. The downside of this approach is that both modalities need to be similar to each other, well aligned and the model needs to be well suited for both modalities.

3.1.7.2 Concatenation of two deep learning models trained on separate modalities

This approach was developed by using pre-trained models on separate modalities with the same number of input volumes in each modality and combine the last layers in order to allow the sharing of information between tasks and produce a single classification output.

If h_i is the model used in modalities $i = 1, ..., M$, then the final prediction p, which represents the model's prediction probability vector, is given by

$$p = F(h_1(v_1), ..., h_m(v_m)), \tag{3.10}$$

where F is the fusion mechanism (neural network) that aggregates the representations of the pre-trained models and computes the final output.

This late fusion method allows the use of different models in different modalities, while enabling the integration of distinct data types (e.g. images and cognitive scores) thus allowing more flexibility. On the other hand, using this method, the number of input samples for each task must be the same. Besides,

because this approach operates in inferences and not the raw inputs, it's not effective at modeling interactions between modalities at input-level.

3.1.8 Model selection and performance evaluation

Usually a model is trained using a training set and evaluated using a separate test set. If the hyper-parameters were chosen based on the estimation of a model's performance on the training set, this estimate would be overoptimistic and the selected model would be too adjusted to the training set. Using another disjoint set as a validation set, it is possible to learn a model in the training set, evaluate the performance of the model at a given time and select the best configuration using the validation set and measure the performance of this selected model in a separate test set. The best model can be chosen using early stopping: interrupting training when the validation loss is no longer improving and using this model as the best one.

Cross-validation (CV) is an approach where all available data can be used for training and testing in different splits, reducing the waste of data. Using CV, the data is split into k smaller folds, usually $k = 5$ or 10. A model is trained using $k - 1$ of the folds as training data and tested on the remaining fold, as exemplified in figure 3.6. The final performance measure reported by the k-fold cross validation is the average of the values computed in each iteration. The training set can be further split into training and validation set, to chose the best model configuration through early stopping.

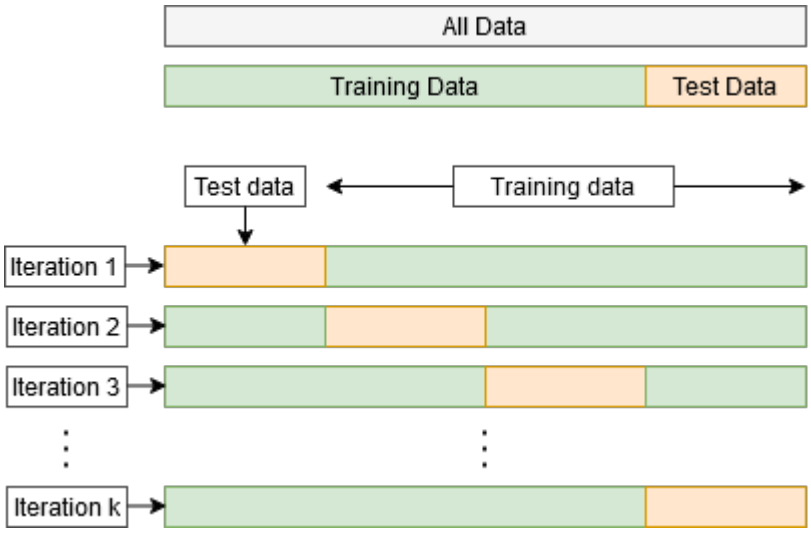


Figure 3.6: K-folds cross validation.

A model's performance in a certain task can be evaluated through a set of metrics, which are usually accuracy, sensitivity (also known as recall), specificity, precision and F1 score, which are expressed by the equations

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.11}$$

$$Sensitivity/Recall = \frac{TP}{TP + FN} \tag{3.12}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3.13}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.14}$$

$$F1 \quad score = 2 \times \frac{precision \times recall}{precision + recall} \tag{3.15}$$

where TP is the number of true positives, TN the number of true negatives, FP the number of false positives and FN the number of false negatives.

These metrics were used to compare the performance between the different learning methods explored, mentioned in sections 3.1.5, 3.1.6, 3.1.7 in the binary classification problem AD vs NC. In this sense, the true positives corresponds to a correctly diagnosed AD subject, a true negative corresponds to a correctly diagnosed NC subject, a false positive means that a subject was misclassified as AD and a false negative represents a subject misclassified as CN.

Accuracy can then be defined as the number of correct predictions made as a ratio of all predictions made, independently of the class, while sensitivity (or recall) can be defined as the number of correct positive results returned by the machine learning model, in this case, the number of subjects correctly diagnosed with AD. Specificity is a measurement of the proportion of correct negative results, in this case, the number of subjects correctly classified as NC and precision can be seen as the fraction of relevant results among the positive results, which means the fraction of subjects correctly classified with AD among the set of subjects that were classified with AD. F1 score, which is mathematically represented in equation 3.15 displays the harmonic mean of precision and recall, so if F1 score is high, both precision and recall of the classifier indicate good results. This allows for the comparison of the performance of two classifiers using just one metric while making sure that the models are working correctly.

All these metrics provide important information for the binary classification problem at hand and were used to evaluate the model's performance. Box plots are another important indicator that was used, providing a visual context on the variability or dispersion of the data. The distribution of data is displayed based on a five number summary, composed by a "minimum", first quartile (Q_1), median, third quartile (Q_3) and a "maximum", which are illustrated in figure 3.7. The median (Q_2) represents the middle value of the dataset. The first quartile (Q_1) is the median of the lower half of the dataset and the third quartile (Q_3), the median of the upper half of the dataset. The interquartile range is the distance

between upper and lower quartiles $IQR = Q_3 - Q_1$. The ”maximum” and ”minimum” values define the highest and lowest data points excluding the outliers. A data point p is an outlier if it follows the condition

$$p > Q_3 + 1.5 \times IQR \quad \vee \quad p < Q_1 - 1.5 \times IQR. \tag{3.16}$$

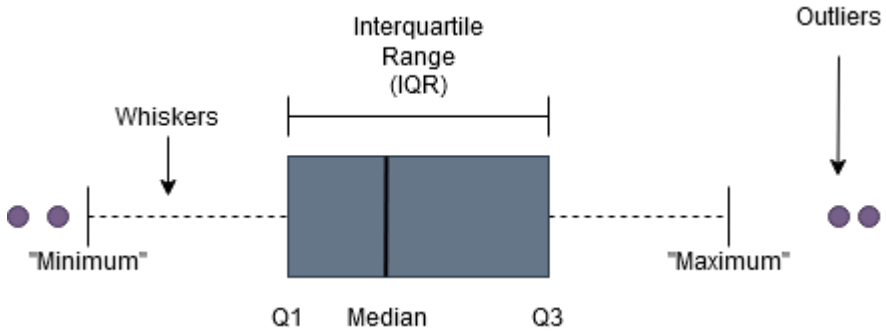


Figure 3.7: Elements of a box plot.

3.1.9 Visualization of heatmaps of intermediate activations

The representations learned by CNNs can be easily visualized, once they are representations of visual concepts. A wide variety of techniques have recently been developed for visualizing and interpreting these representations, one of them being visualizing intermediate activations. This technique consists of displaying the feature maps that are output by the convolution and pooling layers in a network (activations), given a certain input. For 3D CNNs, this results in 4D feature maps, in which the fourth dimension corresponds to the number of channels. This is useful specially for higher layers' activations, which encode more information about the class of the image.

This way, understanding which parts of an input volume led the CNN to the final classification decision, can be done by computing the filter maps output by higher convolutional layers given an input sample. In the 4D output, the filters across all dimensions can be averaged, generating an heatmap of intermediate activations, in order to analyze which parts of the image are given more importance by the convnet, and detect whether the network is correctly learning patterns.

3.2 Experimental Setup

This section describes the experiments conducted as well as the hardware and images available to perform such experiments.

3.2.1 Database

The data used in the experiments came from the ADNI database [2]. The Alzheimer's Disease Neuroimaging Initiative (ADNI) is a longitudinal multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of Alzheimer's disease. Determination of sensitive and specific markers of very early AD progression is intended to aid researchers and clinicians to develop new treatments and monitor their effectiveness, as well as lessen the time and cost of clinical trials. ADNI is the result of efforts of many coinvestigators from a broad range of academic institutions and private corporations, and subjects have been recruited from over 50 sites across the U.S. and Canada, enabling the sharing of data between researchers around the world since its launch in 2004.

A detailed description on how the MRI and PET datasets were acquired can be found in the public ADNI website [2].

3.2.2 Subjects

In this study, 1.5T Magnetic Resonance images and FDG-PET images were acquired from subjects evaluated during a 24 month period. Evaluations of their mental state and collection of brain scans were performed at a baseline month, and 12 and 24 months after the first evaluation. The number of subjects (n) as well as their gender, age, mini-mental state examination (MMSE) and clinical dementia rating (CDR) are shown in table 3.1 for each evaluation time and for AD and NC subjects for each neuroimaging modality.

It's worth noting that there are subjects who don't go through the complete 24 months of observations. There are also subjects whose images are present in only one of the modalities at a given time. In total there are 383 PET volumes from 133 subjects (58 AD and 75 NC) and 648 GM volumes from 316 subjects (144 AD and 172 NC) in the dataset. Considering that some subjects have PET and GM volumes, the total number of subjects evaluated is 354 from which 152 are classified with AD and 202 are NC.

By comparing age and gender information on both modalities, it can be concluded that each subset isn't biased by age or gender of the patients. This is true also when comparing patients evaluated on the same modality, but with different disease status, which ensures that differences in the classifier's

Table 3.1: Subject information for PET and GM modalities. The number of subjects is denoted by n. MMSE and CDR correspond to mini-mental state examination and clinical dementia rating, respectively.

