UNIVERSIDAD DE INVESTIGACIÓN DE TECNOLOGÍA EXPERIMENTAL YACHAY

Escuela de Ciencias Matemáticas y Computacionales

TÍTULO: Deep convolutional generative adversarial networks (DCGANs) applied to synthetic image augmentation to improve object classification for Alzheimer disease

Trabajo de integración curricular presentado como requisito para la obtención del título de Ingeriera en Tecnologías de la Información

Autora:

Moncada Da Silva Claudia Maria

Tutor:

Ph.D. - Fonseca Delgado Rigoberto Salomón

Urcuquí, abril de 2024

Autoría

Yo, **CLAUDIA MARIA MONCADA DA SILVA**, con número de pasaporte 163192716, declaro que las ideas, juicios, valoraciones, interpretaciones, consultas bibliográficas, definiciones y conceptualizaciones expuestas en el presente trabajo; así cómo, los procedimientos y herramientas utilizadas en la investigación, son de absoluta responsabilidad de el/la autor/a del trabajo de integración curricular. Así mismo, me acojo a los reglamentos internos de la Universidad de Investigación de Tecnología Experimental Yachay.

Urcuquí, abril de 2024.

Claudia Maria Da Silva

Claudia Maria Moncada Da Silva

Pasaporte: 163192716

Autorización de publicación

Yo, **CLAUDIA MARIA MONCADA DA SILVA**, con número de pasaporte 163192716, cedo a la Universidad de Investigación de Tecnología Experimental Yachay, los derechos de publicación de la presente obra, sin que deba haber un reconocimiento económico por este concepto. Declaro además que el texto del presente trabajo de titulación no podrá ser cedido a ninguna empresa editorial para su publicación u otros fines, sin contar previamente con la autorización escrita de la Universidad.

Asimismo, autorizo a la Universidad que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación

Urcuquí, abril de 2024.

Claudia Maria DaSilva

Claudia Maria Moncada Da Silva

Pasaporte: 163192716

Dedication

I dedicate this work first of all to God, for giving me life and allowing me to get to this important moment in my professional training.

To my parents, although they are not physically present, their lessons continue to guide me day by day. Thanks to their unconditional love and support, I feel that they are with me at all times and that this moment would have been as special for them as it is for me. Their example of perseverance and dedication has been my greatest motivation. I know they are proud of me from wherever they are.

Finally, I dedicate this achievement to my family, whose support and understanding have been a fundamental pillar in my life. Although we have been far apart during this journey, their unfailing love has given me the courage to move forward.

This achievement is also yours.

Claudia Maria Moncada Da Silva

Acknowledgment

I cannot thank my family enough for their unconditional support. Their support and encouragement has been my greatest strength.

To my friends, who are now dispersed around the world, but who despite the distance have known how to be there for me to support me, guide me, encourage me, motivate me and make me feel that I have a lot of people who love and appreciate me no matter what.

I would like to thank all the people who welcomed me in Ecuador with open arms and supported me in different forms along this way. They have become my second family and I will not have the time or the resources to thank them for everything they have done for me over these years. They will always have an important place in my life.

To my tutor Rigoberto Fonseca for his trust on me to carry out this project, having guided me in this process, and having been the person who, with his guidelines and knowledge, was able to explain to me the details to complete my thesis.

To the members of the School of Mathematical and Computer Sciences for their constant support throughout my university studies. Their knowledge and support have enriched my academic training and have contributed to the completion of this project.

Finally, I would like to dedicate special thanks to all those who, in one way or another, have contributed to this project and have left their mark on my path towards the completion of this research.

Claudia Maria Moncada Da Silva

Resumen

El Alzheimer es una enfermedad neurodegenerativa progresiva que afecta al cerebro causando demencia. Se caracteriza por la pérdida gradual de las funciones cognitivas, la memoria y las habilidades de pensamiento. Debido al impacto de esta enfermedad en nuestra sociedad, este trabajo se centra en la aplicación de redes generativas adversariales (GANs), concretamente utilizando una Red Generativa Adversarial Convolucional Profunda (DCGAN), para la generación de imágenes sintéticas de la enfermedad de Alzheimer basadas en Imágenes de Resonancia Magnética (MRI) reales. La red se entrena sobre un conjunto de datos de imágenes de Alzheimer para aprender las características distintivas de la enfermedad. Por lo tanto, el objetivo principal es mejorar la precisión en tareas de clasificación posteriores entrenando modelos de clasificación con conjuntos de datos aumentados con imágenes sintéticas. El proceso comienza con la implementación de un DCGAN y se utiliza para generar nuevas imágenes sintéticas. Posteriormente, se evalúa el impacto de estas imágenes sintéticas en la precisión de los algoritmos de clasificación. Se utilizan varios modelos, como InceptionV3, AlexNet, VGG-16 y ResNet-18, para realizar las tareas de clasificación. Los resultados muestran que InceptionV3 (94,25% frente a 95,40%) y AlexNet (91,95% frente a 95,40%) obtienen los mejores resultados de precisión en el conjunto de pruebas. Además, se utiliza el algoritmo t-SNE para verificar que las imágenes sintéticas han adoptado características de las originales. Este estudio demuestra la utilidad de la generación de imágenes sintéticas con DCGAN para mejorar la capacidad de los algoritmos de clasificación para identificar las diferentes etapas del Alzheimer.

Palabras Clave:

Redes generativas adversarias, t-SNE, Aumento de datos, Enfermedad de Alzheimer, Neuroimagen, Clasificación de imágenes.

Abstract

Alzheimer is a progressive neurodegenerative disease that affects the brain causing dementia. It is characterized by the gradual loss of cognitive functions, memory and thinking skills. Due to the impact of this disease in our society, this work is focused on the application of generative adversarial networks (GANs), specifically using a Deep Convolutional Generative Adversarial Network (DCGAN), for the generation of synthetic images of Alzheimer disease based on real Magnetic Resonance Image (MRI). The network is trained on a dataset of Alzheimer images to learn the distinctive features of the disease. Therefore, the main goal is to improve accuracy in subsequent classification tasks by training classification models with datasets augmented with synthetic images. The process starts with the implementation of a DCGAN and it is used to generate new synthetic images. Subsequently, the impact of these synthetic images on the accuracy of the classification algorithms is evaluated. Several models, such as InceptionV3, AlexNet, VGG-16 and ResNet-18, are used to perform the classification tasks. The results show that InceptionV3 (94.25% vs 95.40%) and AlexNet (91.95% vs 95.40%) achieve the best accuracy performance on the test set. Additionally, the algorithm t-SNE is used to verify that the synthetic images have adopted features from the originals. This study demonstrates the utility of synthetic image generation with DCGAN to improve the ability of classification algorithms to identify different Alzheimer stages.

Keywords:
Generative adversarial networks, t-SNE, Data augmentation, Alzheimer disease, Neuroimaging, Image classification.

Contents

Dedication **v**

Acknowledgment **vii**

Resumen **ix**

Abstract **xi**

Contents **xiii**

List of Tables **xvii**

List of Figures **xix**

1 Introduction **1**

1.1 Problem description 1

1.2 Justification 2

1.3 Objectives 3

1.3.1 General Objective 3

1.3.2 Specific Objectives 3

1.4 Contribution 3

2 Theoretical Framework **5**

2.1 Generative Adversarial Networks (GANs) 5

2.1.1 Classification of GANs Models 7

2.1.2 GAN Models 8

2.1.3 Hyperparameters 10

2.2 Traditional data augmentation 12

2.3 T-Distributed Stochastic Neighbor Embedding (t-SNE) 13
2.4 Convolutional neural networks 15
2.5 Convolutional neural networks for image classification 15
    2.5.1 InceptionV3 15
    2.5.2 ResNet-18 16
    2.5.3 AlexNet 17
    2.5.4 Very Deep Convolutional Network (VGG) 18
2.6 Adagrad optimizer 19
2.7 Alzheimer Disease and Dementia 20
2.8 Stages of Alzheimer's Disease 21
    2.8.1 Cognitive Normal (CN) 21
    2.8.2 Mild Cognitive Impairment (MCI) 22
    2.8.3 Alzheimer Disease (AD) 22
2.9 Magnetic Resonance Imaging (MRI) 22
    2.9.1 Relaxation Times 23
    2.9.2 Acquisition plane 24

3 State of the Art 27
3.1 Machine learning techniques for ADNI dataset classification 27
3.2 Deep Learning techniques for ADNI dataset classification 29
3.3 GAN-based techniques for data augmentation of ADNI dataset 31

4 Methodology 35
4.1 ADNI dataset 35
4.2 Dataset preparation 36
    4.2.1 Dataset splitting 39
4.3 System design 39
    4.3.1 Data distribution with t-SNE 39
    4.3.2 Synthetic image generation by DCGAN 41
    4.3.3 Classification models 43
4.4 Evaluation metrics 43
    4.4.1 Metrics for classification models 43

5 Results and Discussion 47

5.1 Phase 1: Generation of synthetic images with DCGAN and evaluation of their quality and variability 47

5.1.1 ADNI dataset data distribution with t-SNE 47

5.1.2 DCGAN training step 48

5.1.3 Distribution of the synthetic images from DCGAN with t-SNE . . . 49

5.2 Phase 2: Improvement of the classification algorithm with the augmentation of DCGAN synthetic images of the Alzheimer Disease class 52

5.3 Hardware 60

6 Conclusions 61

6.0.1 Future Work 63

Bibliography 65

Appendices 73

.1 Appendix 1 - Advanced Search for ADNI dataset. 75

.2 Appendix 2 - List of Papers 76

List of Tables

2.1 Main features of the convolutional networks used in this work. 19

4.1 Search criteria and selected features for the acquisition of images from the ADNI dataset 37
4.2 Proportion of images before and after processing part. 38
4.3 Hyperparameters used for the implementation of the t-SNE algorithm with the sklearn library. 41
4.4 Hyperparameters used for the design and training of the DCGAN for ADNI dataset. 42
4.5 Hyperparameters used for the different classification models: VGG-16, ResNet-18, AlexNet and InceptionV3. 44

5.1 Comparison of the results of the different classification models with both the original images and the augmented synthetic images for the AD class. . 54
5.2 Additional metrics for the InceptionV3 model with original images from the ADNI dataset. 58
5.3 Additional metrics for the InceptionV3 model with the augmented synthetic images for the AD class. 58
5.4 Additional metrics for the AlexNet model with original images from the ADNI dataset. 60
5.5 Additional metrics for the AlexNet model with the augmented synthetic images for the AD class. 60

List of Figures

2.1 Typical Generative Adversarial Network (GAN) architecture. 7
2.2 DCGAN Generator Architecture. A 100-dimensional uniform distribution (Z) is projected to a small spatial convolutional representation with some feature maps. This is followed by four fractionally-strided convolutions. Finally, this converts to a representation of 64x64 pixel image. 10
2.3 DCGAN Discriminator Architecture. It uses multiple convolutional layers to process the input image progressively. The last layer uses a Sigmoid function to produce values between 0 (fake) and 1 (real). 11
2.4 Example of how the t-SNE algorithm works. It starts with a high dimensional representation of points to be translated into a low dimensional space. 15
2.5 InceptionV3 architecture which consists of 48 layers formed by symmetrical and asymmetrical compilation blocks. These include convolutions, reductions and fully connected layers [1]. 16
2.6 ResNet-18 architecture which consists of 18 layers in total, including convolution layers, pooling layers and fully connected layers [2]. 17
2.7 AlexNet architecture composed by 8 layers with weights, 5 convolutional layers, and 3 fully connected layers [3]. 18
2.8 VGG-16 Architecture. It is composed of 13 convolutional layers, 5 max-pooling layers, and 3 fully connected layers[4]. 19
2.9 Axial representation of an MRI with Longitudinal relaxation time (T1) [5] 24
2.10 Axial representation of an MRI with Transversal relaxation time (T2) [5]. . 25
2.11 Representation of the three acquisition planes in order from left to right: sagittal, axial and coronal [6]. 25

4.1 Examples of ADNI dataset. First row: Cognitive normal images. Second row: Mild cognitive impairment images. Third row: Alzheimer disease images. 36

4.2 Examples of blurred images from ADNI dataset for the Cognitive Normal class. 38

4.3 Examples of blurred images from ADNI dataset for the Mild Cognitive Impairment class. 38

4.4 Examples of blurred images from ADNI dataset for the Alzheimer Disease class. 38

4.5 Proportion of the dataset for each class before and after the cleaning phase. 39

4.6 Representation of both phases of implementation of the t-SNE algorithm. The first step is to evaluate the original image dataset. The second step is to evaluate the increase of synthetic images for the AD class. 40

4.7 Representation of each of the steps followed for the DCGAN training. This includes the selection and tuning of hyperparameters until the generation of the final model. 41

4.8 Representation of the stages of the classification process for each phase of the work. The first step is to evaluate the performance of the original data. The second step is to evaluate the performance of the synthesized data generated by DCGAN and compare with the previous one. 43

4.9 Representation of a confusion matrix. It includes the results of the classification, considering True Positive (TP),True Negative (TN), False Positive (FP) and False Negative (FN) respectively [7]. 45

5.1 Result of the t-SNE algorithm for the original ADNI dataset images. . . . 48

5.2 DCGAN loss evolution during training for the generator and discriminator models. Blue line (G) corresponds to the loss of the generator. Orange line (D) corresponds to the loss of the discriminator. 49

5.3 Result of t-SNE for the first batch of synthetic images of the DCGAN pretrained model in the AD class of the ADNI dataset. 50

5.4 Result of t-SNE for the second batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset. 50

5.5 Result of t-SNE for the third batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset. 50
5.6 Result of t-SNE for the fourth batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset. 50
5.7 Result of t-SNE for the fifth batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset. 50
5.8 Front view of the t-SNE plot for synthetic images of the AD class of the ADNI dataset. 51
5.9 Top view of the t-SNE plot for synthetic images of the AD class of the ADNI dataset. 51
5.10 Front view of the t-SNE plot for original images for the four classes of the Alzheimer's Dataset from Kaggle. 52
5.11 Top view of the t-SNE plot for original images for the four classes of the Alzheimer's Dataset from Kaggle. 52
5.12 Lot of original images and synthetic images generated by DCGAN. Left side: original images from the ADNI dataset. Right side: synthetic images generated by DCGAN. 53
5.13 InceptionV3 accuracy on train dataset. 54
5.14 InceptionV3 loss on train dataset. 54
5.15 ResNet-18 accuracy on train dataset. 54
5.16 ResNet-18 loss on train dataset. 54
5.17 VGG-16 accuracy on train dataset. 55
5.18 VGG-16 loss on train dataset. 55
5.19 AlexNet accuracy on train dataset. 55
5.20 AlexNet loss on train dataset. 55
5.21 InceptionV3 accuracy on validation dataset. 56
5.22 InceptionV3 loss on validation dataset. 56
5.23 ResNet-18 accuracy on validation dataset. 56
5.24 ResNet-18 loss on validation dataset. 56
5.25 VGG-16 accuracy on validation dataset. 57
5.26 VGG-16 loss on validation dataset. 57

5.27 AlexNet accuracy on validation dataset. 57
5.28 AlexNet loss on validation dataset. 57
5.29 Confusion matrix on the test set for the InceptionV3 model using original images from the ADNI dataset. 57
5.30 Confusion matrix on the test set for the InceptionV3 model using DCGAN synthetic image augmentation for the AD class. 58
5.31 Confusion matrix on the test set for the AlexNet model using original images from the ADNI dataset. 59
5.32 Confusion matrix on the test set for the AlexNet model using DCGAN synthetic image augmentation for the AD class. 59

1 First part of the advanced search section for the ADNI dataset. It contains the following: project phase, research group (CN, MCI, or AD), and type of image (Original). 75
2 Second part of the advanced search section for the ADNI dataset. It contains the following: image modality (MRI), acquisition plane (Axial), acquisition type (2D) and weighting (T2). 75
3 First page of the paper submitted to Intelligent System with Applications Journal. 76

Chapter 1

Introduction

1.1 Problem description

In the field of Artificial Intelligence there are increasing challenges, and one of them is dataset imbalance. This is a problem that is becoming increasingly common and demanding in terms of research and development of techniques to mitigate this problem [8, 9]. In many datasets the classes or categories of interest are not uniformly distributed [10]. In other words, some classes in the set have more images or data than others in the set. This can cause that when training models to solve specific problems such as classification or segmentation, they tend to favor only the class that has the largest amount of data. This makes it much more difficult to identify classes with minority data in the set. Additionally, models trained on unbalanced data sets may have difficulty generalizing to new samples in which minority classes need to be correctly identified. In the area of medicine, class imbalance is a significant problem that is caused by the limited availability of images [11]. There are several reasons why the number of images is limited within the medical industry. One of them is related to the privacy and confidentiality of patient information. Medical images, such as X-rays, computed tomography (CT) scans, and magnetic resonance imaging (MRI), contain sensitive information about patients health. To protect privacy, strict restrictions are placed on the collection, storage and use of these images. Therefore, in many cases, the explicit consent of the patient is required to use their medical images in research or in the creation of data sets. However, obtaining this consent can be a complicated process and may further limit the availability of medical images for research purposes. In addition to privacy concerns, there are legal regulations and ethical codes that govern the collection and use of medical data, including images [12, 13, 14]. For this reason, some countries establish strict restrictions on how medical images can be collected, shared and used. This can make it difficult to build large public data sets. All these cases taken to a particular case such as the early detection of Alzheimer disease make this a difficult disease to identify. In the case of Alzheimer disease, obtaining labeled data from patients at different stages of the disease can be complicated due to privacy and consent issues, limiting the amount of data available for training models. In fact, the existing datasets for current classification tasks are extremely limited to two or three of the most used. To detect Alzheimer disease, it is necessary to have a set of medical images from brain magnetic resonance imaging (MRI) or positron emission tomography (PET) scans, which are subject to characteristics such as age, gender, acquisition conditions, among others. All of these factors contribute to increasing the challenge of detecting Alzheimer disease according to the different stages of the disease.

1.2 Justification

One of the most promising and widely used techniques for synthesized data generation has been Generative Adversarial Networks (GANs) [15, 16]. Specifically, within the family of GANs, Deep Convolutional GANs (DCGANs) have been highlighted as a powerful tool for generating realistic synthetic data. The DCGANs use deep convolutional neural network architectures that are highly effective in capturing patterns and features in visual data. Therefore, this is extremely useful in medical imaging, where as many relevant features as possible must be preserved to aid in the disease detection. The detection of diseases in a more efficient way has several relevant points. Early and accurate detection of diseases, such as Alzheimer's, is essential for several fundamental reasons that have a significant impact on the quality of life of patients and the effectiveness of treatments. Early detection allows intervention in the early stages of the disease, when brain damage is minimal. In addition, in many cases there are treatments and therapies available that can help slow the progression of the disease or alleviate symptoms. Finally, linking the concepts of realistic synthetic image generation related to the medical industry represents a promising strategy to improve the capability of classification models in the detection of Alzheimer's disease.

1.3 Objectives

1.3.1 General Objective

Develop a generator of representative synthetic images using a Deep Convolutional Generative Adversarial Network (DCGAN) and a qualitative visualization with the t-Distributed Stochastic Neighbor Embedding (t-SNE) checking the data distribution to improve the accuracy in Alzheimer disease classification.