		PET		GM	
Follow up period		AD	NC	AD	NC
0 months		n = 58	n = 75	n = 107	n = 124
	Gender (F/M)	24/34	26/49	52/55	65/59
	Age (mean $\pm$ SD)	76.0 $\pm$ 6.6	76.0 $\pm$ 4.6	75.9 $\pm$ 7.6	76.3 $\pm$ 5.2
	MMSE (mean $\pm$ SD)	23.5 $\pm$ 2.0	29.1 $\pm$ 1.0	23.3 $\pm$ 2.0	29.3 $\pm$ 0.9
	CDR (mean $\pm$ SD)	0.8 $\pm$ 0.2	0.0 $\pm$ 0.0	0.8 $\pm$ 0.3	0.0 $\pm$ 0.0
12 months		n = 54	n = 74	n = 98	n = 135
	Gender (F/M)	23/31	26/48	50/48	73/62
	Age (mean $\pm$ SD)	76.8 $\pm$ 6.6	77.1 $\pm$ 4.7	77.5 $\pm$ 7.2	77.2 $\pm$ 5.2
	MMSE (mean $\pm$ SD)	21.0 $\pm$ 4.2	29.1 $\pm$ 1.2	21.3 $\pm$ 4.5	29.2 $\pm$ 1.2
	CDR (mean $\pm$ SD)	1.0 $\pm$ 0.5	0.0 $\pm$ 0.2	0.9 $\pm$ 0.5	0.0 $\pm$ 0.2
24 months		n = 53	n = 69	n = 69	n = 115
	Gender (F/M)	24/29	25/44	36/33	61/54
	Age (mean $\pm$ SD)	78.6 $\pm$ 6.6	77.8 $\pm$ 4.5	78.7 $\pm$ 6.6	78.2 $\pm$ 5.2
	MMSE (mean $\pm$ SD)	19.9 $\pm$ 5.1	29.0 $\pm$ 1.1	19.5 $\pm$ 5.8	29.0 $\pm$ 1.2
	CDR (mean $\pm$ SD)	1.2 $\pm$ 0.7	0.1 $\pm$ 0.2	1.2 $\pm$ 0.7	0.0 $\pm$ 0.2

outcome aren't due to age differences, but due to different disease patterns. All the patients that were diagnosed with AD in month 0 remain with that prognostic during the whole 24 months and the same happens with NC subjects.

3.2.3 Data Division

The training and evaluation of the models was performed using 5 fold cross-validation, as described in section 3.1.8. Furthermore, a validation set was used to perform early stopping. The data split is performed according to each subject: In a dataset containing PET or GM images, 20% of the subjects are used for test and from the remaining 80%, 20% are used for validation and 60% for the training set, while assuring that different images from the same subjects are stored in the same partition. The "information leak" mentioned in section 3.1.1.2 had to be avoided when assigning images from months 12 and 24 to an existing partition. This is done by first assigning the subjects' images from the first evaluation period, month 0, from each modality, to the corresponding training or test sets. Then, the images corresponding to future scans of the same subjects will be assigned to the same partitions as the corresponding first month images. This way, all the images from the same subject in each modality are present in the same partition.

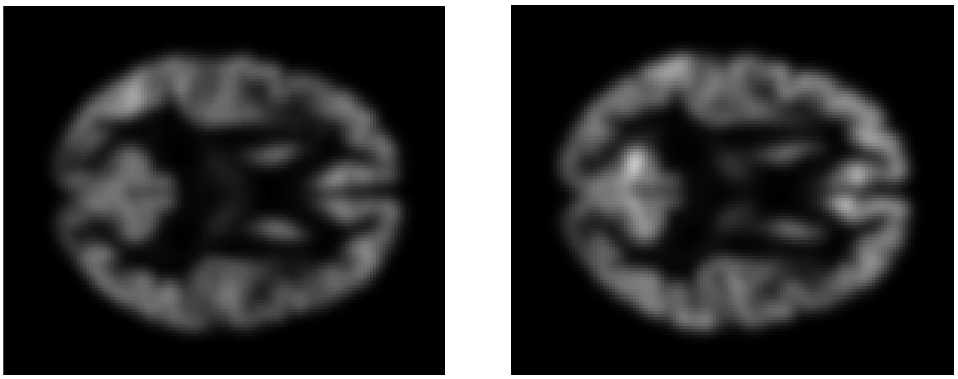
A very important safeguard was in assuring that there wouldn't be any PET images from a patient used in training in the base domain, that could be used in testing in the target domain in the Transfer Learning approach, or vice-versa. This was done by using images from subjects that appear on both modalities in the same partition in each fold. The other subjects that appear on only one modality could be used on either set. This division of data was done in an early phase, in order to allow the use of the same images in the same sets for the several experiments conducted, so that the results of each method could be compared between each other.

3.2.4 Pre-Processing

PET and MR images had already been subject to a series of preprocessing steps performed by the ADNI researchers. The MR images from ADNI were corrected for gradient non-linearity. The B1 non-uniformity procedure is applied to correct non-uniformities in the image's intensity and the N3 histogram peak sharpening is applied to mitigate residual non-uniformities. The PET images were co-registered to each other and averaged and aligned so that the anterior commissure (AC) and posterior commissure (PC) were in the same axial plane (AC-PC alignment). Then the image is resampled using a 1.5 mm grid and filtered so that it's resolution is similar to the lowest resolution scanners used by ADNI.

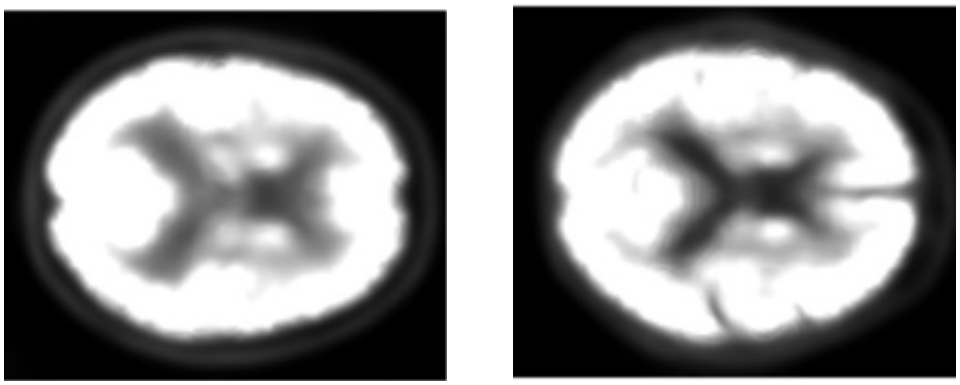
Furthermore, the images retrieved from the ADNI database were warped into the MNI standard space as described in [46]. In this process, to the MR images was performed skull stripping, segmentation into GM and WM, producing gray and white-matter probability maps which were also smoothed with a Gaussian filter. The resulting PET images were normalized using the Yakushev normalization procedure.

Examples of slices of GM and PET images in the horizontal plane are presented in figures 3.8 and 3.9. The presented slices correspond to the same individual in GM and PET modalities, for AD and NC categories. Each slice in the dataset has 145x121 pixels and each volume is composed by 121 slices, that is, each volume has 2 122 945 voxels.



(a) NC (b) AD

Figure 3.8: GM slices from the ADNI dataset for a NC and AD brain at baseline.



(a) NC (b) AD

Figure 3.9: PET slices from the ADNI dataset for a NC and AD brain at baseline.

3.2.4.1 Crop

The images from the brain scans include the area surrounding the brain, as can be seen from figures 3.8 and 3.9, which doesn't present any relevant information for the classification task. This area was cropped, which resulted in a significant reduction in the size of the feature vectors.

Since the images were registered according to the MNI-152 template, with the same dimension as the PET and GM volumes, only the area inside the brain was considered, which is represented by the area in white in figure 3.10, where there are shown representations of the MNI brain mask in several planes.

The volumes' dimension after the cropping was 104x122x98, that is, each volume became composed of 1 243 424 voxels for both PET and GM.

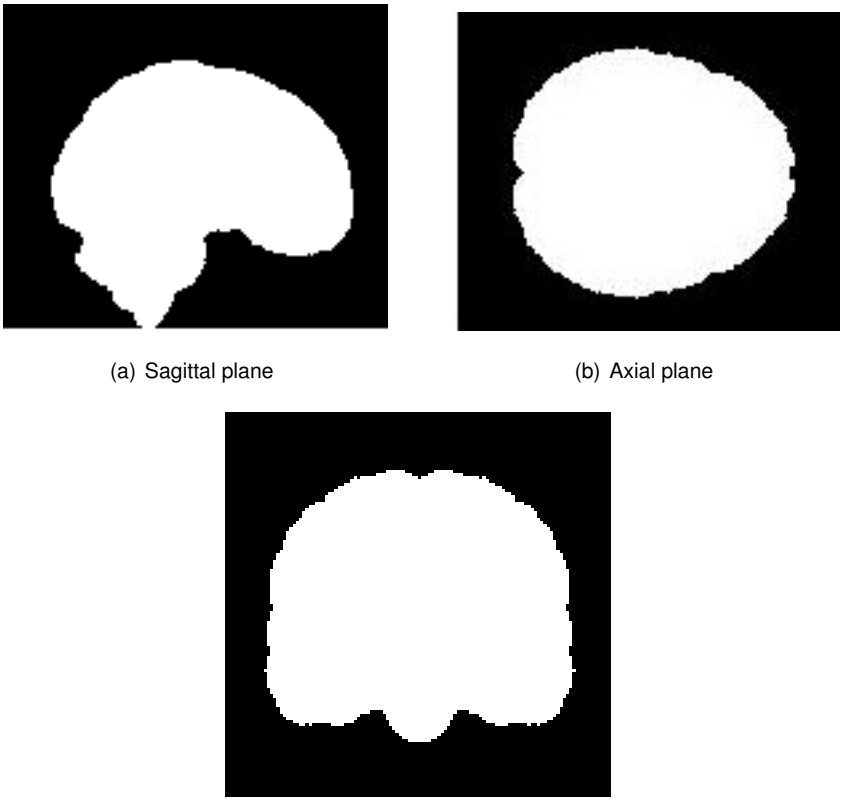
3.2.4.2 Feature Normalization

Besides the normalization process that had been previously applied, as described in [46], the voxel intensities in each volume slice were mapped to the range [-1, 1], as described in section 3.1.2.1.

In each fold, the maximum of the modulus of the volumes from PET and GM in the training set was computed. Then all the volumes in the of GM and PET training, validation and test set were divided by this value, which was computed separately for each modality and after the images from months 12 and 24 were added to the training set.

3.2.5 Deep Learning Network

Tables 3.2 and 3.3 present the final CNN-LSTM and 3D CNN architectures used for the AD vs NC classification problem. Before these configurations were achieved, others were tested, where parameters



(a) Sagittal plane

(b) Axial plane

(c) Coronal Plane

Figure 3.10: Representations of the MNI brain mask in the sagittal, axial and coronal sections of the brain.

such as the number of layers, number of units in each layer, filter's shape, dropout or batch normalization were tested.

Each network was trained using all the available PET and GM data, displayed in table 3.1, including longitudinal data from the subjects, as described in section 3.1.1.2, which achieved better results than using only data from the first follow up month, or only one image per subject.

3.2.5.1 CNN-LSTM network architecture

As seen in table 3.2, each Convolution Block is made by a 2D CNN and a max-pooling layer. The output of the three convolution blocks is flattened, goes into the LSTM and then fed into a densely connected classifier network, with softmax activation, corresponding the output to the binary classification of AD vs NC.

The convolution layers use same padding, whereas max pooling layers use valid padding which in practice means that no padding is used and the operation is only applied to valid windows. The 'same padding' technique applies padding to the input image so that the output image has the same shape as the input. Strides of size 2x2 are used as the distance between two successive windows, meaning that

the output of the first convolutional layer or the max pooling layers is downsampled by a factor of 2 in the second and third dimensions.

The dropout regularization technique is used to reduce overfitting by randomly setting to zero a number of output features of a layer during training. This was introduced in the network by a Dropout layer and by the dropout parameter in the LSTM layer.

Table 3.2: Architecture of the CNN-LSTM network.

Layer Type	Layer Parameters	Filters/Units	Output Size
TimeDistributed(Conv2D)	Same Padding, Stride=(2,2), ReLU	3x3x32	104x61x49x32
TimeDistributed(MaxPooling2D)	window size=2x2, Stride=(2,2)	-	104x30x24x32
TimeDistributed(Conv2D)	Same Padding, ReLU	5x5x64	104x30x24x64
TimeDistributed(MaxPooling2D)	window size=2x2, Stride=(2,2)	-	104x15x12x64
TimeDistributed(Conv2D)	Same Padding, ReLU	5x5x128	104x15x12x128
TimeDistributed(MaxPooling2D)	window size=2x2, Stride=(2,2)	-	104x7x6x128
TimeDistributed(Flatten)	-	-	104x5376
Dropout	50% Dropout	-	104x5376
LSTM	Tanh, 50% Dropout	128	128x1
Dense	Softmax	2	2x1

3.2.5.2 3D CNN network architecture

The 3D CNN model is also made of three convolution blocks, as shown in table 3.3: The first two convolution blocks are made by a 3D CNN layer, a 3D max-pooling layer and two batch normalization layers. The third convolution block only has a batch normalization layer. The output of the convolution blocks is then flattened and fed into a fully connected classifier network, which results in the classification of AD vs NC.

In the convolutional layers and max pooling layers used in the 3D CNN architecture, valid padding is used. Strides of size 1x1x1 are used as the distance between two successive windows for the 3D convolutional layers, whereas a stride of size 1x2x2 is used for the max pooling layers so that the output feature maps are downsampled by a factor of 2 in the second and third dimensions.

Batch normalization [33] was used to standardize the layer's inputs to have zero mean and unit variance in each training mini-batch, accelerating training and providing better generalization.

Table 3.3: Architecture of the 3D CNN network.

Layer Type	Layer Parameters	Filters/Units	Output Size
Conv3D	ReLU, Stride=(1,1,1)	3x3x3x8	102x120x96x8
BatchNormalization	-	-	102x120x96x8
MaxPooling3D	window size=2x2x2, Strides=(1,2,2)	-	101x60x48x8
BatchNormalization	-	-	101x60x48x8
Conv3D	ReLU, Stride=(1,1,1)	3x3x3x16	99x58x46x16
BatchNormalization	-	-	99x58x46x16
MaxPooling3D	window size=2x2x2, Strides=(1,2,2)	-	98x29x23x16
BatchNormalization	-	-	98x29x23x16
Conv3D	ReLU, Stride=(1,1,1)	3x3x3x32	96x27x21x32
BatchNormalization	-	-	96x27x21x32
MaxPooling3D	window size=2x2x2, Strides=(1,2,2)	-	95x13x10x32
Flatten	-	-	395200
Dense	-	64	64x1
BatchNormalization	-	-	64x1
Dense	-	64	64x1
BatchNormalization	-	-	64x1
Dense	Softmax	2	2x1

3.2.6 Transfer Learning

Two methods were tested regarding Transfer Learning: Either pre-training a deep learning network with GM data as the base domain and fine-tuning using the PET dataset as the target domain (TL GM-PET), or instead pre-training a network with PET data and fine-tuning using the GM dataset (TL PET-GM).

Independently of using TL GM-PET or PET-GM, the number of fine-tuned layers and number of top densely connected layers replaced was also tested. Several possibilities regarding the number of replaced layers for each network are illustrated in figure 3.11. For the CNN-LSTM, several configurations were tested: replacing the top four layers (Flatten, Dropout, LSTM and Dense) with the same layers having new randomly initialized weights, replacing the last Dense layer, or not replacing any layer. For the 3D CNN, the last six layers were replaced (Flatten, Dense, BatchNormalization, Dense, BatchNormalization and Dense), or the last Dense layer or no layer was replaced. Choosing one of these configurations, any number of the remaining layers can then be fine-tuned.

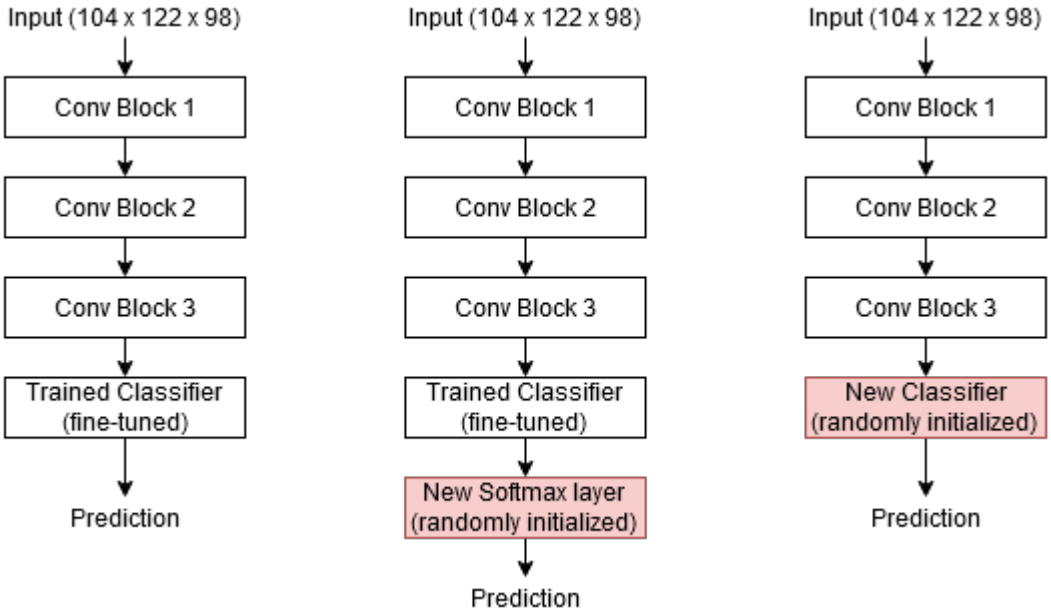


Figure 3.11: Transfer Learning configurations tested for both deep learning models in terms of number of replaced layers with new randomly initialized layers. Any number of the remaining layers can then be fine-tuned.

To perform these experiments, pre-trained models trained separately with PET and GM data were used. Then the last layers of the pre-trained network were replaced by new layers with randomly initialized weights. Some number of the remaining layers were chosen to be fine-tuned and the others remained frozen (their weights could not be adjusted) and the unfrozen layers were trained jointly with the added part. The images used for training and testing were the same images used to train and test each network from scratch.

The results of all the experiments performed are presented in appendix A and the best results from all the experiments from TL GM-PET and TL PET-GM are presented in section 4.

3.2.7 Multi-Task Learning

3.2.7.1 Joint training of both modalities in the same deep learning network

The data used to apply this method on the AD vs NC classification problem is described in table 3.4 and corresponds to the entirety of available data of PET and GM modalities joined in the training of a single model.

Table 3.4: Number of images (n) for different evaluation periods and both imaging modalities using images from both modalities jointly in the same deep learning network and the corresponding number of subjects.

	PET		GM	
Follow up period	AD	NC	AD	NC
0 months	n = 58	n = 75	n = 107	n = 124
12 months	n = 54	n = 74	n = 98	n = 135
24 months	n = 53	n = 69	n = 69	n = 115
Total	n = 165	n = 218	n = 274	n = 313
Subjects	58	75	144	172

3.2.7.2 Concatenation of two deep learning models trained on separate modalities

To apply this method to the AD vs NC classification problem, it was necessary to use the same number of input samples from each modality. From the complete set of images available, in each time window, only subjects with both modalities available were used, so that each input pair could have GM and PET volumes from the same subject and the information concatenated could belong to the same person. The available number of images for each modality is shown in table 3.5. The data was split similarly to what is described in section 3.2.3, but subjects that didn't have volumes of both modalities were discarded.

Table 3.5: Number of images (n) for different evaluation periods and both imaging modalities using concatenation of two models trained on separate modalities and the corresponding number of subjects.

	PET		GM	
Follow up period	AD	NC	AD	NC
0 months	n = 50	n = 45	n = 50	n = 45
12 months	n = 43	n = 36	n = 43	n = 36
24 months	n = 27	n = 20	n = 27	n = 20
Total	n = 120	n= 101	n = 120	n = 101
Subjects	50	45	50	45

Several topologies were tested regarding the number of layers and units in each layer, from which the final model was chosen (appendix B). For the concatenation of two CNN-LSTM networks, the resulting network concatenates the outputs of the last layer of both CNN-LSTM networks and adds two Fully Connected layers on top of the concatenation, to perform the final decision, as shown in table 3.6.

Table 3.6: Architecture of the fusion network for concatenation of two CNN-LSTM networks trained on separate modalities.

Layer Type	Layer Parameters	Filters/Units	Output Size
Concatenate	-	-	4x1
Dense	-	32	32x1
Dense	Softmax	2	2x1

For the concatenation of two 3D CNN networks trained separately on the MRI and PET modalities, the resulting network concatenates the feature representations of the penultimate layer of both 3D CNN networks and adds three Fully Connected layers on top of the concatenation, to perform the final decision, as shown in table 3.7. For both types of deep learning architectures, only the added layers were trained, while the weights from the pre-trained layers were not updated.

Table 3.7: Architecture of the fusion network for concatenation of two 3D CNN networks trained on separate modalities.

Layer Type	Layer Parameters	Filters/Units	Output Size
Concatenate	-	-	128x1
Dense	-	32	32x1
Dense	-	32	32x1
Dense	Softmax	2	2x1

3.2.8 Implementation Details

The deep learning methods are implemented using Keras [17] with a TensorFlow [8] 2.3.0 backend. Keras processes 3D convolutions as 4D tensors of shape (nr_frames, image_height, image_width, image_channels). The first dimension corresponds to the number of slices in a volume of one modality. For a black and white image, the number of image_channels is 1, ie gray levels.