1.3.2 Specific Objectives

- Investigate the recommendations for using a Deep Convolutional Generative Adversarial Network as a method for the generation of synthetic images in medical applications.
- Collect medical imaging data on Alzheimer disease including real images and labels of the corresponding stages.
- Establish criteria to select Alzheimer disease images useful for the classification into different stages.
- Design and train a DCGAN model capable of generating synthetic images that capture the relevant features of Alzheimer disease.
- Evaluate the quality of the synthetic images generated using qualitative metric based on t-SNE and quantitative metrics.
- Compare the performance of different classification models with the original data and the synthetic images.

1.4 Contribution

In the present work, we propose the use of DCGANs to generate quality synthetic images that help in the efficient classification of the various stages of Alzheimer disease. In order to achieve efficient DCGAN performance, necessary hyperparameters are adjusted to contribute to the generation of more realistic images that help to preserve as many features of the original images as possible. On the other hand, a verification method based on the t-Stochastic Neighbor Embedding (t-SNE) algorithm is also proposed to help guide the generation of synthetic images. With the implementation of this algorithm we can check that the characteristics of the synthesized images correspond as closely as possible to the characteristics of the original images in a visual and interactive way. Finally, two different approaches are proposed to evaluate the improvement introduced by synthetic images from DCGAN for the classification of various stages of Alzheimer's disease.

Chapter 2

Theoretical Framework

2.1 Generative Adversarial Networks (GANs)

They are models based on deep neural networks proposed by Ian Goodfellow, et al, in their work ”On distinguishability criteria for estimating generative models” [17]. Specifically, these generative models are constituted by two deep neural networks. First, the generator network, which is in charge of generating synthetic images that look very similar to the training images and then sent to the discriminator. Second, the discriminator network, which functions as a binary classifier, determining whether a generated image is real or fake [17]. This discriminator network takes into account two inputs, which are the images from the real dataset and the images coming from the generator network. Both networks are adversarial because they are in a constant zero-sum adversarial game. This means that the gain or loss of one of the networks is offset by the gain or loss of the other. Through this game the model learns from a generator that creates realistic data. Its training is based on using a set of real images. At this stage the generator looks for ways to generate better false images that can fool the discriminator into believing that the generated images are real. Likewise, the discriminator works to make better detections and classify which of the images it receives are real or fake.

Based on the game theory, there are two networks in GANs, one is generator, and the other is discriminator. The role of the generator is to create as realistic data as possible to deceive the discriminator. The role of the discriminator tries to distinguish fake samples from real ones. GANs are composed of two neural networks: a generator and a discriminator. The generator is responsible for creating synthetic data, while the discriminator evaluates the authenticity of the generated data. The GAN training process is based on a competition between these two networks, where the generator tries to create data that deceives the discriminator and the discriminator tries to distinguish between the generated data and the real data in the training data set.

The training process of a GAN is carried out in several iterations. In each iteration, the generator creates synthetic data from a random noise distribution. The discriminator evaluates the authenticity of the generated data and the real data in the training data set and adjusts its weights to improve its ability to distinguish between the two types of data. In turn, the generator adjusts its weights to try to create data that will fool the discriminator. This process of competition and weight adjustment is repeated until the generator is able to create synthetic data that is indistinguishable from the real data in the training data set. The following functions are used for this purpose:

$$log(D(X)) + log(1 - D(G(Z))) \tag{2.1}$$

Where $G(Z)$ represents the generator output when given an input Z, $D(G(Z))$ represents the probability that the discriminator classifies the generator output as real (1) rather than false (0). Furthermore, $1 - D(G(Z))$ is the complementary probability, that is, the probability that the discriminator classifies the generator output as false (0) instead of real (1). Additionally, $D(X)$ represents the probability that the discriminator classifies an arbitrary real sample as real. In summary, these expressions are components of the loss function used in GAN. The generator aims to minimize the function $log(1 - D(G(G(Z)))$, which means that it tries to make its output as realistic as possible. On the other hand, the discriminator aims to maximize the function $log(D(X)) + log(1 - D(G(G(Z)))$, which means that it tries to correctly classify both real and the synthetic generated samples.

Once the GAN has been trained, the generator can be used to create new synthetic data that is similar to the real data. For example, in the case of image generation, the generator can create synthetic images that are similar to the real images in the training dataset. Figure 2.1 represents the architecture of a typical GAN and all its components. This figure begins with the detail of the initialization of the generation by means of the noise vector. This is followed by the generation of synthesized images and the input of the set of real images to be sent to the discriminator. Finally, the discriminator feeds back to the network with its classification of true or false according to the image it receives.

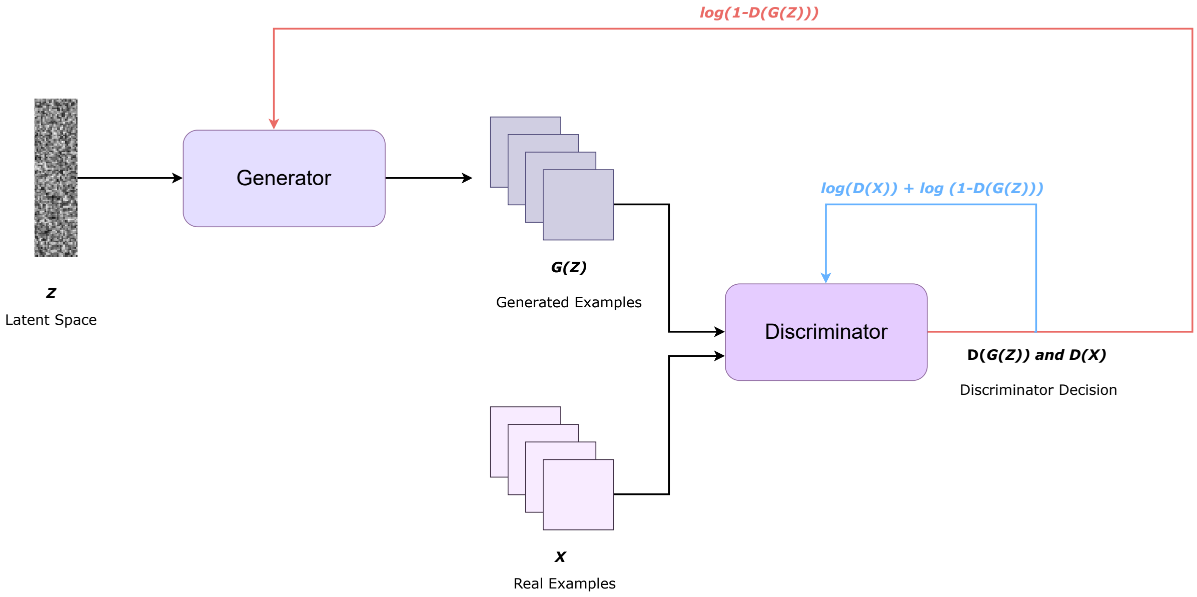


Figure 2.1: Typical Generative Adversarial Network (GAN) architecture.

2.1.1 Classification of GANs Models

Convolution based GANs

In Conv-GANs, both the generator and the discriminator are composed of convolutional layers, which are able to learn spatial features in data such as images. These convolutional layers are fundamental in feature extraction and image processing, as they are able to capture local and global patterns in the input data. Compared to traditional GANs that use fully connected layers, Conv-GANs are more efficient in image processing, as they can take advantage of the structure of images to extract useful information. In addition, Conv-GANs allow for greater flexibility in image generation [18].

Condition based GANs

Allow the user to control and condition the generation of images from a class label or a specific feature vector. In a cGAN, both the generator and the discriminator receive not only the random noise input, but also a class label or a feature vector as conditional input. The class label provides additional information about the object being generated, while the feature vector provides a detailed description of the specific features you want the generated image to have [19].

Autoencoder based GANs

They are a combination of two types of generative models: autoencoders (AE) and generative adversarial networks (GAN). In AE-GANs, the GAN generator is replaced by an autoencoder, consisting of an encoder that maps the input data to a latent representation and a decoder that reconstructs the input data from the latent representation. The GAN discriminator remains the same, trying to distinguish between generated and real images. In addition, the use of AE instead of GANs as a generator can also help overcome some of the instability and collapse problems of the GAN [20].

2.1.2 GAN Models

Deep Convolutional Generative Adversarial Network (DCGAN)

This derivative of GAN was proposed by Radford, et al [21] in their paper entitled ”Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”. The Deep Convolutional Generative Adversarial Networks (DCGANs) are based on the operation of traditional GANs but with the addition of a convolution layer (CNN) to replace the multilayer perceptron. The convolution layer part is used to discriminate between the images received by the discriminative network, and the deconvolution layer part is used to generate the images in the generative network. It uses deep convolutional layers instead of fully connected layers. Fully connected layers are layers of neural networks where each neuron is connected to all neurons in the previous layer and the next layer. Deep convolutional layers, on the other hand, use convolutional filters to extract features from the input images. These layers are more efficient than fully connected layers in image generation, as they take advantage of the spatial structure of the image to reduce the number of parameters that need to be trained. Convolutional operations allow large volumes of data to be processed more efficiently, which can reduce training times and memory requirements [22, 23]. As a result, in contrast to a traditional GAN, in DCGAN convolutional operations are computationally more efficient than fully connected operations, especially in image-related tasks. This is because convolutional operations share parameters and take advantage of the spatial structure of the data, which significantly reduces the number of parameters and computations required. In addition, DCGANs typically use convolutional layers with relatively small filter sizes compared to the fully connected layers used in traditional GANs. This reduces the amount of parameters and computations needed in the network, resulting in less computational overhead.

DCGAN Generator

This generator network takes a 100 dimensional uniform distribution Z random noise input. Then the noise inputs are fed through a fully connected layer that resizes it and converts it to a size suitable for the next step. Next, transposed convolution layers or unsampling layers are used to increase the dimension of the input tensor until the desired dimension is obtained. This increase occurs in height and width, and simultaneously a reduction in channels occurs. These transposed convolution layers help to decompress the noise to be transformed into a more detailed representation. After each transposed convolution stage a normalization layer is applied, in this case Batch Normalization, which helps normalize the output tensor by adjusting the mean and variance when the input data is skewed, helping to stabilize the network training process. An activation function, specifically the Rectified Linear Unit (ReLu) is then applied to introduce nonlinearity into the network. This process of applying transposed convolutions, normalization and activation is repeated several times on the generator network to adjust and refine the representation to be as close as possible to the real data. Finally, for the output layer a transposed convolution with the hyperbolic tangent function (Tanh) is used to return the output to the input range of [-1,1]. In the case of using color images, the output is a 3-dimensional tensor representing the color channels (red, green and blue). A representation of the parts of this architecture is shown in Figure 2.2.

DCGAN Discriminator

This discriminative network takes as input a synthetic image generated by the generator or a real image extracted from the training dataset. The discriminator uses multiple con-

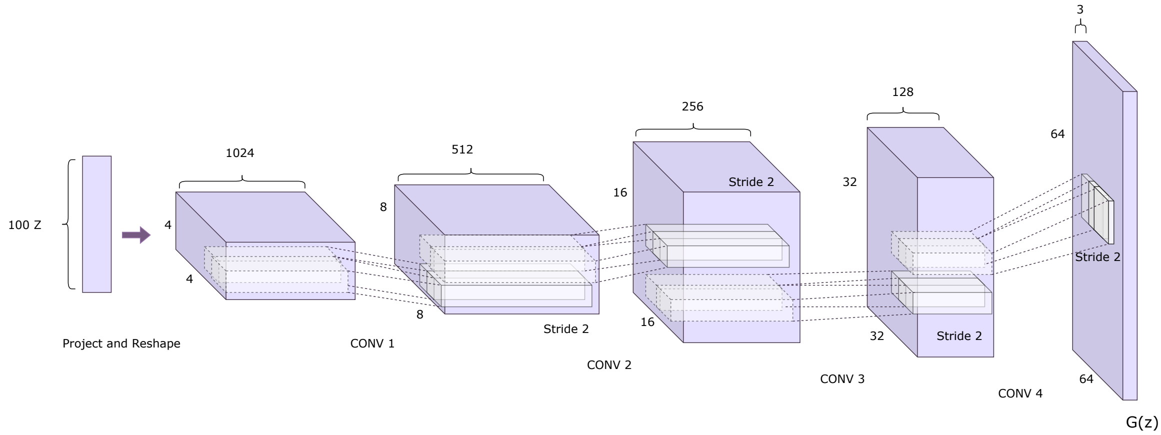


Figure 2.2: DCGAN Generator Architecture. A 100-dimensional uniform distribution (Z) is projected to a small spatial convolutional representation with some feature maps. This is followed by four fractionally-strided convolutions. Finally, this converts to a representation of 64x64 pixel image.

volutional layers to process the input image in a progressive manner. These convolutional layers apply filters to extract relevant image features such as edges, textures and patterns. This image passes through several convolution layers that seek to extract relevant features from the image. These features can be edges, textures or shapes that are specific patterns of the image. After each convolution layer a normalization layer is applied, in this case Batch Normalization, followed by an activation function such as LeakyReLu in this case. Then the final probability is generated as output by means of the Sigmoid activation function. For this step fully connected layers are used to classify the features and produce the final output, which is nothing more than the probability that the image is real or fake by means of a binary classification (0 indicates that the image is fake, 1 indicates that the image is real). A representation of the parts of this architecture is shown in Figure 2.3.

2.1.3 Hyperparameters

For the DCGAN training process, certain settings and values must be followed and adjusted before training the model to obtain the best results. Major hyperparameters include the following:

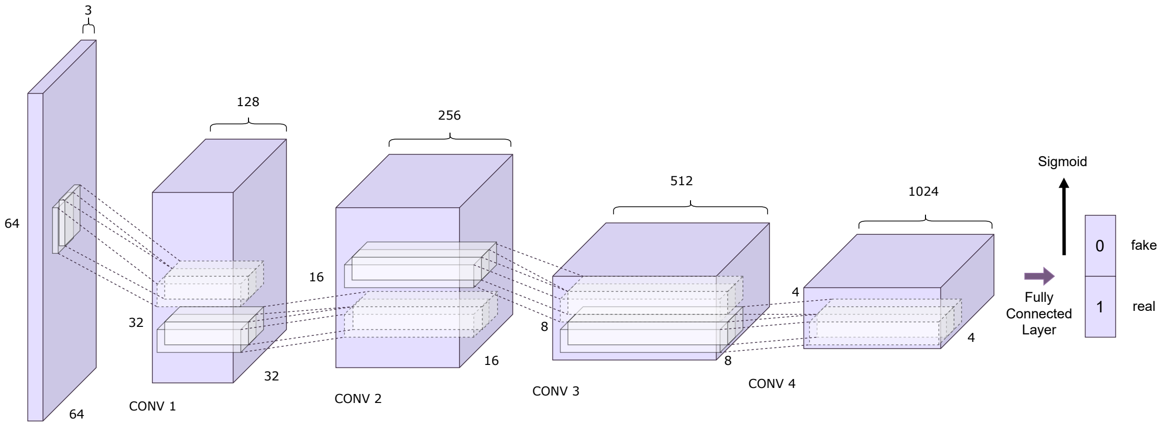


Figure 2.3: DCGAN Discriminator Architecture. It uses multiple convolutional layers to process the input image progressively. The last layer uses a Sigmoid function to produce values between 0 (fake) and 1 (real).

Number of epochs

This is the number of times the full model sees the entire training data set. It determines how many times the network weights will be adjusted during training. Choosing the right number of epochs is crucial to avoid underfitting and overfitting problems.

Batch size

It is the number of training examples that are used in each iteration during training. A larger batch size tends to converge faster, but may also increase the risk of overfitting, especially if the training data set is small. A small batch size can help better generalization, as noise in the weight updates can help prevent overfitting. It can affect the training speed and the stability of the model.

Optimizer

The optimizer determines how the network weights are updated during training. Common optimizers include stochastic gradient descent (SGD) and Adam for this type of architecture.

Learning rate

This is the rate at which the model adjusts its weights during training. Too high a learning rate can cause the model to start to fail, while too low a learning rate can make training too slow. In other words, a high learning rate may cause the model to converge faster, but it may also cause it to cause divergence. A low learning rate leads to a more stable training process but can be slower and get stuck at local minimums.

Latent space dimension

This latent space is a low-dimensional mathematical representation that allows the model to learn and generate new data samples. The dimension of the latent space is the size of the input vector that is used as a seed to generate new samples.

2.2 Traditional data augmentation

This is a technique commonly used in machine learning and computer vision to improve the performance of machine learning models, especially in pattern recognition tasks such as image classification [24]. The idea behind data augmentation is to increase the amount of training data available by generating variants of existing data. The purpose of data augmentation is to increase the amount of effective training data by generating new samples from existing ones, introducing controlled variations and perturbations in the original data. With the arrival of deep learning and the increased availability of massive data sets, data augmentation has become even more important. Some of the techniques for data augmentation include: Rotation, which involves rotating the original image by a specific angle; horizontal mirroring, which is to reflect the image horizontally; shifting the image in a specific direction (horizontal or vertical); zooming, to enlarge or reduce the image; adjusting the brightness and contrast of the image; cropping a random part of the image; as well as other techniques.

2.3 T-Distributed Stochastic Neighbor Embedding (t-SNE)

To introduce this algorithm it is first necessary to talk about the Stochastic Neighbor Embedding algorithm. This is proposed by Hinton and Roweis [25] in order to represent high dimensional vectors in low dimensional spaces keeping most of the characteristics. In this way, points that are similar in the original space remain similar in the reduced space. For this purpose we have an object i and a potential neighbor j. The first step is to calculate the asymmetric probability, denoted by p_{ij}, that they are neighbors according to the following equation:

$$p_{ij} = \frac{exp(-d_{ij}^2)}{\sum_K exp(-d_{iK}^2)} \tag{2.2}$$

One of the main concepts introduced by this Equation 2.2 is the dissimilarity. For each pair of points in the original space, a dissimilarity measure is calculated. For this purpose, the notion of two-point difference of high dimensionality such as x_i, x_j is used:

$$d_{ij}^2 = \frac{\left\|x_i - x_j\right\|^2}{2\sigma_i^2} \tag{2.3}$$

where σ_i^2 is a measure of variance used to fit the distribution. Generally, it is chosen in a way that the probabilities of neighbors are relatively similar for all points. That is why the number of local neighbors or ”perplexity” is hand selected.

Now, once the dimensionality reduction has been performed, the probability of finding a neighbor in the lower dimensional space must be calculated. A similar process is performed to calculate the neighbor probability in the lower dimensional space and is denoted by q_{ij}. In this step, Gaussian neighborhoods with fixed variance are used to find the probability that a point is truly a neighbor. Thus, a point i will select a point j as its neighbor for images y_i in the low dimensional space. This is given by the following equation:

$$q_{ij} = \frac{exp(-\left\|y_i - y_j\right\|)^2}{\sum_K exp(-\left\|y_i - y_K\right\|)^2} \tag{2.4}$$

Therefore, the purpose of these steps is to make both distributions p_{ij} and q_{ij} match in the best way. Therefore, it is important to minimize the cost function by using the difference of the Kullback-Leiber divergences given by the following equation:

$$D_{KL}\left(P \parallel Q\right) = \sum_{S} P(s) ln(\frac{P(s)}{Q(s)}) \tag{2.5}$$

The goal is to minimize the KL divergence between the distributions by adjusting the coordinates of the points in the lower dimensional space so that the similarities between points in the original space are reflected in the lower dimensional space. By minimizing the KL, similar points in the original space have similar distributions. This results in a visual representation where similar points are close to each other.