For the CNN-LSTM model, the Keras TimeDistributed was used, which allows to distribute layers of a 2D CNN across an extra dimension. For the 3D CNN, the Keras Conv3D layer was used, which performs spatial convolution over volumes.

All the experiments were performed in Google Colab [4], which is a colaborative platform that allows the writing and execution of Python in the web browser, with free access to GPUs. Colab offers several types of GPUs, namely Nvidia K80s, T4s, P4s and P100s, but the user can't choose a type of GPU to connect at a given time.

When training the CNN-LSTM model, either using Transfer Learning or the multi-task learning approaches, the mini-batch sizes used were 64 for training and 32 for validation and test, since some folds didn't have 64 images in the validation or test set. This choice of mini-batch sizes was also used for the concatenation of two 3D CNN networks, since only the weights from added layers were updated. For the 3D CNN models, except the concatenation model, it was used a mini-batch size of 16 for the training, validation and test, once training these networks required more memory usage.

5 fold cross-validation was used, in which a portion of the training data was reserved as a validation set to perform early stopping. This way, the model is trained until the loss in the validation set stops improving.

Since the dataset suffers from class imbalance (in a given fold is more frequent to have images corresponding to NC subjects than to AD subjects) a weighted training strategy was applied, in which samples belonging to the class with the majority of data in the training set in a given fold are given a weight equal to one and samples from the other class are given a weight equal to $\frac{N}{M} > 1$, where N is the number of samples from the class with most data and N is the number of samples from the class with less samples, penalizing the misclassification made by the minority class, which has a higher class weight.

The networks were trained using the Adam optimizer [36] with a initial learning rate of 0.001, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The accuracy metric was evaluated during training and the chosen loss function was binary cross-entropy, which computes the cross-entropy loss between true labels and the predicted label.

The visualization of heatmaps of intermediate activations was implemented using Jupyter notebooks [5], which allows for the use of the interactive mode in the matplotlib library, which was used to display the outputs over the several slices of a volume interactively.

4 Results and Discussion

The results obtained through the experiments detailed in section 3.2 are presented in this section. The performance of the models trained from scratch, Transfer Learning methods and multi-task methods are evaluated for the AD vs NC classification problem. The cross-modal Transfer Learning approach is compared with the other approaches, as well as with state of the art approaches. Finally, a visual comparison is made between the heatmaps produced by the output filters of intermediate layers and known biological changes in the brain of Alzheimer's Disease patients.

4.1 Transfer Learning vs training from scratch

Tables 4.1 and 4.2 present the results obtained for both deep learning architectures developed and the best Transfer Learning results obtained for the models pre-trained with GM data and fine-tuned with PET data (TL GM-PET) and models pre-trained on the PET modality and fine-tuned on the GM modality (TL PET-GM).

The complete results from the set of experiments performed regarding the number of fine-tuned layers and number of replaced top layers in the Transfer Learning approach used are presented in appendix A.

Table 4.1: CNN-LSTM results using GM and PET modalities individually and the best TL PET-GM and TL GM-PET results obtained. Mean and SD of the 5 folds.

Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
GM	0.679	0.111	0.665	0.122	0.707	0.222	0.752	0.222	0.695	0.102
PET	0.829	0.034	0.727	0.058	0.915	0.053	0.862	0.053	0.784	0.023
TL PET-GM	0.816	0.060	0.752	0.086	0.888	0.056	**0.893**	0.056	0.815	0.065
TL GM-PET	**0.861**	0.033	**0.806**	0.087	**0.916**	0.046	0.872	0.046	**0.831**	0.025

Comparing the results relative to each model trained from scratch, from tables 4.1 and 4.2, it can be concluded that the CNN-LSTM model outperforms the 3D CNN model for the PET modality and the 3D CNN performs better than the CNN-LSTM model for GM data. These results can be explained taking into account the number of training samples in each modality and the complexity and number of parameters of each model. The CNN-LSTM model requires a total of 3,075,330 training parameters (weights and biases), while the 3D CNN requires 8 times more parameters (25,315,538), which means that the CNN-LSTM model needs less training data, but on the other hand, has less ability to capture inter-slice information than the 3D CNN, since it performs convolutions on slices and only takes into

Table 4.2: 3D CNN results using GM and PET modalities individually and the best TL PET-GM and TL GM-PET results obtained. Mean and SD of the 5 folds.

Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
GM	0.839	0.031	0.824	0.035	0.857	0.061	0.877	0.061	0.849	0.034
PET	0.761	0.058	0.725	0.083	0.802	0.109	0.737	0.109	0.717	0.072
TL PET-GM	**0.864**	0.023	**0.837**	0.054	0.896	0.028	**0.908**	0.028	**0.870**	0.031
TL GM-PET	0.851	0.038	0.778	0.072	**0.909**	0.068	0.865	0.068	0.814	0.041

account the axial plane. Regarding the CNN-LSTM model, a notable performance is achieved when the model is trained on the PET modality, even having less training data than GM, thus being PET the most discriminative modality against AD changes in the brain. In the 3D CNN model, the best results are obtained by training the model using GM data. Since there is more GM training data available, it is easier for the model to learn characteristic patterns of AD from the inputs, given the high number of trainable parameters.

Transfer Learning shows performance improvements compared with training each model from scratch in all the metrics used for evaluation. Using the CNN-LSTM model, fine-tuning with PET data was more effective than using GM data. These results were obtained by replacing the last layer (softmax) and unfreezing all the remaining layers for GM-PET or unfreezing 7 of the remaining layers (corresponding to the densely connected classifier and the second convolution block) for PET-GM. In this case, since the number of parameters is small, using the PET modality as target domain (which has less images but is more discriminative) has no risk of overfitting, hence the base features can be fine-tuned to the new task to improve performance [62].

Regarding the 3D CNN model, the results show fine-tuning with GM data to be more effective. The best results were obtained by replacing the whole densely connected classifier and unfreezing the last convolution block for PET-GM, while for GM-PET there were only fine-tuned 9 layers (the densely connected classifier and the top convolution block). In both cases, the best scenario had to do with fine-tuning the last convolution block, which limits the final TL model's performance. Although these methods achieve better performance than training a network from scratch, the large number of training parameters and lack of data in the target domain make this configuration optimal in terms of fine-tuning, instead of fine-tuning the whole network.

From the results shown in appendix A it is generally evident that initializing the convolutional layers with transferred weights improves classification just by itself, but the layers that constitute the densely connected classifier must be fine-tuned to the target task.

4.2 Transfer Learning vs joint training of both modalities in the same Deep Learning network

The comparison between our Transfer Learning approach and the approach based on training both modalities in the same deep learning network, as described in section 3.1.7.1, is reported in table 4.3 for the CNN-LSTM model and in table 4.4 for the 3D CNN model. The results relative to the early fusion model are expressed according to the nature of the testing set: whether it contained only PET or GM images or contained images from both modalities.

Table 4.3: Results for the model trained using both modalities jointly in the same CNN-LSTM network and Transfer Learning methods. Mean and SD of the 5 folds.

Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
GM	0.778	0.042	0.669	0.092	0.906	0.041	0.901	0.401	0.764	0.060
PET	0.808	0.051	0.731	0.065	0.867	0.106	0.809	0.106	0.762	0.053
PET and GM	0.821	0.031	0.761	0.009	0.881	0.057	0.869	0.057	0.811	0.020
TL PET-GM	0.816	0.060	0.752	0.086	0.888	0.056	**0.893**	0.056	0.815	0.065
TL GM-PET	**0.861**	0.033	**0.806**	0.087	**0.916**	0.046	0.872	0.046	**0.831**	0.025

Table 4.4: Results for the model trained using both modalities jointly in the same 3D CNN network and Transfer Learning methods. Mean and SD of the 5 folds.

Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
GM	0.840	0.028	0.768	0.079	0.924	0.033	0.928	0.033	0.838	0.042
PET	0.832	0.083	0.828	0.063	0.832	0.123	0.792	0.123	0.806	0.084
PET and GM	0.852	0.020	0.824	0.064	0.875	0.064	0.878	0.064	0.848	0.026
TL PET-GM	**0.864**	0.023	**0.837**	0.054	0.896	0.028	**0.908**	0.028	**0.870**	0.031
TL GM-PET	0.851	0.038	0.778	0.072	**0.909**	0.068	0.865	0.068	0.814	0.041

From the results shown in tables 4.3 and 4.4, it can be observed that the models obtained results close to those of the Transfer Learning approaches. The fact that all the available data from both modalities was used in the training of the models, reduces the need for Transfer Learning, by reducing overfitting during training. With this increase in the number of training samples, there is a significant increase in the average training time. In Transfer Learning, the training in the target domain doesn't require as many training samples and has less trainable parameters, which makes training faster.

In TL, the weights are updated according to the modality in the target domain, while in this early fusion approach, the same deep learning model needs to be well suited for both modalities. This flexibility of the TL approach enables performance improvements, while reducing training speed and increasing robustness, by exploiting knowledge acquired from the auxiliary task.

When comparing these results with the results from tables 4.1 and 4.2, it can be concluded that this multi-task approach outperforms training each modality separately for the 3D CNN model. In this sense, the network can take advantage of the shared information between modalities at input-level, given the similarities between the input modalities.

4.3 Transfer Learning vs model concatenation

The comparison between our Transfer Learning approach and the approach based on the concatenation of two networks trained on separate modalities is reported in table 4.5 for the CNN-LSTM model and table 4.6 for the 3D CNN model. The data used in the concatenated model is composed by a reduced dataset which contains subjects that appear on both modalities at the same time instants, as described in section 3.2.7.2.

For a better comparison, the CNN-LSTM and 3D CNN models trained on separate modalities were tested using the same number of images from each modality as this concatenated model, meaning that the GM images used in each fold in the testing set of the concatenation model were used for testing the models trained in the GM modality separately and the PET images used in each fold in the concatenated model were used for testing the models trained in the PET modality separately.

The complete results from the set of experiments performed regarding the number of layers and units in each layer in the fusion network are presented in appendix B.

Table 4.5: Results for the concatenation of two CNN-LSTM networks trained on separate modalities and Transfer Learning methods. Mean and SD of the 5 folds. (1) Results for the CNN-LSTM model evaluated on the reduced dataset which contains only subjects that appear on both modalities at the same time instants.

Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
PET and GM	0.762	0.075	0.556	0.105	0.925	0.093	0.856	0.093	0.670	0.120
TL PET-GM	0.816	0.060	0.752	0.086	0.888	0.056	**0.893**	0.056	0.815	0.065
TL GM-PET	**0.861**	0.033	**0.806**	0.087	**0.916**	0.046	0.872	0.046	**0.831**	0.025
GM (1)	0.610	0.128	0.458	0.194	0.726	0.303	0.612	0.303	0.496	0.162
PET (1)	0.730	0.075	0.518	0.087	0.910	0.092	0.821	0.092	0.631	0.106

Table 4.6: Results for the concatenation of two 3D CNN networks trained on separate modalities and Transfer Learning methods. Mean and SD of the 5 folds. (1) Results for the 3D CNN model evaluated on the reduced dataset which contains only subjects that appear on both modalities at the same time instants.

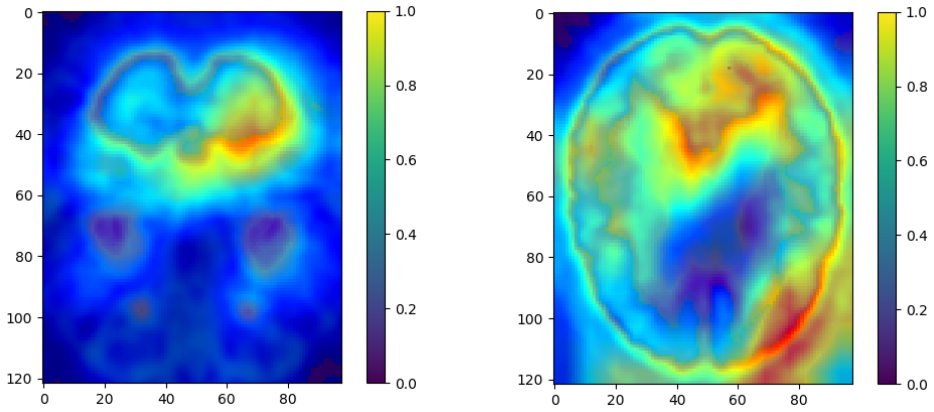
Modalities	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
PET and GM	0.755	0.049	0.599	0.109	0.885	0.447	0.803	0.045	0.681	0.088
TL PET-GM	**0.864**	0.023	**0.837**	0.054	0.896	0.028	**0.908**	0.028	**0.870**	0.031
TL GM-PET	0.851	0.038	0.778	0.072	**0.909**	0.068	0.865	0.068	0.814	0.041
GM (1)	0.716	0.051	0.586	0.100	0.817	0.096	0.733	0.096	0.644	0.092
PET (1)	0.729	0.163	0.670	0.140	0.781	0.187	0.732	0.187	0.695	0.180

From the analysis of tables 4.5 and 4.6, it can be concluded that the concatenation models achieve a better performance than the models trained from scratch, when comparing with data from the same subjects in the test set. A downside of this approach is that the model only accepts the same number of images from both modalities, corresponding to the same number of subjects, discarding relevant data in the datasets that can be used to improve the models. In Transfer Learning, this concern doesn't exist, which allowed the use of more GM data than PET data, since the images don't need to belong to the same subjects in both modalities. This aspect is not to be downplayed, once the number of available training images is usually smaller when using data from several modalities, which is often an obstacle in multimodal studies.

Contrarily to the early fusion method, this concatenation approach allows the use of two specific deep learning models for each neuroimaging modality, but the information shared among modalities is reduced. Furthermore, the concatenation of two deep learning models doesn't solve the problem of lack of training data: The pre-trained models in each concatenation network were trained from scratch with data from the GM and PET modalities. Although the combination of learned features from both modalities can improve classification by exploring commonalities and differences between both types of data, the pre-trained networks can suffer from overfitting, which affects the performance of the concatenation model.

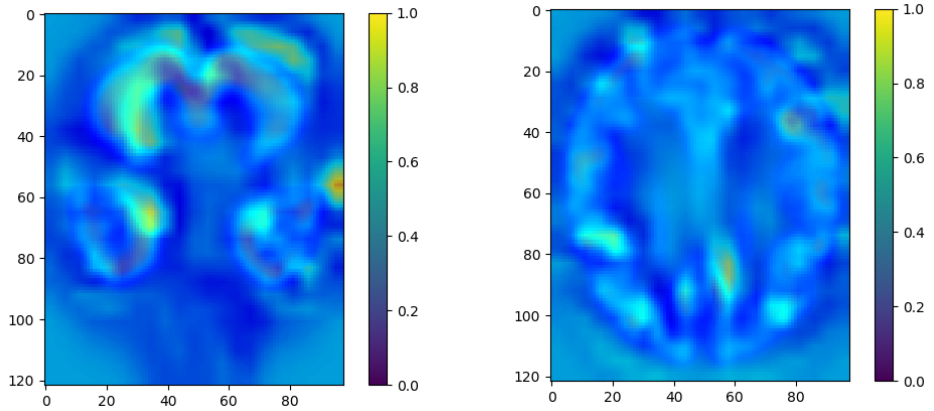
4.4 Visual comparison with biological changes in the brain

The output filters of the last convolutional layer of the CNN-LSTM model with TL GM-PET and the 3D CNN model with TL PET-GM are displayed in figures 4.1 and 4.2 for two selected slices in the axial plane. Samples of individuals in the test set, diagnosed with AD were taken as input by the network. The heatmaps were compared to regions of interest (ROI) known to be relevant for the diagnosis of AD.



(a) (b)

Figure 4.1: Heatmaps of the of the last convolutional layer activations for TL GM-PET using the CNN-LSTM model.



(a) (b)

Figure 4.2: Heatmaps of the last convolutional layer activations for TL PET-GM using the 3D CNN model.

In the heatmaps relative to the PET modality, high intensity values are located in the temporal region, as shown in figure 4.1 (a). In figure 4.1 (b), high intensity values are located in the parietal and posterior cingulate areas, which correspond to relevant ROIs. Regarding the GM images, high intensity values are observed in the temporal region in figure 4.2 (a) and the superior anterior cingulate region in figure 4.2 (b).

4.5 Comparison with state of the art methods

In table 4.7 the performance of our cross-modal TL methods which achieved better results using PET or GM images in the target domain are compared with state of the art methods that use Transfer Learning for AD vs NC classification, using data from the ADNI.

Table 4.7: Comparison between the proposed Transfer Learning methods and state of the art Transfer Learning methods for the classification task of AD vs NC which use images from the ADNI database.

Authors	Learning Algorithm	Biomarker(s)	Subjects (AD/NC)	ACC (%)	SENS (%)	SPEC(%)
[12]	SVM	MRI	186 / 226	94.7	94.1	94.8
[25]	SVM	MRI	54 / 63	82.91	79.63	85.71
[27]	2D CNN + LSTM	MRI	111 / 154	89.5	-	-
[20]	2D CNN + LSTM	MRI	132 / 132	90.62	-	-
[31]	3D CNN	MRI	70 / 70	99.3	98.6	97.2
[47]	3D CNN	MRI	755 / 755	95.39	-	-
[60]	3D CNN	MRI, FDG-PET	145 / 172	91.14	-	-
[9]	2D CNN	MRI/ DTI	188 / 228	92.5	94.7	90.4
TL PET-GM	3D CNN	MRI, FDG-PET	152/202	86.4	83.7	89.6
TL GM-PET	2D CNN + LSTM	MRI, FDG-PET	152/202	86.1	80.6	91.6

Although the developed cross-modal TL approach achieves lower accuracy than most state of the art methods, the results obtained are satisfactory and can still be comparable. From the listed methods, the best results are obtained using 3D CNN as the learning algorithm based on pre-trained 3D autoencoders [47, 31]. The cross-modal Transfer Learning approach developed by Aderghal et al. [9], uses ROI-based features of MRI and DTI, which have low feature dimensions and can be easily interpreted, contributing to better performance. The 2D CNN + LSTM model used in [20] divides the volumes into groups of slices and trains several 2D CNNs based on each group of slices. The final classification was achieved by the combination of models trained on different views (Sagittal, Axial and Coronal). The fact that several brain views were combined and several 2D CNNs were trained for each group of slices presents advantages in comparison to our CNN-LSTM model, which uses a Time Distributed 2D CNN to apply the same 2D convolution to every slice in the axial plane.

4.6 Summary

To summarize the results obtained for the several methods used throughout this thesis, a visual analysis of the dispersion of the accuracy and F1-score metrics obtained in the 5 folds for each method can be made using the box plots in figures 4.3 and 4.4.

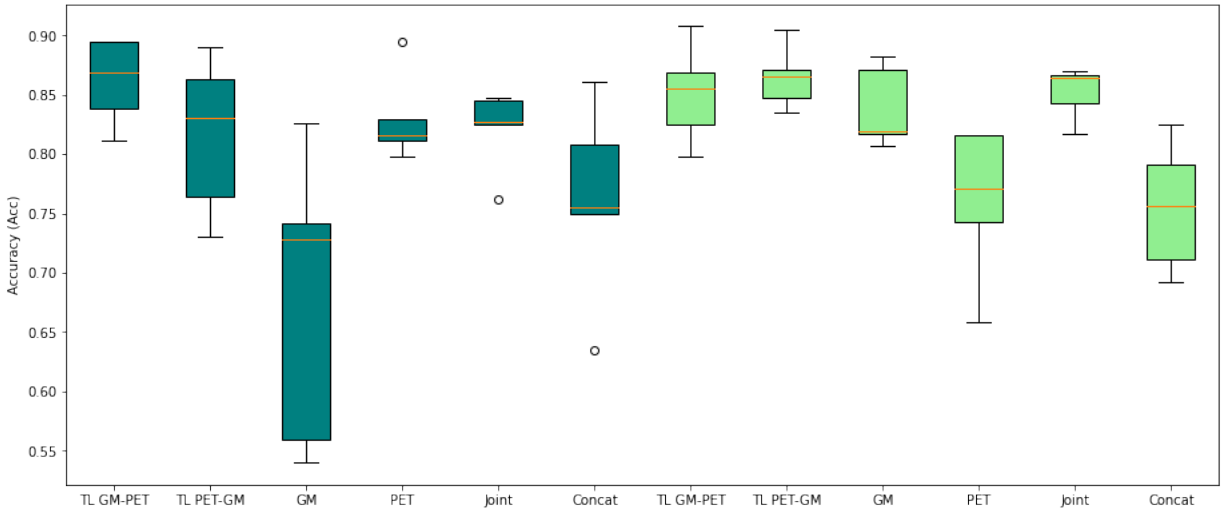


Figure 4.3: Box plots for the accuracy metric. The CNN-LSTM model is represented by the blue plots. The green plots represent the 3D CNN model.

Regarding the accuracy box plots shown in figure 4.3, higher median values can be observed for the Transfer Learning results, either using CNN-LSTM or 3D CNN, although training both modalities in the same deep learning network (denoted as ”joint” in the image), also show high median values. The interquartile range shows a higher dispersion of values specially for the GM modality in the CNN-LSTM model. Transfer Learning values generally show a small dispersion, when compared to the models trained from scratch, which shows a high consistency in the results. Transfer Learning accuracy shows a higher consistency for the 3D CNN model, but the highest median value is obtained for the CNN-LSTM model using TL GM-PET.

Outliers in the CNN-LSTM model using the PET modality, both modalities trained jointly in the same network or the concatenation model show the high variability of the results. The fact that Transfer Learning accuracy results don't show outliers highlights the consistency of these methods.

The same conclusions can be taken regarding the F1-score box plots shown in figure 4.4. A higher dispersion is found for the CNN-LSTM models in comparison to the 3D CNN models. Transfer Learning results have the highest means and present higher consistency, given the short interquartile range and absence of outliers. Since the F1-score metric places more importance in the false negatives and false positives, which are important aspects to be taken into account in classification problems related with medical images, it can be concluded that the use of cross-modal Transfer Learning can provide a more

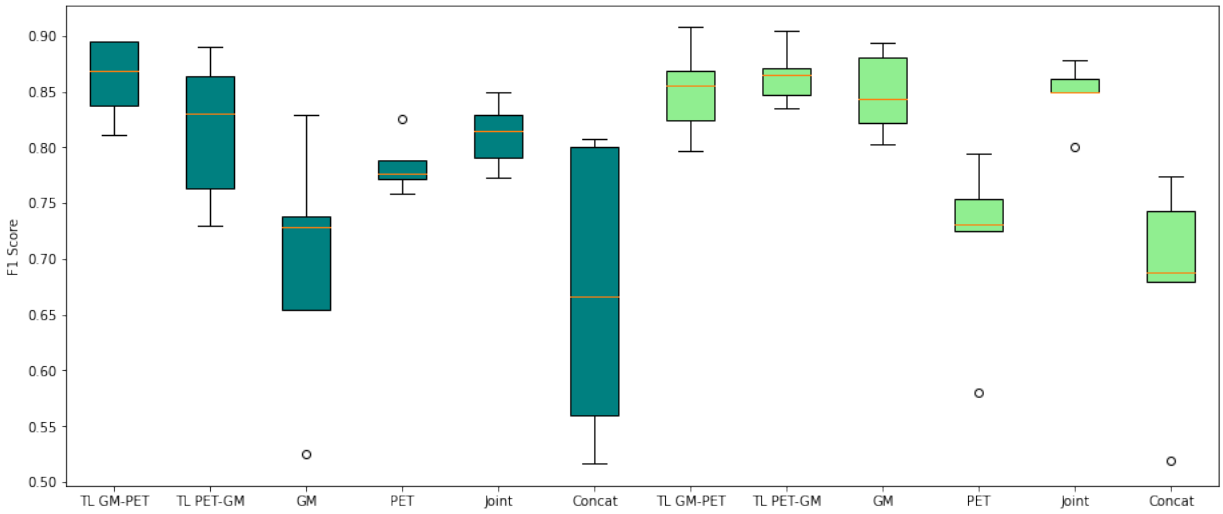


Figure 4.4: Box plots for the F1-score metric. The CNN-LSTM model is represented by the blue plots. The green plots represent the 3D CNN model.

reliable diagnosis of Alzheimer's disease.

Summarizing, both convolution based deep learning networks benefited from using cross-modal Transfer Learning, which provides an alternative way to share complementary information among neuroimaging modalities and deal with the overfitting phenomena, which is a recurrent problem when training deep learning networks with small datasets. These networks could correctly learn patterns associated to AD related changes in the brain and achieved an accuracy of 86.4% for the TL PET-GM method using a 3D CNN and 86.1% for the TL GM-PET method using the CNN-LSTM model for the classification of AD vs NC subjects. High sensitivity and specificity values were also obtained, achieving 83.7% sensitivity and 89.6% specificity for the best TL PET-GM model and 80.6% sensitivity and 91.6% specificity for the best TL GM-PET model, in the set of the 5 folds. Considering that there was still no cross-modal Transfer Learning study focusing on MRI and FDG-PET modalities, the results obtained using this approach are considered satisfactory and open interesting perspectives. Despite some limitations comparatively to the 3D CNN model, the CNN-LSTM network could also achieve satisfactory results, given the fact that this particular model still hasn't been used in AD detection, to the best of our knowledge, which can also open up new paths for deep learning systems applied to AD.

5 Conclusions

This thesis aimed at developing a Transfer Learning approach applied to a deep learning network for classification between AD vs NC subjects from the ADNI database, while exploring the advantages of using multimodal data. This was achieved by using cross-modal Transfer Learning, in which two deep learning models were trained on GM or PET data and later fine-tuned using the opposite modality in the target domain. In order to be able to draw conclusions on the Transfer Learning methods ability to combine different neuroimaging modalities, two other multi-task learning approaches were explored: Combining both modalities as inputs of the same deep learning network and concatenate two deep learning models at decision level. The results of these approaches were compared in terms of accuracy, sensitivity, specificity and F1-score. To verify the Transfer Learning models' ability to learn disease related patterns, heatmaps of intermediate activations were visualized and compared with relevant ROIs in AD diagnosis.

5.1 Achievements

Regarding the achievements of this work, it can be considered that the main objectives were accomplished, since the proposed Transfer Learning method achieves classification accuracies of 86.4% using a 3D CNN fine-tuned on GM data and 86.1% using a CNN-LSTM network fine-tuned on PET data, for the classification between AD vs NC subjects, outperforming the other studied approaches. Transfer Learning allows performance improvements regardless of the number of fine-tuned convolutional layers, as long as all the densely connected classifier layers were fine-tuned to the target task. The combination of two modalities of data through Transfer Learning allowed the deep learning networks to learn complementary information from both modalities and led to a better classification performance.

Considering the two deep learning architectures implemented, the 3D CNN architecture outperformed the CNN-LSTM network, achieving generally higher accuracy values and less dispersion among the results. Nevertheless, the CNN-LSTM model could still achieve an accuracy of 86.1% with Transfer Learning, which are satisfactory results considering the fact that this is the first approach that applies this particular CNN-LSTM model in AD classification.

While the best results obtained with the Transfer Learning approach could not outperform current state of the art methods, it is important to highlight that the deep learning networks and the Transfer Learning methods studied during this thesis, as well as the data used in the training phase, can be subject to substantive improvements, which can eventually lead to a better diagnosis of AD.

5.2 Future Work

Despite the recent trend in Transfer Learning, most of the studies still focus in using pre-trained networks from the ImageNet dataset. While these studies have shown great results, there are more possibilities that can be used for Transfer Learning, which can even combined with other methods, such as:

- In an approach similar to the concatenation method developed, the convolutional layers of each deep learning network could be fine-tuned when training the new classifier which concatenates the two networks;
- Performing Transfer Learning using a pre-trained network with natural images and fine-tuning on target domains with different neuroimaging modalities and concatenating the resulting networks;
- Performing cross-modal Transfer Learning, as developed in this thesis, but instead of using a network trained from scratch, a network previously trained with natural images could be used, or even trained in another dataset containing brain scans from different pathologies.

This cross-modal Transfer Learning approach could also be extended to the classification of the several AD stages, or to the prediction of MCI conversion to AD. To do this, it would be interesting to compare between other biomarkers, such as AV-45 PET, which can track earlier changes in the brain [34]. Training a model on the AD vs NC classes and fine-tuning the weights to classify also between MCI subjects can be an interesting way to apply Transfer Learning to the classification of several AD stages. Furthermore, better results could be achieved using ROI-based features, instead of voxel based features or slice-based features, since they have shown better results [18]. Specially in the case of the CNN-LSTM model, improvements could be made using BiLSTM, and also exploring slices in the sagittal and coronal views besides the axial view. These views could be combined in order to improve performance by capturing complementary information.

References

[1] 2020 alzheimer's Disease Facts and Figures. https://www.alz.org/alzheimers-dementia/facts-figures, Accessed: December 2020.

[2] Alzheimer's Disease Neuroimaging Initiative - ADNI. http://adni.loni.usc.edu/, Accessed: December 2020.

[3] CADDementia - a standardized evaluation framework for computer-aided diagnosis of dementia based on structural MRI data. https://caddementia.grand-challenge.org/About/, Accessed: December 2020.

[4] Colaboratory - frequently asked questions. https://research.google.com/colaboratory/faq.html/, Accessed: December 2020.

[5] Jupyter notebooks. https://jupyter.org/, Accessed: December 2020.

[6] NIH - Alzheimer's Disease fact sheet. https://www.nia.nih.gov/health/alzheimers-disease-fact-sheet, Accessed: December 2020.

[7] OASIS brains - Open Access Series of Imaging Studies. http://oasis-brains.org/, Accessed: December 2020.

[8] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

[9] Karim Aderghal, Alexander Khvostikov, Andrei Krylov, Jenny Benois-Pineau, Karim Afdel, and Gwenaelle Catheline. Classification of Alzheimer disease on imaging modalities with deep CNNs using cross-modal transfer learning. In *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*, pages 345–350. IEEE, 2018.

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

[11] Davide Castelvecchi. Can we open the black box of AI? *Nature News*, 538(7623):20, 2016.

[12] Bo Cheng, Mingxia Liu, Dinggang Shen, Zuoyong Li, Daoqiang Zhang, Alzheimer's Disease Neuroimaging Initiative, et al. Multi-domain transfer learning for early diagnosis of Alzheimer's Disease. *Neuroinformatics*, 15(2):115–132, 2017.

[13] Bo Cheng, Mingxia Liu, Daoqiang Zhang, Brent C Munsell, and Dinggang Shen. Domain transfer learning for MCI conversion prediction. *IEEE Transactions on Biomedical Engineering*, 62(7):1805–1817, 2015.

[14] Bo Cheng, Mingxia Liu, Daoqiang Zhang, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, et al. Robust multi-label transfer feature learning for early diagnosis of Alzheimer's Disease. *Brain imaging and behavior*, 13(1):138–153, 2019.

[15] Danni Cheng and Manhua Liu. Combining convolutional and recurrent neural networks for Alzheimer's disease diagnosis using PET images. In *2017 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–5. IEEE, 2017.