With these concepts in mind, the t-SNE algorithm is now described. It is a dimensionality reduction algorithm used in machine learning and data visualization. It was developed by Laurens van der Maaten and Geoffrey Hinton in 2008 [26]. The main goal of t-SNE is to take high-dimensional data and map it to a lower-dimensional space, usually 2D. In this way, the patterns and relationships between the data are effectively preserved in the new space. In general, the main idea in the application of the t-SNE is the following: Calculate the similarities between pairs of data points in the original space. These similarities are usually expressed as conditional probabilities that represent the probability that two data points are close neighbors. Then, create a low-dimensional space and compute similarities between the same pairs of data points in that low-dimensional space. This is done using a Student's t-distribution to model the similarities in the low-dimensional space. Through an iterative process, the algorithm gradually adjusts the locations of the points in the low-dimensional space until the similarities are as close as possible to the similarities in the high-dimensional space [27, 26, 28]. This is done by minimizing a cost function that compares the similarities in both spaces. For this reason, this algorithm is particularly useful for data visualization in classification, clustering, text analysis problems, among others. To better understand the steps followed by the algorithm for the low dimensional representation, Figure 2.4 shows a general idea of how this algorithm works. The figure begins with a representation of object characteristics in a high dimensional space. Subsequently, these characteristics are transferred to a low dimensional space. Once in this space, we start by verifying the similarities between the existing points in the space. Those points with similar characteristics will move until all the points that share these characteristics in common are joined together.

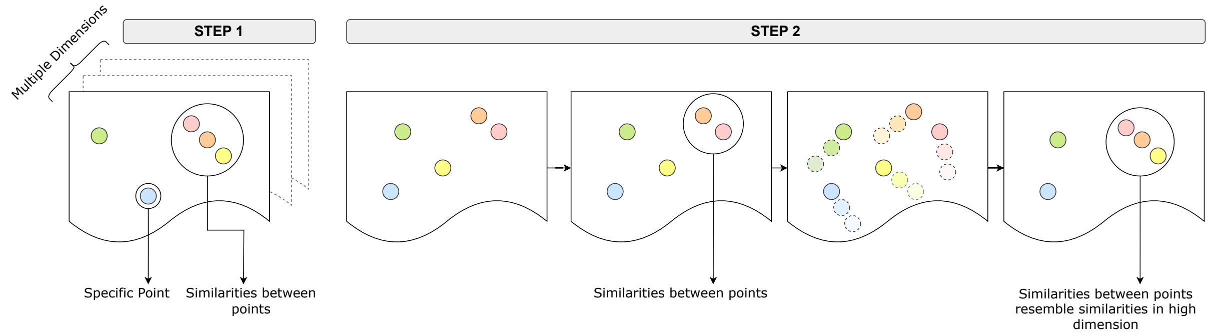

Multiple Dimensions
STEP 1
STEP 2
Specific Point
Similarities between points
Similarities between points
Similarities between points resemble similarities in high dimension

Figure 2.4: Example of how the t-SNE algorithm works. It starts with a high dimensional representation of points to be translated into a low dimensional space.

2.4 Convolutional neural networks

2.5 Convolutional neural networks for image classification

2.5.1 InceptionV3

Proposed by Christian Szehedy, et al, in 2014 for the challenge ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC 2014) [1]. It was designed to improve image classification performance compared to its predecessors, such as InceptionV1 or also known as GoogLeNet. The InceptionV3 architecture takes input images of size 299x299 pixels. The distinguishing feature of InceptionV3 is its Inception module, which is used repeatedly in the network. These Inception modules allow feature extraction at multiple scales and with different levels of abstraction in parallel. Each Inception module consists of multiple convolution filters of different sizes and dimensionality reduction operations, such as 1x1 dimensionality reduction. According to the reduction characteristic, this network uses a significant reduction factor. This results in a deeper but more computationally efficient network. This is achieved by reducing the dimensionality before and after the 3x3 and 5x5 convolution operations using 1x1 convolution layers. The reduction factor helps to reduce the number of parameters and at the same time improve computational efficiency. A more accurate representation of this architecture can be found in Figure 2.5.

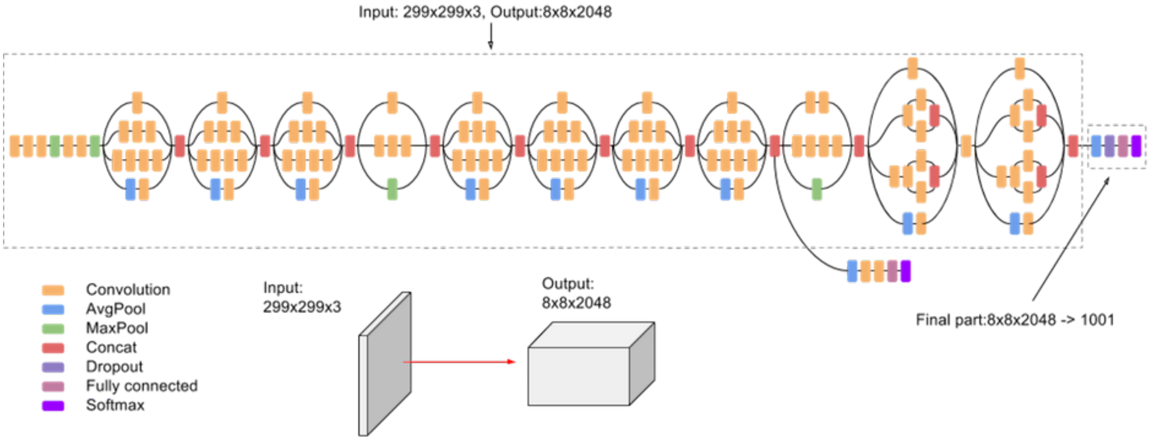


Figure 2.5: InceptionV3 architecture which consists of 48 layers formed by symmetrical and asymmetrical compilation blocks. These include convolutions, reductions and fully connected layers [1].

2.5.2 ResNet-18

Proposed by Kaiming He, et al, in the ILSVRC 2015 classification task [2]. This ResNet network introduced the concept of residual blocks to facilitate the training of deep networks. The ResNet-18 architecture is a smaller variant of the ResNet network that was designed to address the problem of gradient degradation in deep neural networks. This ResNet-18 consists of 18 layers in total, including convolutional layers, pooling layers and fully connected layers. The basic architecture of ResNet-18 is also organized in residual blocks. The main feature of ResNet networks is the use of residual blocks. Instead of trying to directly learn the desired function in a layer, residual blocks introduce a shortcut that allows information to flow directly from the input layers to the subsequent layers without performing any linear operation. This helps to avoid the problem of gradient degradation and facilitates the training of deep networks. In each residual block, the shortcut connection allows information to flow unchanged from one layer to the next. In addition, dimensions are adjusted when necessary by means of 1x1 convolution layers in the shortcut. After each residual block, a 2x2 max-pooling layer with a stride of 2 is used to reduce the spatial resolution and gradually increase the size of the receptive field. In addition, it uses ReLU activation functions after each convolutional layer to introduce nonlinearity. The network ends with a fully connected layer that produces the final output for the specific task, such as image classification. To obtain a more intuitive interpretation, the Softmax function is applied to the scores to convert them into probabilities. Softmax transforms the values obtained into a probability distribution where each value represents the estimated probability that the input belongs to a specific class. Figure 2.6 shows in detail each of the parameters that make up the network architecture.

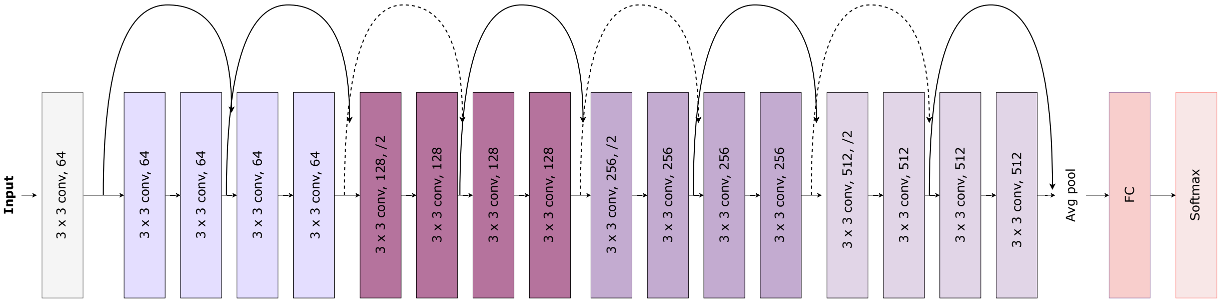


Figure 2.6: ResNet-18 architecture which consists of 18 layers in total, including convolution layers, pooling layers and fully connected layers [2].

2.5.3 AlexNet

Proposed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton in the image classification category of the ImageNet Challenge 2012 [3]. This network takes input images of size 224x224 pixels in three color channels (RGB). It has a total of 8 neural layers, including 5 convolution layers and 3 fully connected layers. The convolutional layers are composed of convolutional filters of different sizes, including 11x11, 5x5 and 3x3 filters. In addition, max-pooling layers are used to reduce the spatial resolution of the features. As for the activation function, this network uses the ReLu function after each convolutional layer. This introduces nonlinearity in the network and speeds up the training. The data is then passed through a set of fully connected layers that are used to perform the final classification. Before entering the fully connected layers, the output of the convolution layers are flattened into a one-dimensional vector. This means that all features extracted in the previous layers are converted into a long vector, which is used as input for the fully connected layers. For this purpose, three fully connected layers are added after the convolution process. The last fully connected layer in AlexNet corresponds to the number of classes in the classification problem. In the case of the original proposal, 1000 classes are split for the ImageNet dataset. A more precise representation of the architecture of this network is presented in Figure 2.7.

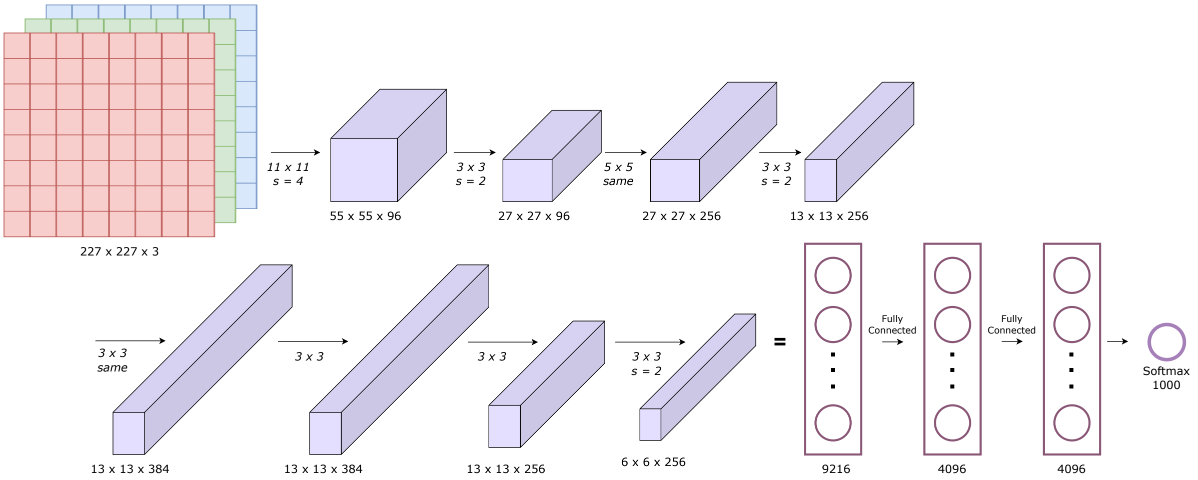


Figure 2.7: AlexNet architecture composed by 8 layers with weights, 5 convolutional layers, and 3 fully connected layers [3].

2.5.4 Very Deep Convolutional Network (VGG)

Karen Simonyan and Andrew Zisserman from Oxford University in 2014 presented a work named “Very Deep Convolutional Networks for Large-Scale Image Recognition” to introduce this network [29]. This architecture was notable for its simplicity and its ability to achieve high performance in computer vision tasks, especially in image classification. Initially, it takes 224x224 pixel input images, which are then processed through the convolutional and pooling layers. It consists of 16 convolutional and fully connected layers. This network is mainly composed of 3x3 convolutional layers with 1 pixel stride and 1 pixel padding to keep the image size. After each set of two convolutional layers, a 2x2 max-pooling layer with 2-pixel stride follows. This gradually reduces the spatial resolution of the features. In addition, the ReLU (Rectified Linear Unit) activation function is mainly used after each convolutional layer. This helps to introduce nonlinearity into the model and speeds up the training. Throughout the convolutional layers, weights are shared to maintain the simplicity of the architecture and reduce the number of trainable parameters. After the convolutional and max-pooling layers, VGG-16 includes three fully connected layers with 4096 units in each, followed by an output layer with the number of classes in the classification problem. Subsequent to the presentation of this network, variations of the network have been presented. VGG-19 is an extension of VGG-16, which includes 19 layers instead of 16. The only difference is that three additional convolutional layers are

Table 2.1: Main features of the convolutional networks used in this work.

Features	***InceptionV3***	***ResNet-18***	***AlexNet***	***VGG-16***
Year	2015	2015	2012	2014
Depth	48 layers	18 layers	8 layers	16 layers
Parameters	23 million	11 million	60 million	138 million
Input size	Typically 299x299x3	Typically 224x224x3	227x227x3	224x224x3
Branching	1x1, 3x3 and 5x5 convolutions in parallel	Uses shortcut connections in residual blocks	Not applicable, uses multiple cascading convolutional layers	Not applicable, uses multiple cascading convolutional layers
Size of convolutions	Conv1x1, Conv3x3, Conv5x5	Conv3x3, Conv1x1	Conv11x11, Conv5x5, Conv3x3	Conv3x3, Conv3x3, Conv3x3

added, which increases its capacity to learn more complex features. The details of the architecture can be seen in Figure 2.8.

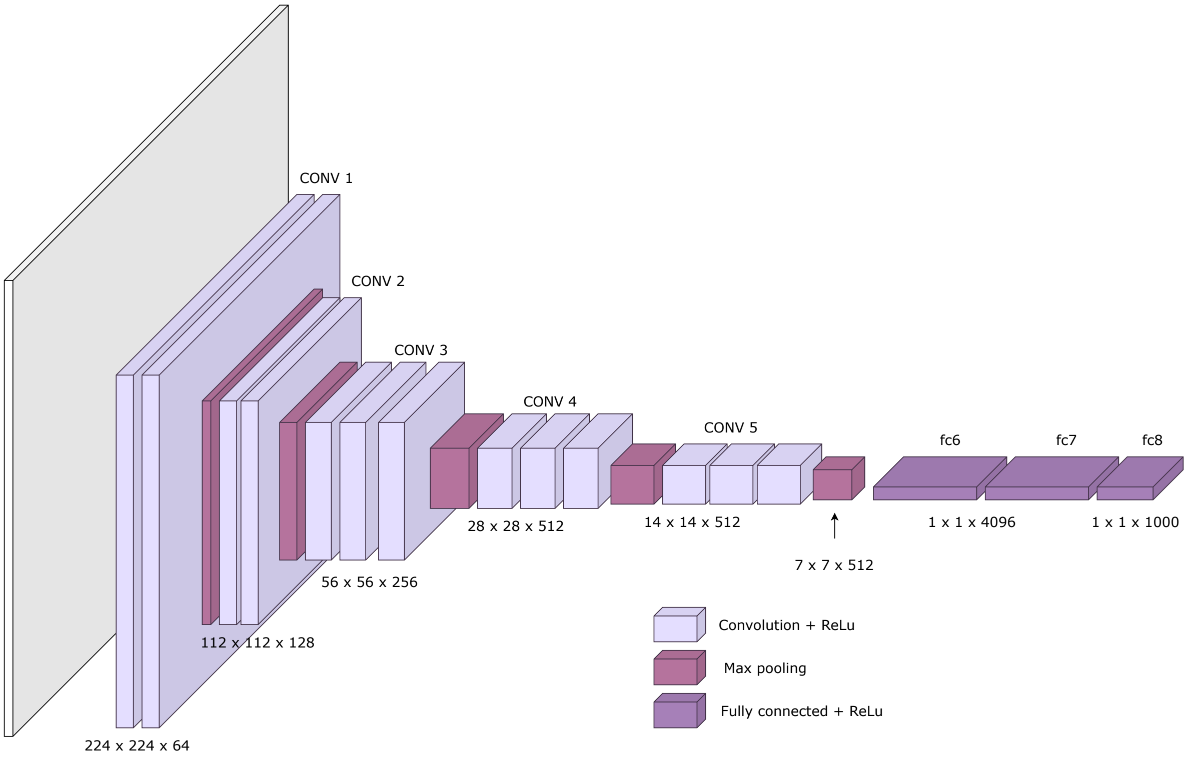


Figure 2.8: VGG-16 Architecture. It is composed of 13 convolutional layers, 5 max-pooling layers, and 3 fully connected layers[4].

2.6 Adagrad optimizer

An optimizer in the context of a classifier refers to an algorithm that is used to adjust the internal parameters of the classifier to minimize some loss function or error. The loss function measures how well or how poorly the classifier is performing compared to the training data and known labels. Optimizers are essential in the training process of a classifier, as they allow the model to learn from the data and adjust appropriately.

Adagrad, short for ”Adaptive Gradient Algorithm”, was proposed bu Duchi *et al.* in 2011 [30].It is an optimization algorithm used in the training of machine learning models, such as neural networks, to adjust model weights during the training process. The main feature of Adagrad is that it adapts the learning rate of each parameter in the model based on the gradient history of that parameter. In other words, parameters that have larger gradients will have a smaller learning rate, and parameters with smaller gradients will have a larger learning rate. La actualización de los parámetros θ en Adagrad se calcula de la siguiente manera para el paso de tiempo t in Equation 2.6:

$$\theta_{t+1,i} = \theta_{t,i} - \frac{n}{\sqrt{G_{t,ii} + \in}} \cdot g_{t,i} \tag{2.6}$$

Where $\theta_{t,i}$ is the value of the parameter i in the time t, n is the learning rate, $G_{t,ii}$ is the sum of the squares of the cumulative gradients up to time t for parameter i, $g_{t,i}$ is the gradient of parameter i at time t, and $\in$ is a small constant to avoid division by zero.

In the case of classification tasks, Adagrad automatically adjusts the learning rate for each model parameter. This is beneficial in classification tasks where different features may have very different scales and vary in importance. Adagrad allows features with large gradients to have a lower learning rate and vice versa, which can lead to faster convergence and better results. In classification tasks, it is common for some features to be infrequent but informative. Adagrad naturally treats these features by assigning them higher learning rates, which allows the model to learn from them more quickly. Thus, it is useful for classification tasks because it allows a dynamic adaptation of the learning rate for each model parameter, which can lead to faster convergence and better results, especially in scenarios where features have very different scales or are infrequent.

2.7 Alzheimer Disease and Dementia

The first approach to Alzheimer’s disease was given at the 37th Meeting of Southwest German Psychiatrists held in Tübingen on November 3, 1906. This disease was reported by the German physician Alois Alzheimer. The report included an unusual case study involving a peculiar severe disease process of the cerebral cortex [31]. Alois Alzheimer examined the case of a patient named Auguste Deter. Deter presented with unusual symptoms of memory loss, confusion, hallucinations and personality changes. After his death in 1906, Alzheimer performed an autopsy of his brain and discovered structural abnormalities. These findings provided the basis for the understanding of what would later become known as Alzheimer's disease. It was not until 1910 that the psychiatrist Emil Kraepelin used the term ”Alzheimer's disease” in his psychiatry textbook to describe the condition [32]. The use of the name Alzheimer's to describe the disease rather than just a descriptive term such as ”presenile dementia” or ”senile dementia” helped to consolidate the recognition and identity of this disease in the medical and scientific community [33]. This marked the official recognition of the disease and the beginning of its official name in the field of medicine.

2.8 Stages of Alzheimer's Disease

Alzheimer's disease is a progressive neurodegenerative disease characterized by a gradual deterioration of cognitive functions and memory [34]. In order to describe and understand the evolution of the disease over time, several classifications of its stages or phases have been proposed. These stages correspond to various stages of degeneration of the patient's neurological capacity. According to its severity, it can be considered within one of the following stages:

2.8.1 Cognitive Normal (CN)

The term ”cognitive normal” refers to a state in which a person has cognitive function considered to be within typical, healthy parameters for his or her age. Cognitive function refers to the mental ability to process information, which includes aspects such as memory, attention, perception, reasoning and problem-solving skills. When someone is said to be ”cognitively normal,” it means that his or her cognitive ability is in a range that is considered normal and shows no obvious signs of cognitive impairment or neurological disorder [35]. It is generally used in the context of cognitive function assessment to determine whether a person is experiencing any type of cognitive impairment or neurological disorder, such as Alzheimer's disease or mild cognitive impairment [36].

2.8.2 Mild Cognitive Impairment (MCI)

It is an intermediate stage between normal aging and dementia, such as Alzheimer's disease. It is characterized by cognitive decline that is more pronounced than expected for age, but does not meet the criteria for a full diagnosis of dementia [37]. People in the MCI stage report noticeable cognitive problems, such as difficulty remembering recent information, loss of short-term memory, problems in decision making, or difficulty in problem solving. Although MCI does not always lead to dementia, there is an increased risk of people with MCI developing dementia, especially Alzheimer's disease [38]. However, unlike dementia, in the MCI stage, daily living skills and functional independence are generally maintained. People with MCI can continue to perform daily activities without significant assistance [39].

2.8.3 Alzheimer Disease (AD)

Alzheimer's disease is a chronic, progressive neurodegenerative disease that affects the brain. It is the most common form of dementia in older adults. Alzheimer's disease is characterized by the gradual deterioration and loss of cognitive functions, such as memory, thinking, orientation, comprehension, calculation, learning ability and judgment. As the disease progresses, changes in personality and behavior may also appear [40, 41]. As the disease progresses, there is a significant loss of brain cells, resulting in a generalized contraction of the brain. Early symptoms of Alzheimer's disease often include short-term memory problems, temporal and spatial disorientation, difficulty finding words in conversation, and challenges in decision making and problem solving [42]. As the disease progresses, personality and behavioral changes such as agitation, depression, anxiety, apathy and sleep changes may appear. In short, this phase represents the manifestation of the disease.

2.9 Magnetic Resonance Imaging (MRI)

In general, it is a medical diagnostic technique used to obtain detailed images of the interior of the human body, including organs, soft tissues, muscles, bones, and the brain, without using ionizing radiation. This technique uses strong magnets and radio waves to create detailed images of the body. Inside an MRI machine is an extremely powerful magnet that creates a strong, uniform magnetic field around the area of the body to be examined [43]. This magnetic field aligns the hydrogen nuclei in the body's tissues. Brief pulses of radio waves are emitted in the area of the body under the magnetic field. These radio waves cause the hydrogen nuclei to get temporarily out of alignment. After the radio waves stop, the hydrogen nuclei return to their original alignment within the magnetic field, releasing energy in the process. The energy signals released by the hydrogen nuclei are detected by special antennas in the MRI machine. These signals are used to create detailed images of the internal structures of the body. Finally, the captured signals are processed by a computer to generate black and white or color three-dimensional images that represent the different features and tissues of the examined body area [44].

In the case of Alzheimer's detection, although MRI cannot directly diagnose Alzheimer's disease, it can play a crucial role in helping doctors rule out other causes of symptoms and assess the state of a patient's brain. MRI is used initially to rule out other medical conditions that can cause Alzheimer's-like symptoms, such as brain tumors, strokes, or other neurodegenerative diseases [45, 46]. MRI provides detailed images of the brain that allow doctors to identify any structural abnormalities. On the other hand, Alzheimer's disease is associated with progressive shrinking of the brain. MRI images can show changes in the size and shape of the brain over time, which can help doctors determine if a patient is experiencing brain atrophy characteristic of Alzheimer's disease. MRI is also used to track the progression of Alzheimer's disease over time. Repeated MRI images can show how brain structure changes as the disease progresses [47]. MRI can provide valuable information in the evaluation of patients with suspected Alzheimer's disease, the definitive diagnosis of the disease is usually based on a combination of clinical evaluation, neuropsychological tests, among others .

2.9.1 Relaxation Times

Longitudinal relaxation time (T1)

This refers to the time taken for the atomic nuclei in a tissue to return to their equilibrium state in the direction of the main magnetic field after being excited by a radiofrequency pulse. In simpler words, the T1 mode type images represent the speed at which these nuclei return to their original position in the magnetic field. Keeping this concept of time in mind, in T1-weighted images the tissues with short T1 relaxation times appear brighter. On the other hand, those tissues with long T1 relaxation times appear darker. The relevance of using this type of imaging is that brain structures and anatomical details of the brain can be better visualized [48].

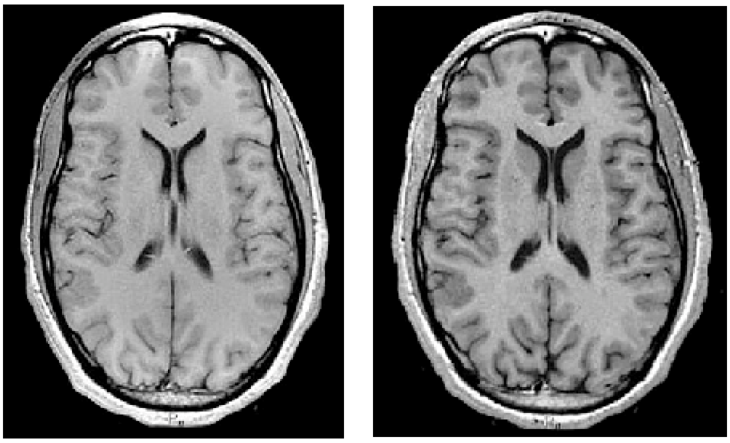


Figure 2.9: Axial representation of an MRI with Longitudinal relaxation time (T1) [5]

Transversal relaxation time (T2)

This refers to the time it takes for atomic nuclei to lose coherence with each other after a radiofrequency stimulus. In other words, it is the time it takes for the nuclei of the atoms to lose their alignment after being perturbed. For this reason, in T2-weighted images, tissues with long T2 relaxation times appear brighter, while those with short T2 relaxation times appear darker. The relevance of using this type of imaging is its usefulness for visualizing fluids in the body, such as cerebrospinal fluid in the brain. In addition, T2 sequence allows a clearer visualization of anatomical structures in the brain. This can help identify structural abnormalities, such as brain atrophy, dilation of the cerebral ventricles, or changes in the shape and size of brain structures. It is also useful for visualizing changes in the white matter of the brain. This is important in the study of neurodegenerative diseases [48].

2.9.2 Acquisition plane

To perform an MRI, you can select different directions in which the images will be obtained. This allows you to obtain timely information about the field of exploration that has been

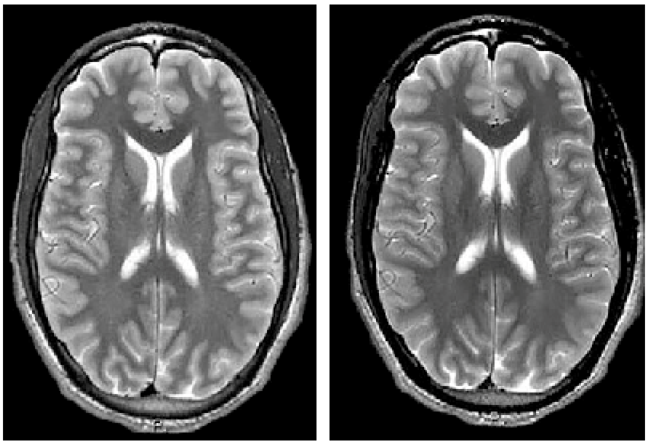


Figure 2.10: Axial representation of an MRI with Transversal relaxation time (T2) [5].

selected with the MRI. Depending on the area, different anatomical structures of the body can be visualized according to the selected plane. There are three acquisition planes for MRI images: axial plane, sagittal plane and coronal plane. A visual representation of the three acquisition planes mentioned above is shown in Figure 2.11. For the purposes of this work, the plane selected for image acquisition corresponds to an axial plane.

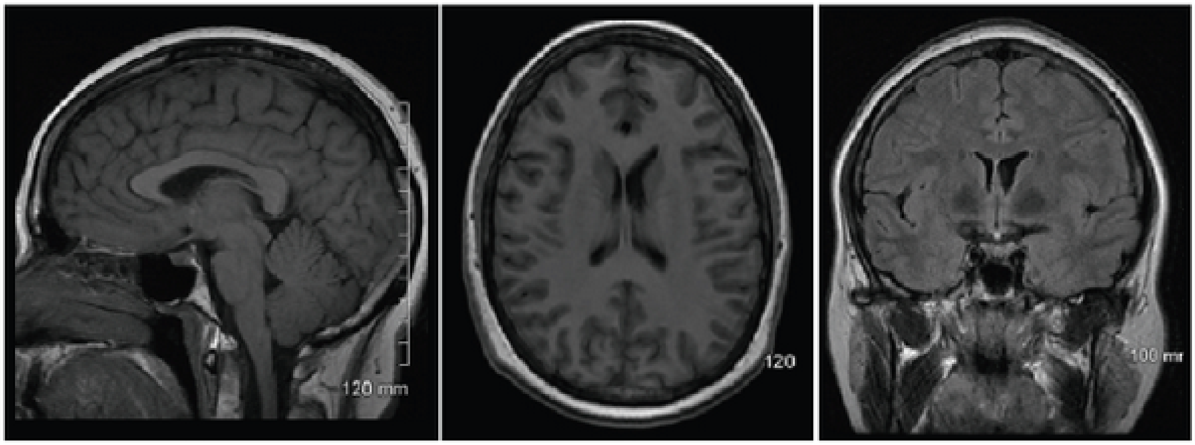


Figure 2.11: Representation of the three acquisition planes in order from left to right: sagittal, axial and coronal [6].

Axial plane

With this type of plane, a division of the patient is obtained both top and bottom in a horizontal way. The axial plane is oriented horizontally across the body, so that the resulting images divide the body into cross sections. Thanks to this type of slice it is possible to study the structures of the brain that extend from right to left in totality. As the MRI scan is performed, the body is cut into sections. Therefore, multiple axial sections are acquired along the length of the body. In these images, the upper part represents the patient's head, while the lower part corresponds to the feet [49]. In the head, it allows to view the brain in cross sections that are useful for the detection of brain tumors, lesions, hemorrhages and other neurological disorders.

Chapter 3

State of the Art

The use of diverse Deep Learning techniques to solve medical problems is becoming increasingly important. In our case, it is important to focus on traditional classification techniques and how they have been improved with the use of Deep Learning. In addition, with the advent of novel techniques such as GANs, the landscape of medical disease detection and classification work has greatly improved. In this introduction to the state of the art, we will explore the most outstanding contributions in the application of GANs for Alzheimer's related image classification. Specifically, work related to the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset will be addressed.

3.1 Machine learning techniques for ADNI dataset classification

Effectively classifying data of any kind is a fundamental challenge. Especially, this has become very relevant within the medical industry due to the massive amount of information. In this context, the use of machine learning techniques has shown promise in identifying patterns, predicting diagnoses and contributing to the understanding of neurodegenerative diseases. For this reason, machine learning based techniques were relevant enough to introduce to image classification tasks. This was thanks to the variety of existing methods that could be applied for various tasks. In this sense, Cuingnet *et al.* [50], proposed the use of Support Vector Machine to classify various stages of the disease including: cognitive normal (CN), mild cognitive impairment (MCI) and alzheimer disease (AD). In addition, they have ten different approaches (five voxel-based methods, three cortical thickness-based methods and two hippocampal-based methods) to evaluate the classification performance of the presented disease stages. At the end, the authors can conclude that the best binary classification they could obtain was with the AD vs CN classes. For this experiment, the sensitivity values were 81% and 95% for the specificity metric. On the contrary, for the classification of CN vs. MCI the sensitivity values were much lower due to the overlap of both classes.

Along the same lines, Yang *et al.* [51], set out the importance of recognizing changes in the structure, blood flow in the brain, and other parameters of people with Alzheimer's disease. For this purpose, they propose the use of SVM for classification of MRI images of CN, MCI and AD stages. In addition, they propose combining this technique with Independent Component Analysis (ICA) to extract important information about brain structures. Specifically, they treated important areas such as gray matter and whole brain imaging to analyze important alterations as the disease progresses. With these parameters, they were able to obtain an accuracy of 76.9% for whole brain images and an accuracy of 80.7% with images focused only on gray matter.

Another relevant approach is given by Campos *et al.* [52], who addressed the problem of classifying images with missing data. For this purpose, they propose the use of imputation methods such as mean, median, zero-filling, k-nearest neighbours (kNN), among others. With this idea they randomly remove from 10% to 50% of information from CSF, MRI, PET, CSF-MRI, CSF-PET, CSF-PET and MRI-PET images of various stages of Alzheimer's disease such as CN, MCI and AD. Finally, a comparison is made on the classification of images using imputation methods. Respectively, SVM and random forest (RF) are used and the accuracy values obtained for each method are compared. In the case of SVM, better results were obtained for the classification of AD vs. CN with an accuracy of 88.7% with the zero-filling technique. For the same comparison, in the case of RF, a better accuracy of 88.4% was obtained with the imputation Winsor technique. On the other hand, for the classification of ICM and CN using SVM, a better accuracy of 73.1% was obtained with the expectation maximisation technique. Meanwhile, with the use of RF, a better accuracy of 73.6% was obtained with the zero-filling technique.

Lastly, Shahbaz *et al.* [53], proposed an extensive comparison of four machine learning algorithms for the classification of five different stages of Alzheimer's disease. The algorithms they compare are: k-nearest neighbours (k-NN), decision tree (DT), rule induction, Naive Bayes (NB) and generalized linear model (GLM). The five stages of the disease studied are: cognitively normal (CN), early mild cognitive impairment (EMCI), late mild cognitive impairment (LMCI), subjective memory complaint (SMC), and alzheimer's disease (AD). With these parameters, an accuracy of 88.24% was obtained in the test set using the GLM algorithm. In addition, with this model, a recall of 100%, 94.44%, 90.62%, and 86.01% was obtained for the AD, CN, EMCI and LMCI classes, respectively.

3.2 Deep Learning techniques for ADNI dataset classification

The exponential growth in the availability of medical data has led to a growing interest in the use of advanced deep learning techniques. This has allowed for more promising and accurate results, which is of vital importance in the field of medicine. For this reason, it is important to analyze the advances in deep learning techniques applied to ADNI data classification. To begin with, we have the work proposed by Sarraf *et al.* [54] in which a LeNet-5 convolutional neural network is used to classify people with normal Alzheimer's disease control. For this purpose, they used magnetic resonance imaging with T1 relaxation time. With this data characterization they obtained a 96.85% accuracy rate for characterizing healthy patient data from MRI brain scans.

In a similar line of ideas, Korolev *et al.* [55] propose the use of simple 3D convolutional networks and residuals but omitting the feature extraction step. They use a VoxCNN and a ResNet with 21 layers. The authors' idea is to compare a traditional implementation with convolutional sections and pooling layers and an implementation with residual networks. As for the data, they used images corresponding to classes AD, LMCI, EMCI, and CN. All these images correspond to MRI brain scans with T1 relaxation time. With this in mind there are six possible combinations for the evaluation of disease stage classification. Thus, the best result obtained corresponds to the AD vs. CN classification with the ResNet network obtaining 80% accuracy. Similarly, Basaia *et al.* [56] propose the use of convolutional networks (CNN) for AD, CN and Mild Cognitive Impairment (MCI) classification. For the images they used MRI scans with a T1 and T2 relaxation time with a 3D plane. The most favorable results were for the classification of the AD vs. CN stages with 99% accuracy.

Another work that stands out within this theme is the one proposed by Cheng *et al.* [57] with the implementation of multilevel convolutional neural networks for feature combination. First, the 3D CNN network is constructed for each of the image types used in the work. These images correspond to 3D images from MRI and Positron emission tomography (PET) respectively. With this step, the features of each type of image are learned and propagated to the next network. This network corresponds to a 2D CNN that combines the learned features and performs the final classification. For this purpose they used images of the AD and CN stages to which data augmentation techniques such as shift, sampling and rotation were applied. This was done due to the limited amount of data originally existing for the training set. With the cascade method proposed by the authors, they obtained an accuracy of 87.13% for PET images and an accuracy of 85.47% for MRI images. Moreover, there are other approaches related to the use of CNN for Alzheimer stages classification. For this purpose, Choi *et al.* [58] proposed the use of a CNN to classify the stages CN, MCI and AD from images from MRI brain scans. In addition, the images have a T1 relaxation time and the protocol used is a 3D sagittal plane sequence. The relevant point of this work is the use of the Consortium for Brain Mapping template (ICBM) to highlight important aspects of the brain according to the stage of the disease. In other words, it helps to extract relevant information related to the coordinates of the brain. With these conditions, the authors obtained an accuracy of 92.3% for the classification of AD vs CN. However, favorable results were also obtained for the AD vs MCI classification with an accuracy of 85.6%.

There are also other contributions using deep neural networks such as the one proposed by Prajapati *et al.* [59]. In their paper, the authors propose a deep neural network (DNN) with fully connected layers in order to perform a binary classification. The classes used to perform the classification are CN, MCI, and AD to perform binary classifications between these classes. To analyze the performance of the proposed network, a comparison is made with traditional ML techniques such as Naive Bayes, KNN, CART and SVM. The best validation accuracy score obtained 85.19%, 76.93%, and 72.73% accuracy on the test data for AD vs. CN, Mild Cognitive Impairment (MCI) vs. CN, and AD vs. MCI classifications, respectively. Similarly, Simon *et al.* [60] proposed the transfer learning approach using pre-trained neural networks. The networks used are: AlexNet, ResNet and GoogleNet which are modified to are trained with the new images from Alzheimer. The classes used for this work are CN, EMCL, MCI, LMCI and AD and the type of images are from MRI, PET and fMRI scans respectively. Once all networks were trained under the same conditions, the best performance was obtained for AlexNet with a near accuracy of 95%, followed by ResNet18 and finally GoogleNet.