[16] Hongyoon Choi, Kyong Hwan Jin, Alzheimer's Disease Neuroimaging Initiative, et al. Predicting cognitive decline with deep learning of brain metabolism and amyloid imaging. *Behavioural brain research*, 344:103–109, 2018.

[17] François Chollet et al. "Keras". `https://keras.io/`, 2015.

[18] François Chollet. *Deep Learning with Python*. Manning Publications Co, 2018.

[19] Alexander Drzezga, Nicola Lautenschlager, Hartwig Siebner, Matthias Riemenschneider, Frode Willoch, Satoshi Minoshima, Markus Schwaiger, and Alexander Kurz. Cerebral metabolic changes accompanying conversion of mild cognitive impairment into Alzheimer's Disease: a PET follow-up study. *European journal of nuclear medicine and molecular imaging*, 30(8):1104–1113, 2003.

[20] Amir Ebrahimi-Ghahnavieh, Suhuai Luo, and Raymond Chiong. Transfer Learning for Alzheimer's Disease Detection on MRI Images. In *2019 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, pages 133–138. IEEE, 2019.

[21] 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*, 187:105242, 2020.

[22] Shaker El-Sappagh, Tamer Abuhmed, SM Riazul Islam, and Kyung Sup Kwak. Multimodal multitask deep learning model for Alzheimer's disease progression detection based on time series data. *Neurocomputing*, 412:197–215, 2020.

[23] Lei Fan, Zhaoqiang Xia, Xiaobiao Zhang, and Xiaoyi Feng. Lung nodule detection based on 3D convolutional neural networks. In *2017 International Conference on the Frontiers and Advances in Data Science (FADS)*, pages 7–10. IEEE, 2017.

[24] Chiyu Feng, Ahmed Elazab, Peng Yang, Tianfu Wang, Baiying Lei, and Xiaohua Xiao. 3D convolutional neural network and stacked bidirectional recurrent neural network for Alzheimer's disease diagnosis. In *International Workshop on PRedictive Intelligence In MEdicine*, pages 138–146. Springer, 2018.

[25] Roman Filipovych, Christos Davatzikos, Alzheimer's Disease Neuroimaging Initiative, et al. Semi-supervised pattern classification of medical images: application to mild cognitive impairment (MCI). *NeuroImage*, 55(3):1109–1119, 2011.

[26] Giovanni B Frisoni, Nick C Fox, Clifford R Jack Jr, Philip Scheltens, and Paul M Thompson. The clinical use of structural MRI in Alzheimer Disease. *Nature Reviews Neurology*, 6(2):67, 2010.

[27] Linlin Gao, Haiwei Pan, Fujun Liu, Xiaoqin Xie, Zhiqiang Zhang, Jinming Han, Alzheimer's Disease Neuroimaging Initiative, et al. Brain disease diagnosis using deep learning features from longitudinal MR images. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, pages 327–339. Springer, 2018.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[30] Marcia Hon and Naimul Mefraz Khan. Towards Alzheimer's Disease classification through transfer learning. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1166–1169. IEEE, 2017.

[31] Ehsan Hosseini-Asl, Georgy Gimel'farb, and Ayman El-Baz. Alzheimer's Disease diagnostics by a deeply supervised adaptable 3D convolutional network. *arXiv preprint arXiv:1607.00556*, 2016.

[32] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and¡ 0.5 MB model size. *arXiv preprint arXiv:1602.07360*, 2016.

[33] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[34] Clifford R Jack Jr, David S Knopman, William J Jagust, Ronald C Petersen, Michael W Weiner, Paul S Aisen, Leslie M Shaw, Prashanthi Vemuri, Heather J Wiste, Stephen D Weigand, et al. Tracking pathophysiological processes in Alzheimer's disease: an updated hypothetical model of dynamic biomarkers. *The Lancet Neurology*, 12(2):207–216, 2013.

[35] Keith A Johnson, Nick C Fox, Reisa A Sperling, and William E Klunk. Brain imaging in Alzheimer disease. *Cold Spring Harbor perspectives in medicine*, 2(4):a006213, 2012.

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

[37] Stefan Klöppel, Cynthia M Stonnington, Josephine Barnes, Frederick Chen, Carlton Chu, Catriona D Good, Irina Mader, L Anne Mitchell, Ameet C Patel, Catherine C Roberts, et al. Accuracy of dementia diagnosis—a direct comparison between radiologists and a computerized method. *Brain*, 131(11):2969–2974, 2008.

[38] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[39] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[40] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[41] Garam Lee, Kwangsik Nho, Byungkon Kang, Kyung-Ah Sohn, and Dokyoon Kim. Predicting Alzheimer's disease progression using multi-modal deep learning approach. *Scientific reports*, 9(1):1–12, 2019.

[42] Kuan Liu, Yanen Li, Ning Xu, and Prem Natarajan. Learn to combine modalities in multimodal deep learning. *arXiv preprint arXiv:1805.11730*, 2018.

[43] Donghuan Lu, Karteek Popuri, Gavin Weiguang Ding, Rakesh Balachandar, and Mirza Faisal Beg. Multimodal and multiscale deep neural networks for the early diagnosis of Alzheimer's disease using structural MR and FDG-PET images. *Scientific reports*, 8(1):1–13, 2018.

[44] Siyuan Lu, Zhihai Lu, and Yu-Dong Zhang. Pathological brain detection based on AlexNet and transfer learning. *Journal of computational science*, 30:41–47, 2019.

[45] Muazzam Maqsood, Faria Nazir, Umair Khan, Farhan Aadil, Habibullah Jamal, Irfan Mehmood, and Oh-young Song. Transfer learning assisted classification and detection of Alzheimer's disease stages using 3D MRI scans. *Sensors*, 19(11):2645, 2019.

[46] Pedro M Morgado, Margarida Silveira, Alzheimer s Disease Neuroimaging Initiative, et al. Minimal neighborhood redundancy maximal relevance: Application to the diagnosis of Alzheimer s disease. *Neurocomputing*, 155:295–308, 2015.

[47] Adrien Payan and Giovanni Montana. Predicting Alzheimer's Disease: a neuroimaging study with 3D convolutional neural networks. *arXiv preprint arXiv:1502.02506*, 2015.

[48] Arjun Punjabi, Adam Martersteck, Yanran Wang, Todd B Parrish, Aggelos K Katsaggelos, and Alzheimer's Disease Neuroimaging Initiative. Neuroimaging modality fusion in Alzheimer's classification using convolutional neural networks. *Plos one*, 14(12):e0225759, 2019.

[49] Farheen Ramzan, Muhammad Usman Ghani Khan, Asim Rehmat, Sajid Iqbal, Tanzila Saba, Amjad Rehman, and Zahid Mehmood. A deep learning approach for automated diagnosis and multi-class classification of Alzheimer's disease stages using resting-state fMRI and residual neural networks. *Journal of Medical Systems*, 44(2):37, 2020.

[50] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. In *Classification in BioApps*, pages 323–350. Springer, 2018.

[51] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[52] Saman Sarraf, Ghassem Tofighi, et al. DeepAD: Alzheimer s Disease classification via deep convolutional neural networks using MRI and fMRI. *BioRxiv*, page 070441, 2016.

[53] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[54] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016.

[55] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[56] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.

[57] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.

[58] Kim-Han Thung and Chong-Yaw Wee. A brief review on multi-task learning. *Multimedia Tools and Applications*, 77(22):29705–29725, 2018.

[59] A Vedaldi, Y Jia, E Shelhamer, J Donahue, S Karayev, J Long, and T Darrell. Convolutional architecture for fast feature embedding. *Cornell University, arXiv: 1408.5093 v12014*, 2014.

[60] Tien Duong Vu, Hyung-Jeong Yang, Van Quan Nguyen, A-Ran Oh, and Mi-Sun Kim. Multimodal learning using convolution neural network and Sparse Autoencoder. In *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 309–312. IEEE, 2017.

[61] Congling Wu, Shengwen Guo, Yanjia Hong, Benheng Xiao, Yupeng Wu, Qin Zhang, Alzheimer's Disease Neuroimaging Initiative, et al. Discrimination and conversion prediction of mild cognitive impairment using convolutional neural networks. *Quantitative Imaging in Medicine and Surgery*, 8(10):992, 2018.

[62] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[63] Daoqiang Zhang, Yaping Wang, Luping Zhou, Hong Yuan, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, et al. Multimodal classification of Alzheimer's disease and mild cognitive impairment. *Neuroimage*, 55(3):856–867, 2011.

Appendices

A - Results from the Transfer Learning experiments

In this appendix, the results obtained from the experiments performed using our cross-modal Transfer Learning approach, regarding the number of fine-tuned layers and number of replaced top layers are presented.

Table A.1: Transfer Learning results for the CNN-LSTM model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), without replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.555	0.103	0.511	0.407	0.498	0.402	0.488	0.402	0.381	0.286
1	0.629	0.107	0.451	0.123	0.765	0.102	0.592	0.102	0.507	0.145
2	0.814	0.044	0.740	0.033	0.875	0.093	0.813	0.093	0.769	0.048
4	0.766	0.068	0.724	0.058	0.815	0.139	0.751	0.139	0.722	0.075
6	0.824	0.042	0.760	0.063	0.885	0.087	0.830	0.087	0.786	0.033
8	0.827	0.047	0.789	0.047	0.863	0.076	0.813	0.076	0.796	0.038
10	0.824	0.045	0.767	0.073	0.875	0.039	0.818	0.039	0.787	0.041

Table A.2: Transfer Learning results for the CNN-LSTM model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.627	0.148	0.599	0.170	0.688	0.263	0.632	0.263	0.578	0.110
1	0.789	0.056	0.704	0.058	0.862	0.099	0.789	0.099	0.738	0.064
3	0.808	0.064	0.715	0.061	0.885	0.110	0.832	0.110	0.763	0.048
5	0.776	0.062	0.661	0.147	0.889	0.113	0.818	0.113	0.708	0.086
7	0.805	0.076	0.717	0.147	0.894	0.087	0.825	0.087	0.752	0.102
9	0.861	0.033	0.806	0.087	0.916	0.046	0.872	0.046	0.831	0.025

Table A.3: Transfer Learning results for the CNN-LSTM model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), replacing the top 4 layers.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.790	0.035	0.754	0.070	0.838	0.113	0.776	0.113	0.750	0.044
2	0.824	0.034	0.689	0.067	0.933	0.023	0.877	0.023	0.768	0.032
4	0.837	0.034	0.717	0.031	0.932	0.050	0.886	0.050	0.790	0.017
6	0.830	0.056	0.766	0.042	0.889	0.090	0.832	0.090	0.790	0.078