3.3 GAN-based techniques for data augmentation of ADNI dataset

The generation of synthetic data through techniques based on GAN (Generative Adversarial Networks) has revolutionized the field of data mining in the area of deep learning. GANs offer a unique and powerful way to create synthetic samples that are indistinguishable from real data, making them an essential tool for improving the effectiveness of deep learning models in a wide range of applications. For this reason it is important to analyze GAN-based techniques for data mining and their impact on the improvement of existing models. In this sense, Konidaris *et al.* [61] propose the use of a traditional GAN for data augmentation to help improve image classification tasks. A comparison of traditional data augmentation techniques including: horizontal flip, brightness adjustment, scaling, and rotation is performed. A ResNet18 network was used for image classification. At the end, using only the GAN synthesized data, an accuracy of 72.81% was obtained compared to the original data classification which only achieved 63.11%. Following the same idea of using traditional GANs, Shin *et al.* [62] achieved to generate synthetic MRI-derived images for Alzheimer stages for segmentation models. Images of AD and MCI stages with T1 relaxation time are used. To achieve a correct segmentation, they extract the information of the anatomy of the brain according to the stage of the disease and based on it, synthesized images are generated with the GAN. With this they can manage to generate synthesized images that allow patient anonymity, so the data can be used without so many restrictions for any segmentation model.

In addition to the application of traditional GAN, variations of GAN have also been developed to improve the quality of the resulting images. For this, Baumgartner *et al.* [63] propose to address the problem of visual attributes in brain imaging through the use of Wasserstein GANs. The important point is to recognize the relevant regions of the brain that are affected as Alzheimer's disease progresses. Therefore, they use the MCI and AD classes with a T1 relaxation time for MRI images. Thanks to these considerations, it is possible to generate more realistic synthetic images that keep the original relationship of the brain architecture according to the stage of the disease. In the literature there are also several applications for synthetic image generation using the Deep Convolutional Generative Adversarial Network (DCGAN). For instance, Sajjad *et al.* [64] propose the implementation of a DCGAN to solve the data imbalance problem. Specifically, they work with the CN, MCI and AD classes under the PET imaging modality. Subsequently, with the images generated, a classification process is developed with the use of a VGG16. Finally, a final accuracy of 72% was obtained for the synthetic images generated. Examining the binary classifications, a better percentage of accuracy was obtained for the AD vs CN classification with a value of 83%. Moreover, Islam *et al.* [65] employed a DCGAN for the generation of synthesized images for stages of Alzheimer's disease. For this purpose, they use CN, MCI and AD classes from PET scan images. For the evaluation of the quality of the synthesized images generated, the authors propose the quantitative metrics peak signal to noise ratio (PSNR) and structural similarity Iniex (SSIM). In the case of the PSNR metric, a value of 32.83 was obtained. For the SSIM metric, a value of 77.48 was obtained. With the first metric it can be deduced that there is less error between the original and the generated images. This represents greater visual similarity. With the second metric it can be estimated that the similarity of the original pixels and the generated images is quite high. This means that it is related to the original characteristics of the images.

On the other hand, there are also much more advanced and complex derivations of GANs for synthetic image generation. For this reason, we can mention the work of Mukherkjee *et al.* [66] who propose the implementation of an AGGrGAN. This is a GAN derived from the use of a Deep Convolutional Generative Adversarial Adversarial Network (DCGAN) and a Wasserstein GAN (WGAN). The purpose of this implementation is to generate synthetic images from MRI scans of brain tumors such as: glioma, meningioma and pituitary. Finally, with the synthesized images for each class they perform a comparison of classification models to determine the impact of adding these images to the training set. The networks used are VGG-16, InceptionResNetV2, and ResNet152V2. The accuracy percentages obtained were better with the use of VGG-16 ranging from 88.28% to 94.24% depending on the percentage of synthetic images added to the training set. Another more advanced derivation of the GANs is proposed by Chong *et al.* [67] who propose a variation of GAN customized by the authors. First, they used a Super-Resolution GAN to learn the variations of the brain according to the progression of Alzheimer's disease. Second, the generator and discriminator are trained using a Wasserstein GAN with Gradient Penalty (WGAN-GP). Finally, a pix2pix GAN conditional is applied for texture and contrast recognition. With these adjustments the authors were able to condition the GAN models to learn the characteristics necessary for Alzheimer detection.

Chapter 4

Methodology

4.1 ADNI dataset

The dataset used for the present study is Alzheimer's Disease Neuroimaging Initiative (ADNI) [68]. This is a project that focuses on the study of Alzheimer disease and other related neurological disorders. The primary goal of ADNI is to advance the understanding of Alzheimer disease through the collection, analysis, and distribution of neuroimaging data, biomarkers, and clinical data from study participants. The main objective of ADNI is to provide a wide range of data, including magnetic resonance imaging (MRI), positron emission tomography (PET) images, cerebrospinal fluid analysis, genetic and clinical data, from a diverse group of participants, including people with Alzheimer disease (AD), people with mild cognitive impairment (MCI), and cognitive normal (CN). The ADNI dataset includes participants recruited at various research sites in the United States and Canada. Participants are assessed regularly over time, allowing longitudinal monitoring of their cognitive health status. Access to the data in this repository is open to researchers with prior authorization from the authors. All the information corresponding to the study can be found in the Image and Data Archive (IDA) portal[1] which is part of the University of Southern Carolina and the Stevens Neuroimaging and Informatics Institute [2]. For more details, Figure 4.1 shows a batch of the ADNI dataset images used in this work. Each row represents a stage of the disease, and the order is as follows: the first row represents images for the CN class, the second row represents images for the MCI class, and the last

---

[1] `https://ida.loni.usc.edu/login.jsp?project=ADNI`

[2] `https://www.ini.usc.edu/`

row represents images for the AD class.

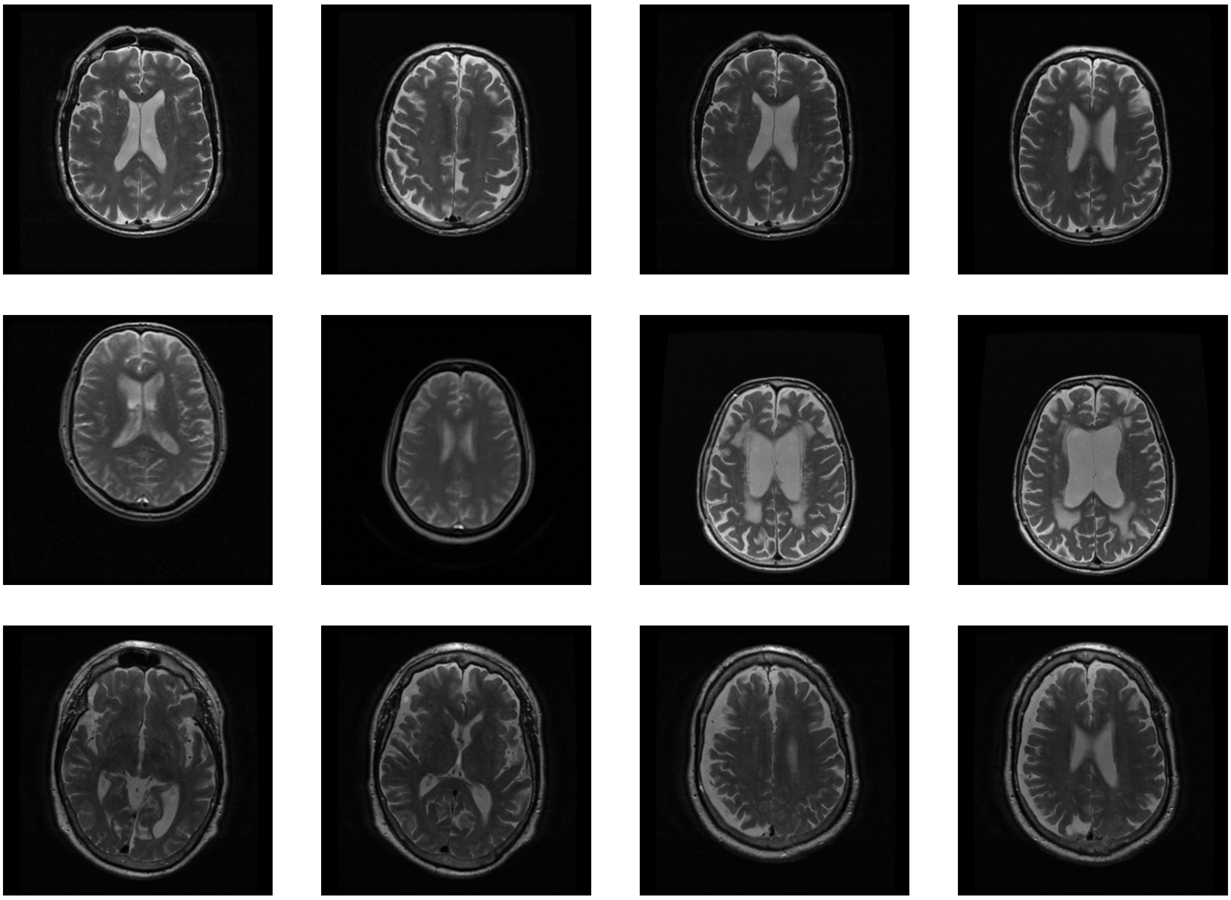


Figure 4.1: Examples of ADNI dataset. First row: Cognitive normal images. Second row: Mild cognitive impairment images. Third row: Alzheimer disease images.

4.2 Dataset preparation

The process of collecting images from the ADNI dataset was composed of some phases. These phases include acquiring the images from the IDA portal, converting them to png format, and finally cleaning the data obtained. First of all, the ADNI search repository contains several criteria to perform an advanced image search. The filters used to obtain the images from the repository were the following: project: ADNI; research group: AD, MCI or CN; image modality: MRI; acquisition plane: axial; acquisition type: 2D; weighting: T2; image type: original. This last filter means that the images were not subjected to any pre-processing or post-processing phase. Table 4.1 shows the characteristics used for the acquisition of the portal images. Additionally, appendix .1 shows how filters are applied in the advanced search section of the portal. In general, all these filters help to select the necessary information for downloading the images. This way, the type of images being downloaded is controlled and we can focus on the specific characteristics of each image.

Table 4.1: Search criteria and selected features for the acquisition of images from the ADNI dataset

Search Criteria	**Selected Feature**
Selected study	ADNI
Search criteria	ADNI Portal
Subject: Research group	AD, MCI or CN
Image: Modality	MRI
Image protocol: Acquisition plane	Axial
Acquisition type	2D
Weighting	T2
Image type	Original

The images are acquired in DICOM format since they contain the complete slices that comprise an MRI image. As mentioned previously, DICOM format images contain each slice that generates an MRI modality image. The MRI are typically made up of several slices that represent sections of the anatomical area being studied. Each of these slices is an individual image showing a section of the body in a specific direction. However, some of these slices may be blurry or contain information from parts of the brain not relevant to the study. All these images of unnecessary or blurry parts were eliminated to avoid inducing errors both in the generation of synthetic images and their subsequent classification. It is important to mention that the images were converted from DICOM format to PNG to be able to carry out this cleaning phase. Figures 4.2, 4.3, 4.4 represent examples of this type of images that were eliminated by each class. The selection of these filters was made according to the reviewed documentation and the concepts explained in Section 2.9.

Subsequently, the following number of images were collected per class: 1584 for AD, 1770 for MCI, and 1952 for CN. All these images represent the total slice for each DICOM image downloaded in its original form from the ADNI repository. Finally, after the part of manually cleaning the data mentioned, a total of the following images by class were obtained: 127 for AD, 491 for MCI, and 251 for CN. The proportion before and after this pre-processing stage is detailed in Figure 4.5 and Table 4.2.

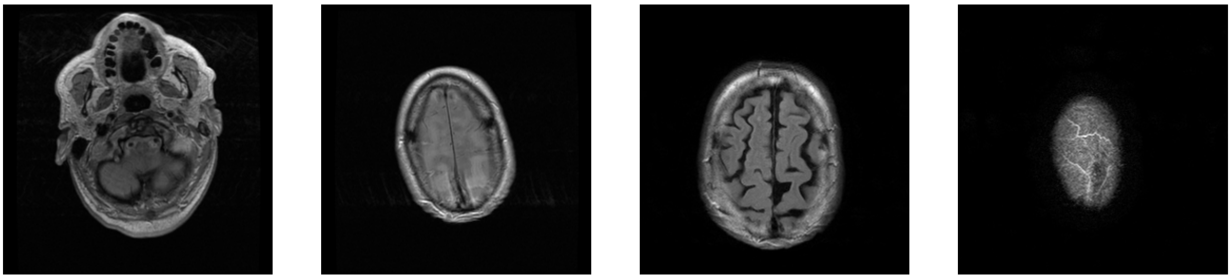


Figure 4.2: Examples of blurred images from ADNI dataset for the Cognitive Normal class.

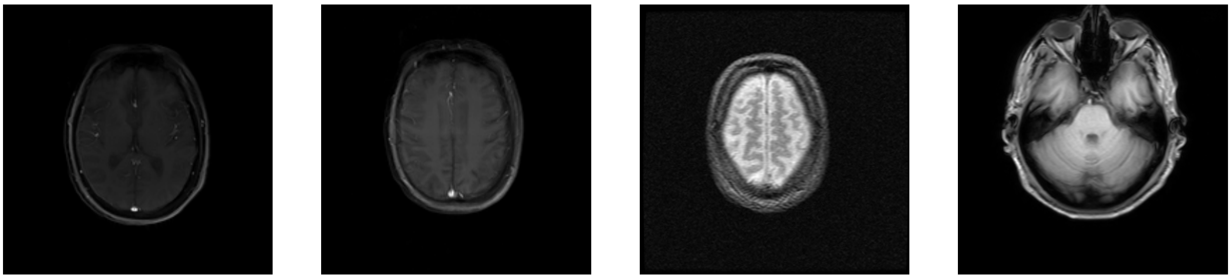


Figure 4.3: Examples of blurred images from ADNI dataset for the Mild Cognitive Impairment class.

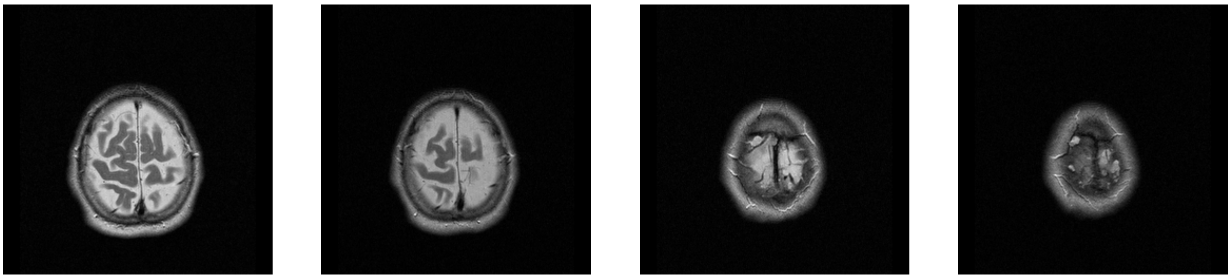


Figure 4.4: Examples of blurred images from ADNI dataset for the Alzheimer Disease class.

Table 4.2: Proportion of images before and after processing part.

	Before	*After*
Cognitive Normal (CN)	1952	251
Mild Cognitive Impairment (MCI)	1770	491
Alzheimer Disease (AD)	1584	127

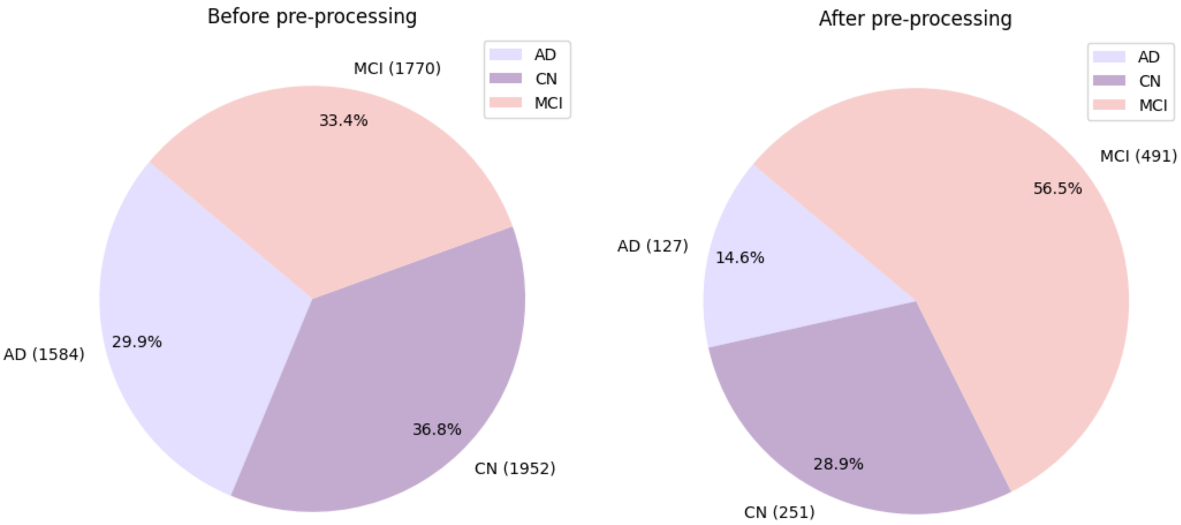


Figure 4.5: Proportion of the dataset for each class before and after the cleaning phase.

4.2.1 Dataset splitting

In order to perform the corresponding training for both the DCGAN training and the final application of the classification models, the following data division was followed. This corresponds to the training, validation and test percentages:

- Training: 80%.
- Validation: 10%.
- Test: 10%.

4.3 System design

4.3.1 Data distribution with t-SNE

One of the most important aspect to work with images is to understand the distribution of the images according to the subcategories contained in the set. It is important to understand that each class or category has its own characteristics and that these must be sufficiently independent to guarantee their division. With this in mind, the t-SNE algorithm was used to evaluate the distribution of these characteristics. This is an algorithm that is based on the deployment of a visual interface that allows visualizing the different classes that belong to the set of images in the form of clusters. Each cluster represents a different class of the set and they are grouped at a considerable distance. This means that each cluster keeps its own characteristics and therefore the representation of all of them is different.

The first step consisted of analyzing the original image set of the ADNI dataset. This step is fundamental to understand how the features of the original images behave. It also works as a method of verifying how diverse the classes are within the original dataset. This allows us to make sure that the three classes belonging to the ADNI dataset have independent characteristics that are respected between each set of images. Subsequently, the analysis is repeated but augmenting the synthetic images generated by DCGAN for the Alzheimer Disease class. The original images are mixed with the synthetic images and the t-SNE algorithm is executed to evaluate the distribution of the new set of images. This step is essential to evaluate that the synthetic images generated by DCGAN respect the original features of their class. This is visually checked if the synthetic images are grouped in the same cluster corresponding to their corresponding class. In other words, a cluster containing both the original images for the AD class and the synthetic images generated for the AD class should be displayed. Both steps of the implementation of the t-SNE algorithm are shown in Figure 4.6.

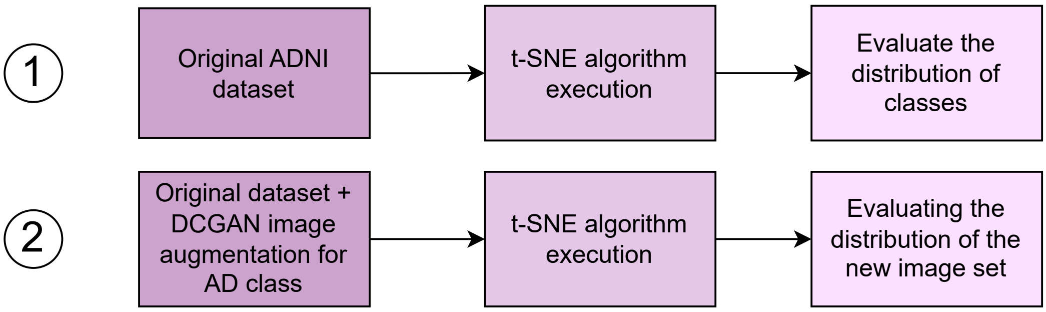

1
Original ADNI dataset
t-SNE algorithm execution
Evaluate the distribution of classes
2
Original dataset + DCGAN image augmentation for AD class
t-SNE algorithm execution
Evaluating the distribution of the new image set

Figure 4.6: Representation of both phases of implementation of the t-SNE algorithm. The first step is to evaluate the original image dataset. The second step is to evaluate the increase of synthetic images for the AD class.

The sklearn library was used to implement this algorithm. The hyperparameters set for the execution of the function were the following: number of components = 2, random state = 42, perplexity = 30 and number of iterations = 1000. Additionally, for a second test on the data distribution, all the above hyperparameters were maintained except the number of components which was set to 3 in order to visualize the data in 3D. A detailed summary of all hyperparameters used is given in Table 4.3.

Table 4.3: Hyperparameters used for the implementation of the t-SNE algorithm with the sklearn library.

Hyperparameters	**Value**
Perplexity	30
Random state	42
Number of components	2 or 3
Number of iterations	1000

4.3.2 Synthetic image generation by DCGAN

This is one of the most crucial stages of the experiment. It consists of generating synthetic images with DCGAN for the class of the original dataset with the lowest amount of data. After evaluating the original dataset, we concluded that the class with the least amount of data is Alzheimer disease (AD). With this in mind, this was the class that was selected to train the DCGAN and subsequently balance the dataset. In order to train the DCGAN, several hyperparameters were considered in order to adjust and obtain favorable results. These favorable results include having quality images, variability and maintaining the original characteristics of their class. The specific hyperparameters used for DCGAN training are listed in Table 4.4. However, the hyperparameters that had the greatest impact on achieving favorable results were batch size, learning rate and number of epochs. These hyperparameters helped to a higher measure to stabilize the DCGAN training and to obtain images that were adapted to the original characteristics of the dataset for the AD class. These hyperparameters were constantly modified until the desired configuration was found. Each of the steps involved in the training of the proposed DCGAN model are detailed in Figure 4.7.

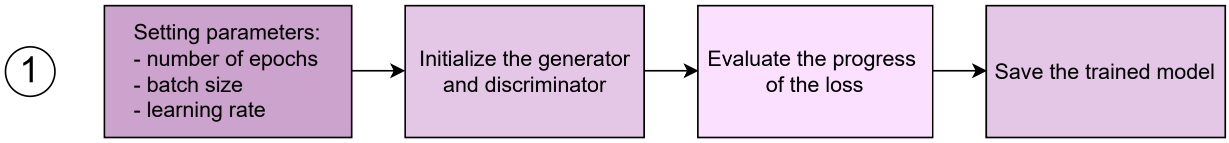


Figure 4.7: Representation of each of the steps followed for the DCGAN training. This includes the selection and tuning of hyperparameters until the generation of the final model.

Table 4.4: Hyperparameters used for the design and training of the DCGAN for ADNI dataset.

Hyperparameters	**Value**
Learning rate	0.0002
Number of epochs	500
Batch size	16
Latent vector	100
Image size	64
Feature maps (generator and discriminator)	64
Optimizer	Adam
Beta1 for optimizer	0.5
Workers	2
Number of channels	3

Subsequently, we proceeded to check, using the t-SNE algorithm, if the generated synthetic images are closely related to the other images belonging to the AD class of the original dataset. For this purpose, 100 synthesized images were added to the original set of images and plotted using the t-SNE algorithm. This verifies that the cluster initially created with the original data is maintained once the synthetic data is augmented. In addition, we verify that the synthesized data are placed in the same cluster of their corresponding class. This provides an immediate check on the quality of the images according to the corresponding class. Once this check is obtained, the trained DCGAN model is saved. This is done in order to later use this model to generate more synthetic images without the need to retrain the model and saving the configurations that were verified to work through the t-SNE algorithm.

On the other hand, it is important to verify that each time our DCGAN model is run for the AD class the data is plotted in the corresponding cluster with the t-SNE algorithm. In other words, each time we obtain a new set of synthetic images they must preserve the characteristics of the original data. To verify this point, 5 different batches of synthetic images were run for the AD class with the DCGAN model pre-trained in the previous step. Subsequently, each of the 5 batches of synthetic images was plotted with the t-SNE algorithm. This was done as a method to check the reliability of our DCGAN model to generate synthetic images for the AD class.

4.3.3 Classification models

An important part of the presented study is the usefulness that can be given to the synthetic images generated by DCGAN. For that reason, the performance of different classification models is evaluated for both the original data and the synthetic data generated by DCGAN. These two steps are performed to compare the performance of both models and to conclude if there are differences with the augmentation of synthetic data. Hence, the first step of this section consists in evaluating the classification models presented in Section 2.5 with the original data set coming from the ADNI dataset under the splitting mentioned in Section 4.2.1. Once these data have been recorded, we proceed to repeat the experiment but with the original data augmented with the DCGAN synthetic data for the AD class. Figure 4.8 details each of the steps used for the two experiments mentioned in this section.

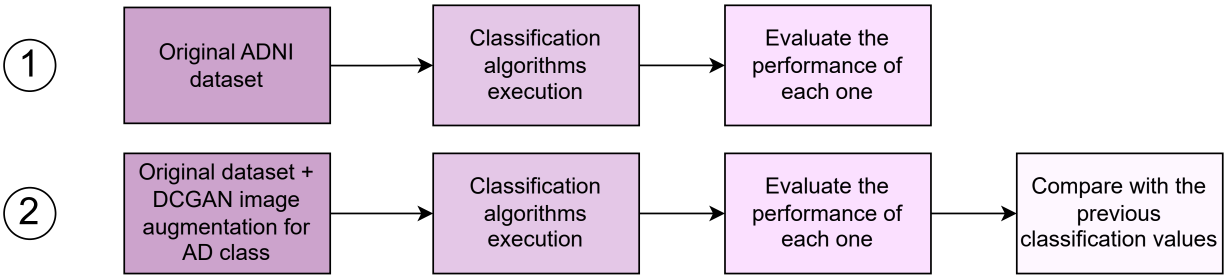

①
Original ADNI dataset
Classification algorithms execution
Evaluate the performance of each one
②
Original dataset + DCGAN image augmentation for AD class
Classification algorithms execution
Evaluate the performance of each one
Compare with the previous classification values

Figure 4.8: Representation of the stages of the classification process for each phase of the work. The first step is to evaluate the performance of the original data. The second step is to evaluate the performance of the synthesized data generated by DCGAN and compare with the previous one.

Additionally, for each of the four architectures used as classification models, various hyperparameters were used for training. The hyperparameters used for the four architectures and both experiments using original images and synthetic images from the DCGAN are presented in Table 4.5.

4.4 Evaluation metrics

4.4.1 Metrics for classification models

The evaluation of classification models is a fundamental aspect of model development and performance measurement. The effectiveness of these models must be rigorously evaluated

Table 4.5: Hyperparameters used for the different classification models: VGG-16, ResNet-18, AlexNet and InceptionV3.

Hyperparameters	**Value**
Number of classes	3 (AD, MCI, and CN)
Batch size	8
Number of epochs	10
Optimizer	Adagrad
Learning rate	0.0001

to ensure their usefulness and reliability in practice. For this reason, the following metrics are used:

1. Accuracy: This metric measures the proportion of correct predictions made by the model compared to the total number of examples in the test data set.

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{4.1}$$

2. Precision: This refers to the ratio of true positive predictions (true positives) to the total number of positive predictions made by the model.

$$precision = \frac{TP}{TP + FP} \tag{4.2}$$

3. Recall: It represents the proportion of positive samples correctly identified by the model.

$$recall = \frac{TP}{TP + FN} \tag{4.3}$$

4. F1-Score: This metric is useful when seeking a balance between the model's ability to correctly predict positive examples (precision) and its ability to correctly identify all positive examples in the data set (recall). In addition, F1-score is a useful metric when there is an imbalance between classes or when it is important to avoid both false positives and false negatives.

$$F1\,Score = 2\,\cdot\,\frac{Precision\,\cdot\,Recall}{Precision\,+\,Recall} \tag{4.4}$$

5. Confusion matrix: This tool is used to evaluate the performance of a classification model. This matrix shows the relationship between the predictions made by the model and the actual classes of a data set. The data it shows are as follows: True positives (TP), these are the cases in which the model correctly predicted a positive class; false positives (FP), these are the cases in which the model incorrectly predicted a positive class when it was actually negative; true negatives (TN), these are the cases in which the model correctly predicted a negative class; and finally false negatives (FN), which are the cases in which the model incorrectly predicted a negative class when it was actually positive. In addition, it is used to calculate various model performance evaluation metrics, such as accuracy, recall, precision and f1-score. These metrics provide detailed information on how the model is classifying the different classes presented. A detailed representation of what a common confusion matrix looks like is shown in Figure 4.9.

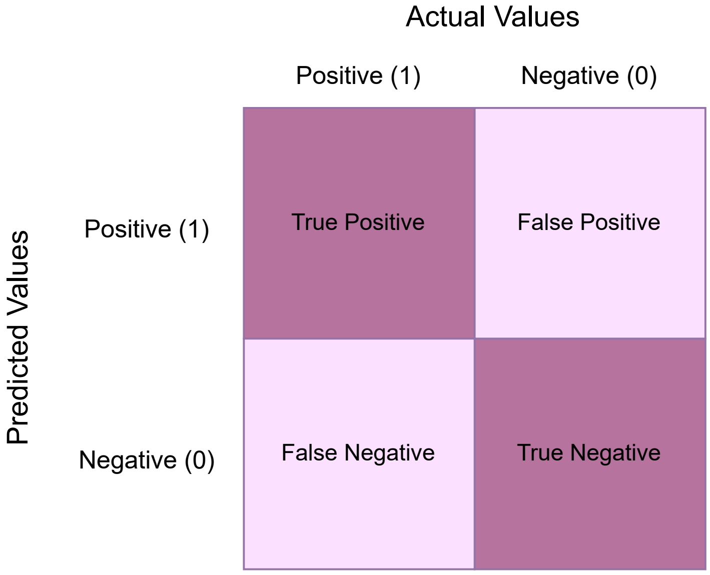


Figure 4.9: Representation of a confusion matrix. It includes the results of the classification, considering True Positive (TP),True Negative (TN), False Positive (FP) and False Negative (FN) respectively [7].

Chapter 5

Results and Discussion

5.1 Phase 1: Generation of synthetic images with DCGAN and evaluation of their quality and variability

5.1.1 ADNI dataset data distribution with t-SNE

One of the first questions we had before performing the DCGAN training was about the distribution of the data for the ADNI dataset. With this in mind, the t-SNE algorithm was implemented on the original ADNI dataset data to evaluate the distribution of characteristics by class. The idea is to check that the 3 classes that compose the ADNI dataset (Cognitive Normal (CN), Mild Cognitive Impairment (MCI) and Alzheimer Disease (AD)) have independent characteristics and to avoid overlapping between the DCGAN training and the original images. The algorithm was executed with the parameters specified in Section 4.3.1. The results of the distribution are shown in Figure 5.1. The graph shows three clusters divided by colors which represent each of the classes that constitute the ADNI dataset. In addition, it can be seen how the AD class painted green has less representation in terms of quantity inside the ADNI dataset. With this distribution it can be guaranteed that there is no overlapping of features in the original dataset. Likewise, this distribution graph is the reference point for the following comparisons to be made with the synthetic data generated by DCGAN.

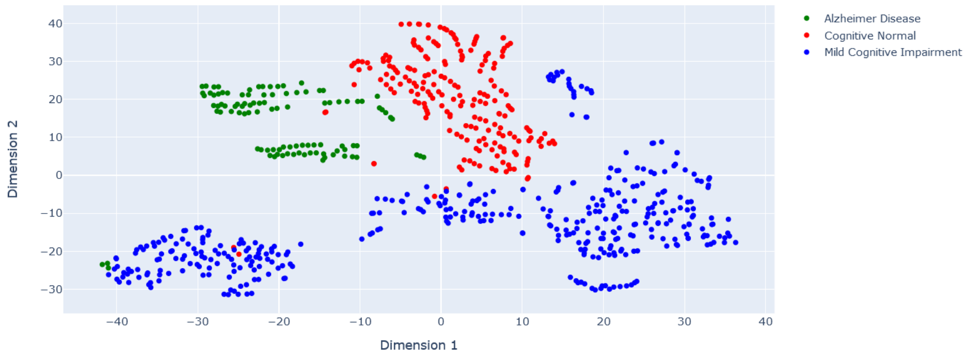


Figure 5.1: Result of the t-SNE algorithm for the original ADNI dataset images.

5.1.2 DCGAN training step

The next step is the training of the DCGAN to generate the synthetic images for the AD class of the ADNI dataset. This class was selected because of its smaller representation in the original dataset. Thus, by generating synthetic images for this particular class, the original dataset can be balanced. In order to generate the synthetic images, the training of the proposed DCGAN is performed with the hyperparameters detailed in the Section 4.3.2. This configuration of parameters allowed to achieve a more stable training according to the nature of the GANs. For the evaluation of the DCGAN training, see Figure 5.2 which details the loss curve for the generator and discriminator model. The same sample until epoch 250 approximately a very large loss for the generating model. This is due to the difficulty that the model has at the beginning to generalize and learn the information from the set of training images. A desired training of the DCGAN consists of reducing the loss of the generator as much as possible and increasing the loss of the discriminator. In the case of the proposed training, both conditions are met according to the training data. This serves as a first method of checking the quality of the DCGAN training and the quality of the images obtained by this model. However, evaluating the training loss for the generator and discriminator model is not enough to conclude that the images obtained are related to the training images. For this reason, the use of the t-SNE algorithm is again implemented as a method of checking the quality of the generated synthetic images.

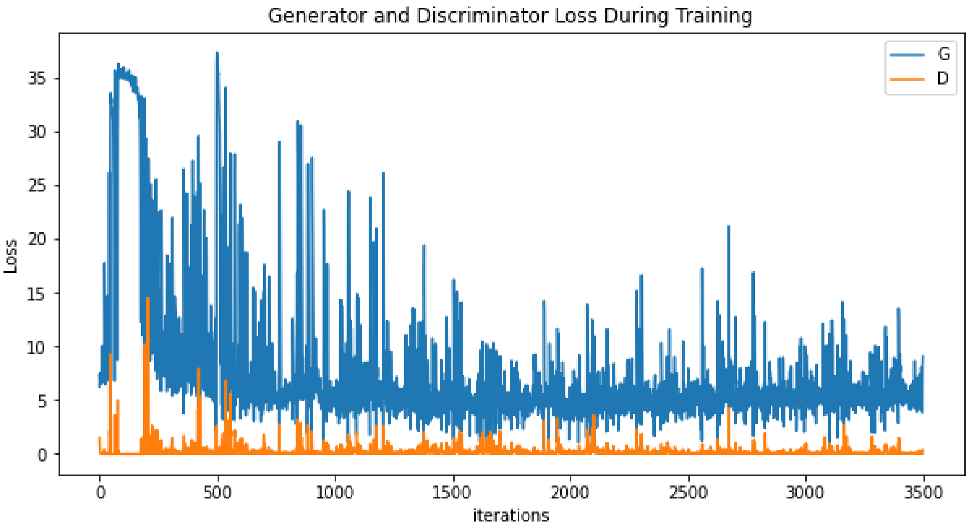


Figure 5.2: DCGAN loss evolution during training for the generator and discriminator models. Blue line (G) corresponds to the loss of the generator. Orange line (D) corresponds to the loss of the discriminator.

5.1.3 Distribution of the synthetic images from DCGAN with t-SNE

One of the most difficult aspects when working with adversary networks is evaluating the quality of the images obtained. In this case, the same concern arises since it is a fundamental step to guarantee the employability of the images in cases of real applications. For this reason, the use of the t-SNE algorithm is implemented again to evaluate the quality of the synthetic images obtained. To do this, the synthetic data is added to the set of original images of the ADNI dataset. In other words, the images produced by the DCGAN for the AD class are added to the set of original images of the ADNI dataset. Subsequently, the t-SNE algorithm is applied to evaluate to which of the three initially obtained clusters the synthetic images are positioned. For this evaluation, five batches of different images produced by our trained DCGAN model were executed. Each batch contains 100 different images for each one with the purpose of having diversity of data for evaluation with the t-SNE algorithm. The five cases are presented in Figure 5.3 to Figure 5.7 where are shown in detail the distribution of the new data obtained from the DCGAN. The data corresponding to the AD class is marked in green. It can easily be visualized how the original data of the AD class of the ADNI dataset augmented with the synthetic data of the AD class of the DCGAN are grouped in the same cluster. Furthermore, a certain homogeneity can be observed for the other two clusters corresponding to the CN and MCI classes respectively.

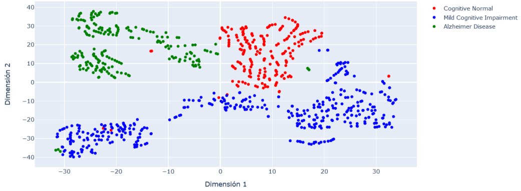


Figure 5.3: Result of t-SNE for the first batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset.

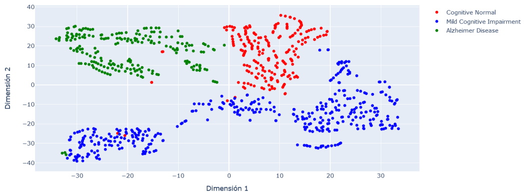


Figure 5.4: Result of t-SNE for the second batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset.

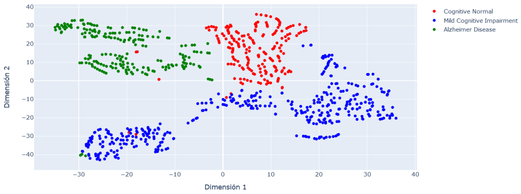


Figure 5.5: Result of t-SNE for the third batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset.