Table A.4: Transfer Learning results for the CNN-LSTM model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), without replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.461	0.025	0.022	0.016	1.0	0.0	0.8	0.0	0.043	0.030
1	0.724	0.059	0.542	0.105	0.877	0.068	0.772	0.068	0.632	0.099
2	0.667	0.078	0.531	0.086	0.780	0.080	0.661	0.080	0.587	0.111
4	0.698	0.047	0.510	0.083	0.841	0.066	0.727	0.066	0.598	0.090
6	0.801	0.050	0.741	0.099	0.872	0.023	0.875	0.023	0.800	0.065
8	0.802	0.038	0.719	0.079	0.900	0.020	0.898	0.020	0.797	0.051
10	0.808	0.042	0.740	0.096	0.885	0.029	0.889	0.029	0.805	0.059

Table A.5: Transfer Learning results for the CNN-LSTM model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.570	0.035	0.594	0.191	0.534	0.264	0.630	0.264	0.591	0.068
1	0.736	0.113	0.810	0.082	0.628	0.320	0.774	0.320	0.779	0.064
3	0.808	0.042	0.796	0.064	0.822	0.066	0.846	0.066	0.818	0.049
5	0.810	0.041	0.802	0.060	0.816	0.041	0.842	0.041	0.821	0.047
7	0.816	0.060	0.752	0.086	0.888	0.056	0.893	0.056	0.815	0.065
9	0.758	0.147	0.782	0.052	0.743	0.347	0.836	0.347	0.792	0.087

Table A.6: Transfer Learning results for the CNN-LSTM model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), replacing the top 4 layers.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.736	0.096	0.690	0.132	0.786	0.083	0.797	0.083	0.736	0.103
2	0.767	0.076	0.690	0.166	0.854	0.046	0.855	0.046	0.753	0.103
4	0.772	0.055	0.695	0.111	0.861	0.036	0.858	0.036	0.765	0.077
6	0.769	0.034	0.709	0.107	0.837	0.089	0.852	0.090	0.767	0.053

Table A.7: Transfer Learning results for the 3D CNN model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), without replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.602	0.134	0.652	0.357	0.498	0.405	0.480	0.405	0.517	0.265
3	0.779	0.031	0.683	0.101	0.872	0.105	0.808	0.105	0.721	0.021
6	0.832	0.052	0.812	0.044	0.844	0.069	0.789	0.069	0.799	0.074
9	0.851	0.038	0.778	0.072	0.909	0.068	0.865	0.068	0.814	0.041
13	0.824	0.042	0.790	0.066	0.855	0.047	0.794	0.047	0.789	0.058
17	0.829	0.030	0.792	0.043	0.862	0.054	0.808	0.054	0.796	0.031

Table A.8: Transfer Learning results for the 3D CNN model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.758	0.054	0.717	0.039	0.792	0.076	0.717	0.076	0.712	0.081
2	0.797	0.056	0.650	0.127	0.925	0.068	0.872	0.068	0.730	0.054
5	0.803	0.051	0.825	0.036	0.794	0.073	0.741	0.073	0.776	0.074
8	0.827	0.033	0.793	0.065	0.858	0.057	0.803	0.057	0.794	0.029
12	0.819	0.024	0.791	0.041	0.845	0.044	0.786	0.044	0.785	0.033
16	0.833	0.009	0.792	0.057	0.865	0.046	0.804	0.046	0.795	0.045

Table A.9: Transfer Learning results for the 3D CNN model pre-trained with GM data and fine-tuned on the PET modality (TL GM-PET), replacing the top 6 layers.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.835	0.050	0.792	0.058	0.864	0.082	0.820	0.082	0.801	0.058
3	0.832	0.040	0.801	0.046	0.843	0.073	0.803	0.073	0.801	0.038
7	0.798	0.032	0.743	0.059	0.841	0.037	0.768	0.037	0.752	0.059
11	0.830	0.027	0.778	0.041	0.860	0.038	0.810	0.038	0.794	0.027

Table A.10: Transfer Learning results for the 3D CNN model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), without replacing the last dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.587	0.114	0.675	0.358	0.483	0.361	0.511	0.361	0.569	0.289
3	0.804	0.036	0.775	0.060	0.839	0.052	0.854	0.052	0.811	0.045
6	0.863	0.024	0.813	0.076	0.919	0.042	0.931	0.042	0.865	0.033
9	0.838	0.034	0.825	0.061	0.850	0.078	0.877	0.078	0.847	0.037
13	0.860	0.025	0.813	0.068	0.912	0.035	0.922	0.035	0.862	0.035
17	0.853	0.026	0.826	0.035	0.886	0.035	0.898	0.035	0.860	0.031

Table A.11: Transfer Learning results for the 3D CNN model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), replacing the top dense layer.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.722	0.041	0.693	0.078	0.758	0.056	0.779	0.056	0.730	0.048
2	0.792	0.066	0.799	0.063	0.785	0.075	0.816	0.075	0.807	0.069
5	0.834	0.055	0.789	0.061	0.895	0.118	0.907	0.118	0.839	0.052
8	0.857	0.015	0.824	0.033	0.895	0.052	0.910	0.052	0.864	0.017
12	0.855	0.026	0.796	0.069	0.922	0.069	0.935	0.068	0.856	0.033
16	0.841	0.041	0.818	0.043	0.870	0.063	0.883	0.063	0.848	0.044

Table A.12: Transfer Learning results for the 3D CNN model pre-trained with PET data and fine-tuned on the GM modality (TL PET-GM), replacing the last 6 layers.

Fine-tuned layers	ACC		SENS		SPEC		PREC		F1 Score	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
0	0.846	0.030	0.808	0.058	0.889	0.032	0.899	0.032	0.850	0.040
3	0.864	0.023	0.837	0.054	0.896	0.028	0.908	0.028	0.870	0.031
7	0.864	0.028	0.822	0.062	0.912	0.036	0.922	0.036	0.868	0.036
11	0.857	0.018	0.838	0.052	0.878	0.033	0.896	0.033	0.864	0.023

B - Results from the model concatenation experiments

In this appendix, the results obtained from the experiments performed using the concatenation approach are detailed. These results are relative to the experiments were made regarding the number of layers dense layers and units in each layer used after concatenating the features, in order to perform the final decision. Experiments regarding the concatenation of the outputs of the last dense layer of each network (denoted as ”with last” in the tables) were compared with concatenation of the outputs of the penultimate layer of each network (denoted as ”last cut”).

Table B.1: Results for the concatenation of two CNN-LSTM networks.

# dense layers	ACC		SENS		SPEC		PREC		F1 Score	
and units	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
128-32-2 (last cut)	0.737	0.073	0.535	0.092	0.906	0.103	0.836	0.103	0.645	0.100
64-32-2 (last cut)	0.736	0.058	0.548	0.093	0.892	0.083	0.798	0.083	0.645	0.099
32-32-2 (last cut)	0.758	0.073	0.56	0.101	9156	0.097	0.846	0.097	0.670	0.115
128-64-2 (last cut)	0.749	0.063	0.561	0.101	0.901	0.082	0.810	0.082	0.660	0.111
64-2 (last cut)	0.747	0.073	0.542	0.092	0.917	0.091	0.839	0.091	0.654	0.109
128-2 (last cut)	0.736	0.066	0.548	0.093	0.893	0.086	0.796	0.086	0.646	0.109
32-2 (last cut)	0.758	0.073	0.561	0.101	0.916	0.097	0.846	0.097	0.670	0.115
32-2 (with last)	0.762	0.075	0.556	0.105	0.925	0.093	0.856	0.093	0.670	0.120
64-2 (with last)	0.757	0.076	0.556	0.105	0.917	0.091	0.836	0.091	0.666	0.124
128-2 (with last)	0.753	0.078	0.556	0.105	0.913	0.092	0.821	0.092	0.662	0.128
2 (with last)	0.554	0.096	0.270	0.274	0.745	0.184	0.333	0.184	0.288	0.270
2 (last cut)	0.757	0.071	0.567	0.107	0.909	0.085	0.824	0.085	0.669	0.120
32-32-2 (with last)	0.748	0.072	0.556	0.105	0.902	0.085	0.806	0.085	0.657	0.123

Table B.2: Results for the concatenation of two 3D CNN networks.

# dense layers	ACC		SENS		SPEC		PREC		F1 Score	
and units	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
128-32-2 (last cut)	0.736	0.055	0.561	0.083	0.878	0.06	0.781	0.062	0.651	0.097
64-32-2 (last cut)	0.736	0.053	0.576	0.094	0.870	0.056	0.774	0.056	0.657	0.092
32-32-2 (last cut)	0.755	0.049	0.599	0.109	0.885	0.447	0.803	0.045	0.681	0.088
128-64-2 (last cut)	0.710	0.101	0.554	0.083	0.844	0.133	0.756	0.133	0.634	0.126
64-2 (last cut)	0.740	0.062	0.594	0.105	0.861	0.068	0.768	0.068	0.666	0.106
128-2 (last cut)	0.718	0.090	0.583	0.093	0.835	0.106	0.740	0.106	0.650	0.125
32-2 (last cut)	0.736	0.048	0.565	0.086	0.877	0.056	0.784	0.056	0.653	0.083
32-2 (with last)	0.695	0.138	0.586	0.100	0.799	0.185	0.724	0.185	0.639	0.146
64-2 (with last)	0.698	0.085	0.550	0.098	0.817	0.108	0.715	0.108	0.617	0.123
128-2 (with last)	0.676	0.108	0.577	0.118	0.763	0.143	0.675	0.143	0.614	0.135
2 (with last)	0.478	0.187	0.631	0.158	0.389	0.315	0.497	0.031	0.528	0.147
2 (last cut)	0.753	0.047	0.594	0.105	0.886	0.063	0.880	0.063	0.676	0.093
32-32-2 (with last)	0.703	0.119	0.605	0.150	0.789	0.171	0.735	0.171	0.644	0.144
64-32-2 (with last)	0.693	0.070	0.577	0.117	0.792	0.098	0.689	0.098	0.621	0.118
128-32-2 (with last)	0.694	0.097	0.588	0.128	0.787	0.125	0.693	0.125	0.628	0.136
64-64-2 (with last)	0.693	0.081	0.579	0.097	0.793	0.106	0.692	0.106	0.626	0.118
64-64-2 (last cut)	0.741	0.059	0.565	0.086	0.887	0.051	0.792	0.051	0.658	0.097