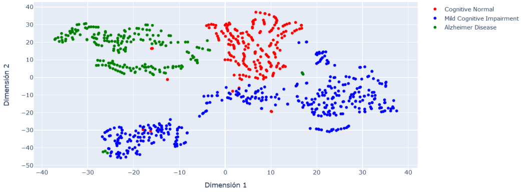


Figure 5.6: Result of t-SNE for the fourth batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset.

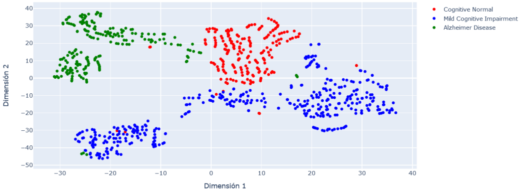


Figure 5.7: Result of t-SNE for the fifth batch of synthetic images of the DCGAN pre-trained model in the AD class of the ADNI dataset.

Moreover, it is important to mention that these first results were obtained for the implementation of the t-SNE algorithm in two dimensions. However, as mentioned in Section 2.3, this algorithm can be implemented up to three-dimensional representations. Thus, it was decided to implement one of the aforementioned image bumps but using a three-dimensional representation of the t-SNE algorithm. With this, a second verification can be made that there is no overlap between the clusters represented by the algorithm. In other words, the characteristics for each class of images are independent of each other. The results for the three-dimensional implementation are shown in Figure 5.8 for a front view and in Figure 5.9 for a top view. With this last representation it can be more accurately visualized that the synthetic images managed to adapt to the corresponding cluster of the AD class.

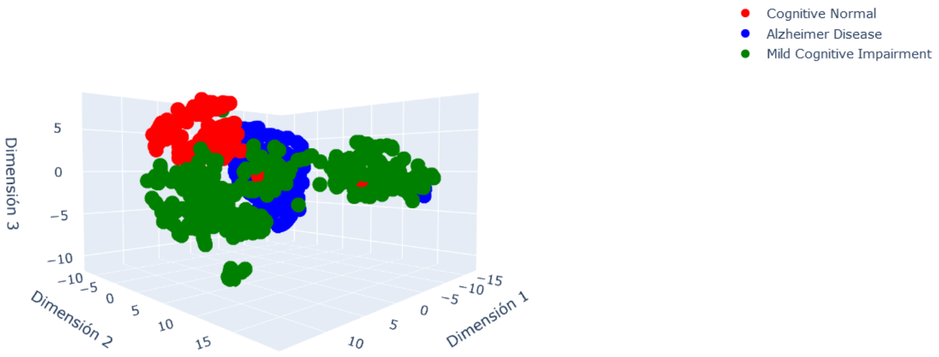


Figure 5.8: Front view of the t-SNE plot for synthetic images of the AD class of the ADNI dataset.

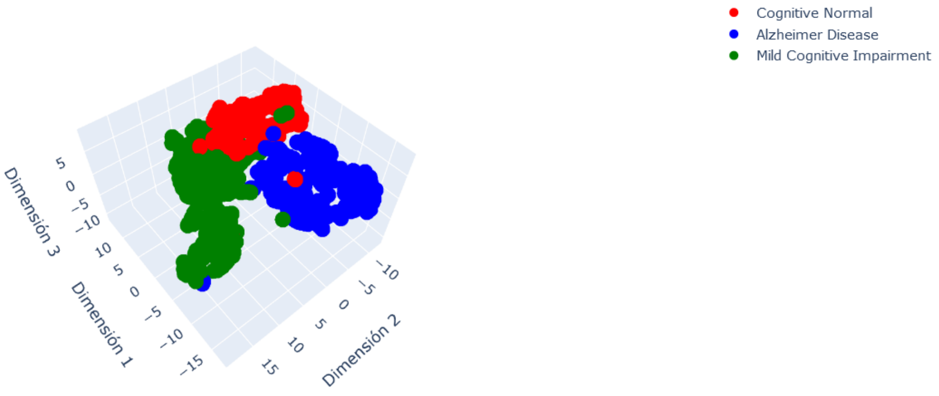


Figure 5.9: Top view of the t-SNE plot for synthetic images of the AD class of the ADNI dataset.

Finally, to make a comparison on data that do not meet the class independence criterion the Figures 5.10 5.11 are presented. This evaluates a set of images from Kaggle [1]

---

[1] https://www.kaggle.com/datasets/tourist55/alzheimers-dataset-4-class-of-images?resource=download

corresponding to images of different stages of Alzheimer's disease. In the images it can be clearly seen an overlap of the four classes that form the dataset. With these preliminary results of the t-SNE algorithm, independence of characteristics between classes can not be guaranteed. Therefore, these images are a perfect example of a set that is not useful for training a generative adversarial network. This test was done to have a contrast between the favorable results obtained previously and what the visualization of a case of failure would be like.

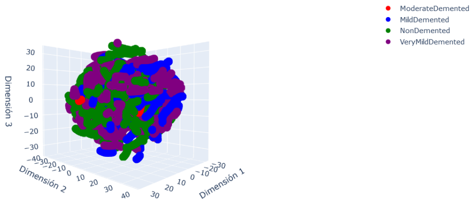


Figure 5.10: Front view of the t-SNE plot for original images for the four classes of the Alzheimer's Dataset from Kaggle.

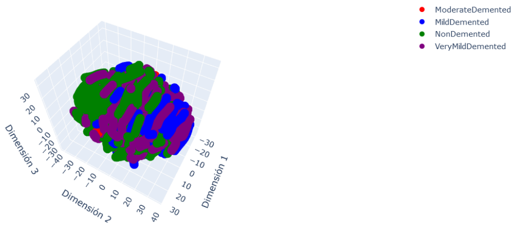


Figure 5.11: Top view of the t-SNE plot for original images for the four classes of the Alzheimer's Dataset from Kaggle.

Finally, from the DCGAN model obtained through training, various sets of synthetic images can be generated for the AD class. Thus, in Figure 5.12 are shown two batches of images: the first corresponds to the real images of the AD class from the ADNI dataset. The second batch corresponds to the synthetic imagenes generated by the DCGAN for the AD class.

5.2 Phase 2: Improvement of the classification algorithm with the augmentation of DCGAN synthetic images of the Alzheimer Disease class

In this section, the results corresponding to the various classification algorithms implemented to compare the performance with the increase of synthetic images generated by the DCGAN are presented. By augmenting the dataset with these DCGAN generated images, we seek to address the challenges associated with the limited availability of real-world data and improve the accuracy of classification algorithms. The dataset was randomly divided into training, validation, and test sets according to the percentages presented in

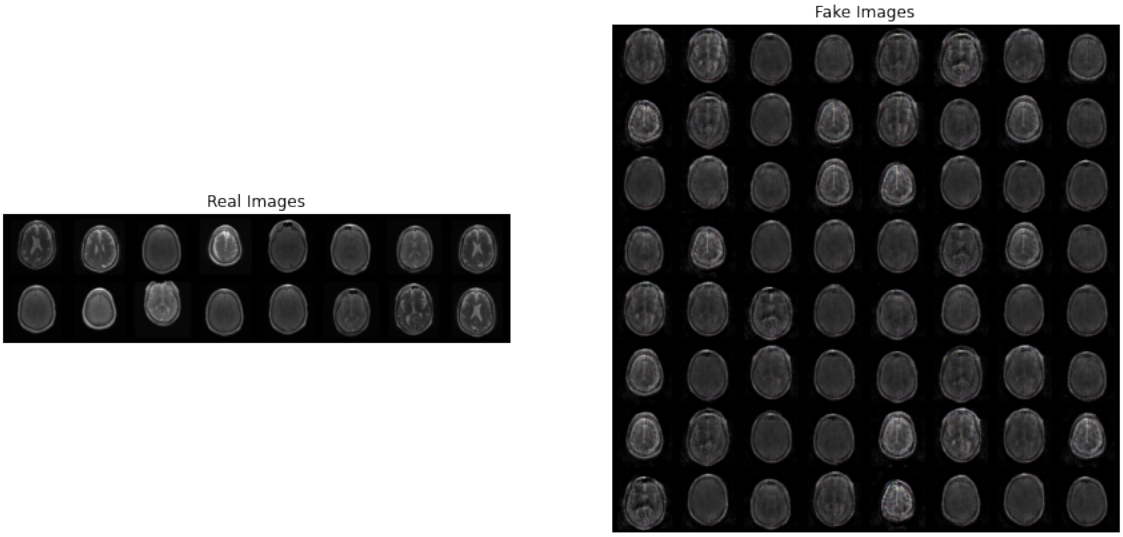


Figure 5.12: Lot of original images and synthetic images generated by DCGAN. Left side: original images from the ADNI dataset. Right side: synthetic images generated by DCGAN.

Section 4.2.1.

For each of the examples developed, a cross entropy was used as a loss function. In addition, the use of the Adagrad optimizer with a learning rate value of 0.0001 was considered for each example. The models were trained with 10 epochs and a batch size of 8 according to the limited quantity of data available into the original dataset. The results for the accuracy and loss evolution into the train dataset can be found in Figures [5.13-5.20]. From this set of graphs it can be deduced that the best results were obtained with the InceptionV3 model according to the learning and loss curves. However, the difference presented for the training set is minimal for the cases of the VGG-16 and AlexNet models respectively.

In the case of the validation set, the results for the accuracy and loss evolution into the validation dataset can be found in Figures [5.21-5.28]. For this case, better results can be considered for the InceptionV3 and AlexNet classification models. In the case of the accuracy computation, an improvement in the performance of each model is shown in Figure 5.21 for the InceptionV3 model and Figure 5.27 for the AlexNet model. In the case of the ResNet-18 and VGG-16 models, the results were very unstable for the presented validation set. This is true for both the accuracy and loss analysis for both models.

However, it is important to analyze the results obtained on the test dataset to verify the

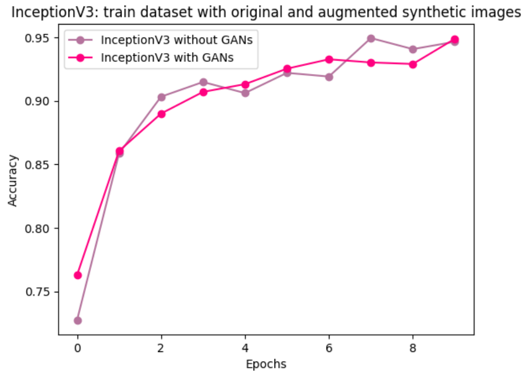


Figure 5.13: InceptionV3 accuracy on train dataset.

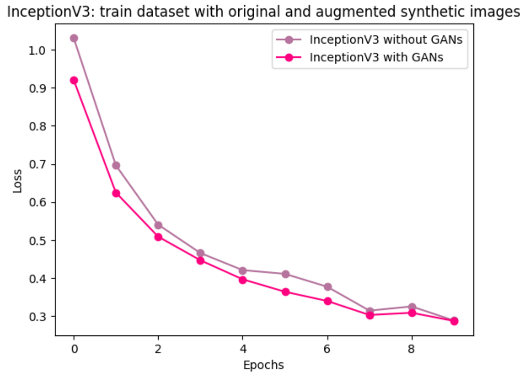


Figure 5.14: InceptionV3 loss on train dataset.

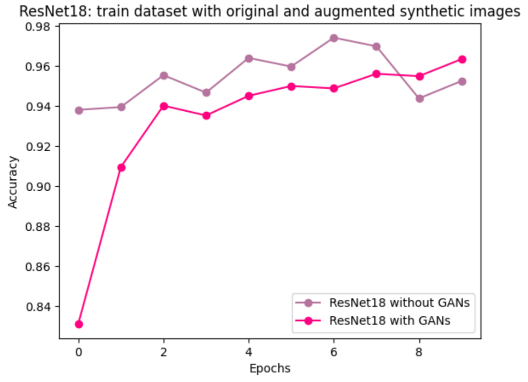


Figure 5.15: ResNet-18 accuracy on train dataset.

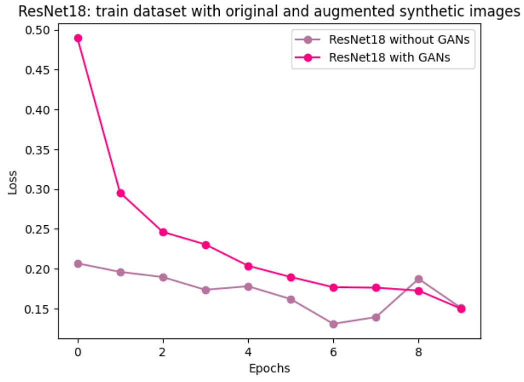


Figure 5.16: ResNet-18 loss on train dataset.

performance of each model. The images implemented for the original dataset are 869 (101 for AD class, 200 for CN and 392 for MCI) and the synthetic augmented images are 100 for the AD class. With this increase in the number of synthetic images, a more balanced dataset between classes was achieved, with a total of 969 images (200 for AD class, 200 for CN, and 392 for MCI) to perform the experiments. Finally, the accuracy results are presented in Table 5.1.

Table 5.1: Comparison of the results of the different classification models with both the original images and the augmented synthetic images for the AD class.

	Accuracy (%)	
Models	Original images	Original images + synthetic images from DCGAN
InceptionV3	94.25	**95.40**
ResNet-18	95.40	94.25
VGG-16	98.85	97.70
AlexNet	91.95	**95.40**

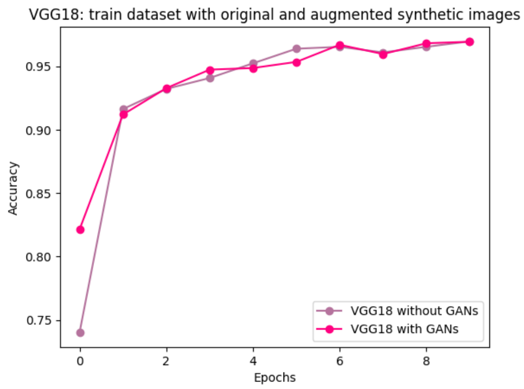


Figure 5.17: VGG-16 accuracy on train dataset.

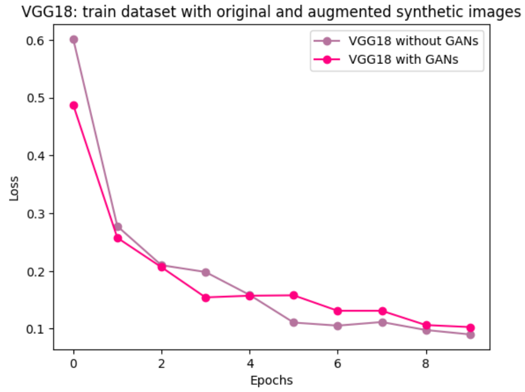


Figure 5.18: VGG-16 loss on train dataset.

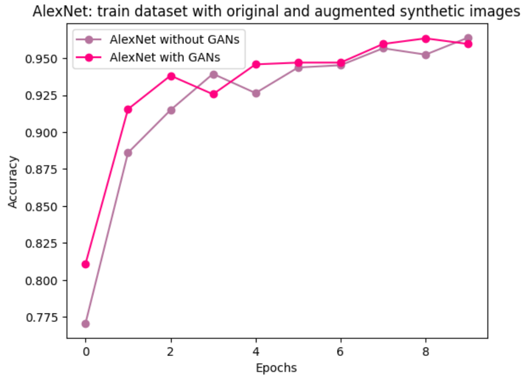


Figure 5.19: AlexNet accuracy on train dataset.

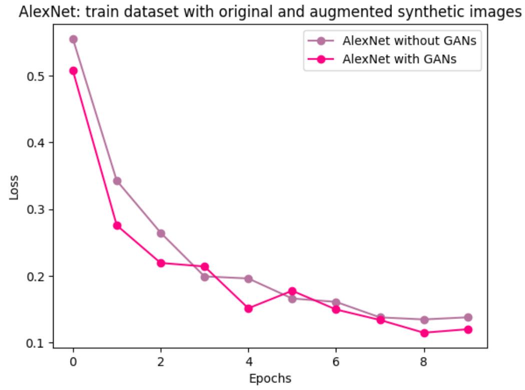


Figure 5.20: AlexNet loss on train dataset.

The InceptionV3 and AlexNet models have been shown to perform better compared to other classification models, and this is largely due to their significant increase in test set accuracy. It can be seen that there was an increase in the percentage of accuracy for the InceptionV3 and AlexNet models of 95.40% respectively. It is also important to mention that the model that shows the greatest increase in the percentage of accuracy is AlexNet. Taking as a starting point these two models as the ones that obtained an increase in the percentage of accuracy with the synthetic images argument, a deeper analysis of the classification performance by class can be made.

The analysis of the confusion matrices in the context of classification allows us to evaluate the performance of the models for each class. Therefore, the confusion matrices are evaluated in two different scenarios: first, only using the images of the ADNI dataset and second, by augmenting synthetic images for the AD class. The first case to be evaluated

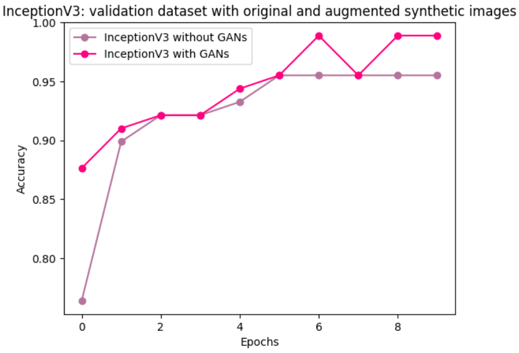


Figure 5.21: InceptionV3 accuracy on validation dataset.

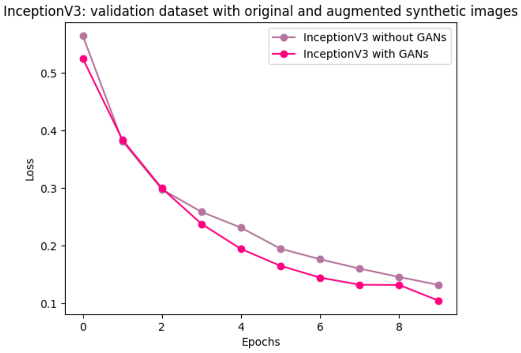


Figure 5.22: InceptionV3 loss on validation dataset.

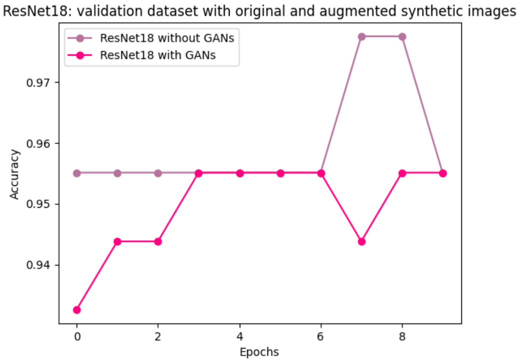


Figure 5.23: ResNet-18 accuracy on validation dataset.

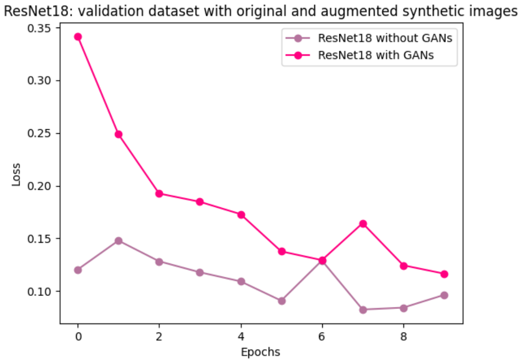


Figure 5.24: ResNet-18 loss on validation dataset.

is the InceptionV3 model. The following results were obtained from Figure 5.29, which corresponds to the use of the original images from the ADNI dataset. On the other hand, Figure 5.30 is the result of augmenting synthetic images from DCGAN for the AD class. In this case an increase in the values correctly classified for the Alzheimer disease and Cognitive normal classes can be noted. It is also important to mention that there is not too great an alteration or deterioration for the remaining class.

These matrices allow obtaining the point values of True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). In this way, additional metrics such as accuracy, recall and f1-score can be evaluated for each of the classes and the two planned scenarios (without synthetic images versus with increased synthetic images). The additional metrics corresponding to the InceptionV3 model only with original images from the ADNI dataset are presented in Table 5.2. Additional metrics corresponding to the scenario with synthetic image augmentation for the AD class are presented in Table 5.3. In the

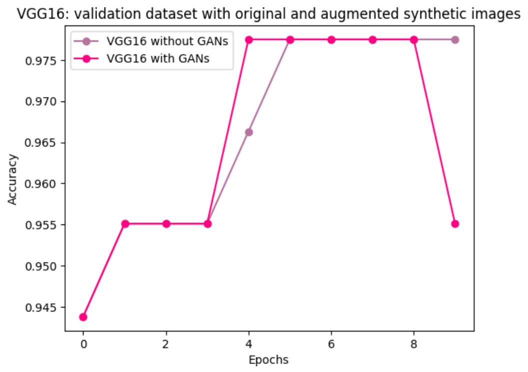


Figure 5.25: VGG-16 accuracy on validation dataset.

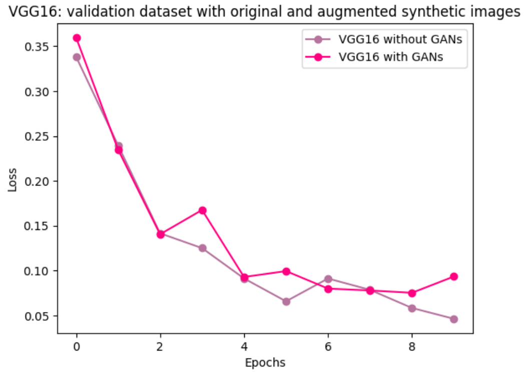


Figure 5.26: VGG-16 loss on validation dataset.

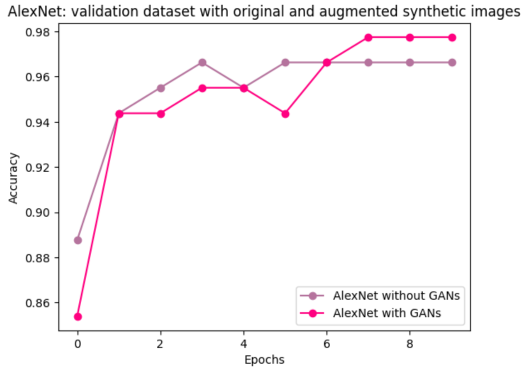


Figure 5.27: AlexNet accuracy on validation dataset.

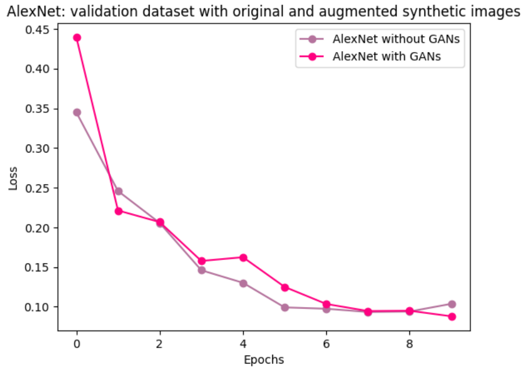


Figure 5.28: AlexNet loss on validation dataset.

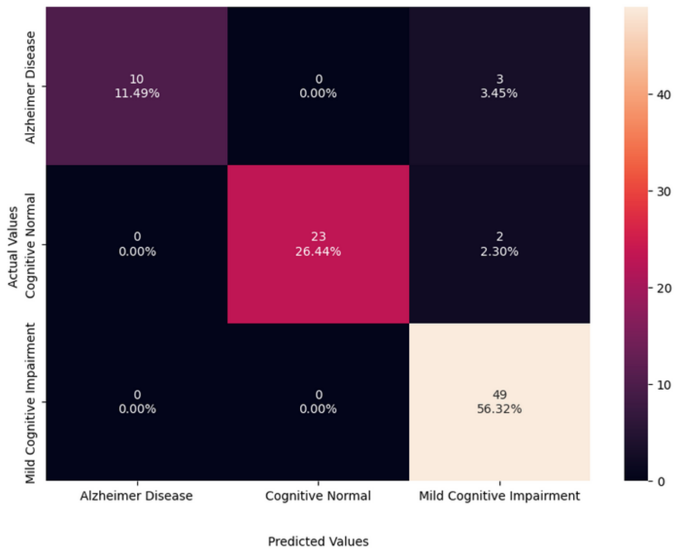


Figure 5.29: Confusion matrix on the test set for the InceptionV3 model using original images from the ADNI dataset.

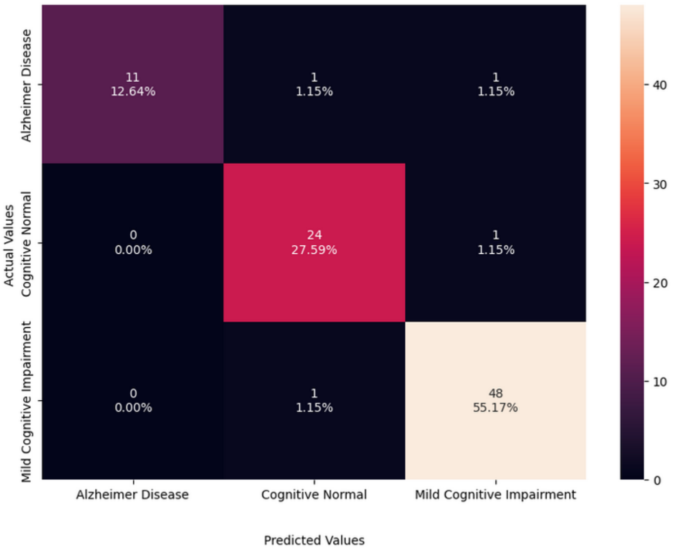


Figure 5.30: Confusion matrix on the test set for the InceptionV3 model using DCGAN synthetic image augmentation for the AD class.

second data set (with synthetic images) for the Alzheimer disease class, the recall increased from 0.77 to 0.85. This indicates that the model with GANs is better at identifying most of the real Alzheimer's cases. The F1-Score also increased from 0.87 to 0.92, so the model has a better balance between precision and recall in this class.

Table 5.2: Additional metrics for the InceptionV3 model with original images from the ADNI dataset.

	Precision	Recall	F1-Score
Alzheimer Disease	1.00	0.77	0.87
Cognitive Normal	1.00	0.92	0.96
Mild Cognitive Impairment	0.91	1.00	0.95

Table 5.3: Additional metrics for the InceptionV3 model with the augmented synthetic images for the AD class.

	Precision	Recall	F1-Score
Alzheimer Disease	1.00	**0.85**	**0.92**
Cognitive Normal	0.92	**0.96**	0.94
Mild Cognitive Impairment	**0.96**	0.98	**0.97**

Similarly, the second case to be evaluated is the AlexNet model. The Figure 5.31 shows the confusion matrix that corresponds to the use of original data from the ADNI dataset. In addition, the results of the confusion matrix for the synthetic image augmentation for the AD class are shown in Figure 5.32. An increase in the number of correctly classified cases of Alzheimer disease can be observed. In the same way, in the case of the Mild Cognitive Impairment class an increase in the number of cases classified as correct can also be observed. As in the previous model, it is also important to mention that there is not too great an alteration or deterioration for the remaining class.

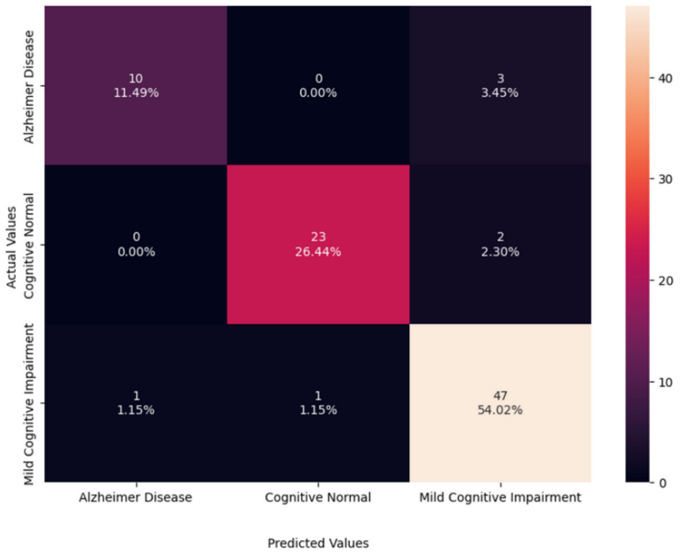


Figure 5.31: Confusion matrix on the test set for the AlexNet model using original images from the ADNI dataset.

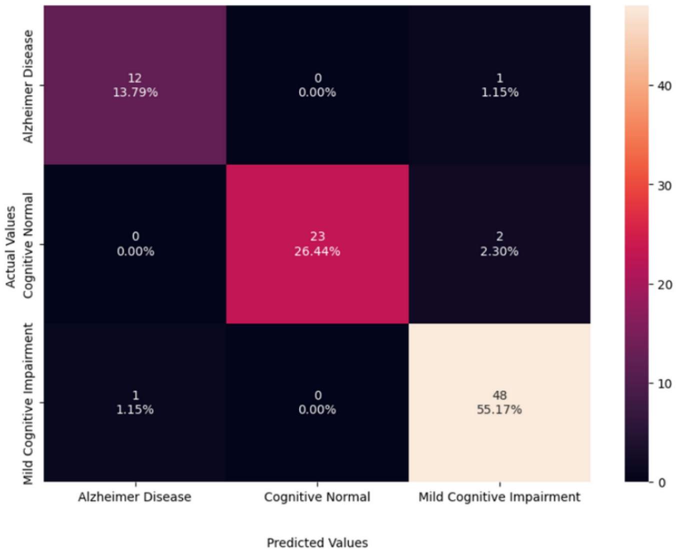


Figure 5.32: Confusion matrix on the test set for the AlexNet model using DCGAN synthetic image augmentation for the AD class.

Moreover, for this case it is also possible to evaluate additional metrics of precision, recall and f1-score in both scenarios for the AlexNet model. As a result, Table 5.4 shows the additional metrics using only the original data, while Table 5.5 shows the results with the augmentation of synthetic images for the AD class. In the second data set (with synthetic images) for the Alzheimer disease class, the precision has improved slightly, which means that the model is doing a better job of predicting cases of Alzheimer disease. Also, according to the increase of the F1-Score value a better balance between precision and recall metrics can be achieved.

Table 5.4: Additional metrics for the AlexNet model with original images from the ADNI dataset.

	Precision	Recall	F1-Score
Alzheimer Disease	0.91	0.77	0.83
Cognitive Normal	0.96	0.92	0.94
Mild Cognitive Impairment	0.90	0.96	0.93

Table 5.5: Additional metrics for the AlexNet model with the augmented synthetic images for the AD class.

	Precision	Recall	F1-Score
Alzheimer Disease	**0.92**	**0.92**	**0.92**
Cognitive Normal	**1.00**	0.92	**0.96**
Mild Cognitive Impairment	**0.94**	**0.98**	**0.96**

5.3 Hardware

For both phases in order to perform the training of the DCGAN and classification tasks, a NVIDIA A100 SXM4 of 40gb GPU was used on CEDIA HPC [2] with a total of 16 CPU cores and 32MB of total memory to allocate.

---

[2] `https://cedia.edu.ec/en/benefit/supercomputer/`

Chapter 6

Conclusions

In the present work, we proposed an alternative to generate synthetic images for Alzheimer disease from MRI scans using generative adversarial networks (GANs), specifically using a Deep Convolutional Generative Adversarial Network (DCGAN). The synthetic images generated and subsequently introduced to the original batch corresponded only to the Alzheimer disease class. Subsequently, it was proposed to evaluate the impact of introducing these synthetic images for classification tasks using a VGG-16, ResNet-18, InceptionV3 and AlexNet respectively. The three stages to be classified were: cognitive normal, mild cognitive impairment and Alzheimer disease. The initial objective was to help improve the classification of the diverse stages of Alzheimer's disease through class balancing by introducing synthetic images. In the initial stage, we began by consulting the literature to determine the relevant characteristics of Alzheimer disease images in their different stages. The different approaches used for both the generation of synthetic images with GANs and to determine the classification models used in the medical field were also consulted. Finally, two approaches were obtained, both for the generation of synthetic images and for the classification models, which were both able to contribute to the achievement of the initial objective of the work.

First, different configurations were tested to achieve a stable training of the DCGAN model and generate quality synthetic images for the AD class. For this purpose, parameters such as bath size, learning rate and number of epochs were modified. The initial method of checking the correct training performance of the DCGAN model was done by visualising the loss curves for the generator and discriminator correspondingly. Once this model had been obtained, different batches of synthetic images were generated and used in the t-SNE algorithm. This was useful to make sure that the model found was able to produce synthetic images as similar as possible to the original images. In other words, this means that the features of the original images could be effectively captured in the DCGAN training process. Subsequently, these features were transferred in the synthetic image generation process. As a result of this verification, different batches of synthetic images could be produced to be added to the original image sets in order to evaluate their impact on the performance of the proposed classification models.

On the other hand, the initial hypothesis of the present work was to improve classification tasks using synthetic images. As a consequence, the results obtained show that there was a significant improvement in two of the classification algorithms used. First, for the Inception V3 model, an increase of about 1.10% of the overall accuracy of the model was reported. Regarding its confusion matrix, a better classification was achieved for the Alzheimer disease (AD) and cognitive normal (CN) class. Second, for the AlexNet model, an increase of about 3.40% of the overall accuracy of the model was reported. Looking in detail at the confusion matrix, a better classification was achieved for the AD and MCI class. In the case of the VGG-16 and ResNet-18 models, no improvement in the classification task was obtained. On the contrary, there was an deterioration in both algorithms with the introduction of synthetic images.

It is important to note that there are several factors involved in the improvement of the classification algorithms. The nature of these algorithms as well as their structure may be one of the reasons why no improvement in the classification of these models was obtained. In addition, the hyperparameters of the algorithm may need specific adjustments to adapt to the new configuration with synthetic data. Parameters such as learning rate, model depth or regularisation may influence how the model exploits synthetic data. As a result, the model architecture may not be best suited to handle the information introduced by synthetic images.

It is also important to mention that in the field of medicine any improvement is significant, no matter how small it may seem. In our case, an improvement in the classification related to stages of Alzheimer disease helps to have more accurate diagnoses, which leads to more effective and targeted treatments for the gradual advancement of the disease. Different stages of Alzheimer disease may require specific therapeutic approaches, and improved classification tasks contributes to the implementation of more appropriate treatments for each patient. Finally, improved accuracy in diagnosis and classification can help optimise the use of medical resources, reduce costs associated with inappropriate treatments, and improve the quality of life of patients and their family members.

6.0.1 Future Work

One of the main tasks is to consider the inclusion of more training images to improve the diversity and representation of the data. This may help to reinforce the ability of the model to recognise patterns in different stages and variations of the disease. For the specific case of the Alzheimer disease, it is also important to extend the model to cover additional stages of the disease, such as Early Mild Cognitive Impairment (EMCI) and Late Mild Cognitive Impairment (LMCI). This could improve the model's ability to detect and distinguish specific patterns at various stages of the disease. It is even essential to explore the possibility of incorporating other types of medical imaging, such as Positron Emission Tomography (PET) imaging.

In terms of image generation, perform a comprehensive comparison with other Generative Adversarial Networks (GANs) models available in the literature. Evaluate the performance of the proposed model in terms of image generation quality and generalisation capability compared to existing alternatives. It is even important to consider exploring further optimisation of the hyperparameters of the model to improve its performance. This could include adjustments to the network architecture, learning rate, batch size, among others.

Bibliography

[1] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[4] M. Ferguson, R. Ak, Y.-T. T. Lee, and K. H. Law, "Automatic localization of casting defects with convolutional neural networks," in *2017 IEEE international conference on big data (big data)*. IEEE, 2017, pp. 1726–1735.

[5] L. N. Tanenbaum, A. J. Tsiouris, A. N. Johnson, T. P. Naidich, M. C. DeLano, E. R. Melhem, P. Quarterman, S. Parameswaran, A. Shankaranarayanan, M. Goyen *et al.*, "Synthetic mri for clinical neuroimaging: results of the magnetic resonance image compilation (magic) prospective, multicenter, multireader trial," *American journal of neuroradiology*, vol. 38, no. 6, pp. 1103–1110, 2017.

[6] P. S. Kumar and V. Dharun, "A study of mri segmentation methods in automatic brain tumor detection," *Int. J. Eng. Technol*, vol. 8, no. 2, pp. 609–614, 2016.

[7] S. Narkhede, "Understanding confusion matrix," *Towards Data Science*, vol. 180, no. 1, pp. 1–12, 2018.

[8] H. Ali, M. N. M. Salleh, R. Saedudin, K. Hussain, and M. F. Mushtaq, “Imbalance class problems in data mining: A review,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 3, pp. 1560–1571, 2019.

[9] D. Ramyachitra and P. Manikandan, “Imbalanced dataset classification and solutions: a review,” *International Journal of Computing and Business Research (IJCBR)*, vol. 5, no. 4, pp. 1–29, 2014.

[10] L. Wang, M. Han, X. Li, N. Zhang, and H. Cheng, “Review of classification methods on unbalanced data sets,” *IEEE Access*, vol. 9, pp. 64 606–64 628, 2021.

[11] M. M. Rahman and D. N. Davis, “Addressing the class imbalance problem in medical datasets,” *International Journal of Machine Learning and Computing*, vol. 3, no. 2, p. 224, 2013.

[12] L. Kunde, E. McMeniman, and M. Parker, “Clinical photography in dermatology: ethical and medico-legal considerations in the age of digital and smartphone technology,” *Australasian Journal of Dermatology*, vol. 54, no. 3, pp. 192–197, 2013.

[13] C. K. Lau, H. H. Schumacher, and M. S. Irwin, “Patients’ perception of medical photography,” *Journal of Plastic, Reconstructive & Aesthetic Surgery*, vol. 63, no. 6, pp. e507–e511, 2010.

[14] I. Berle, “The principles of ethical practice in professional clinical photography,” *Journal of Audiovisual Media in Medicine*, vol. 27, no. 1, pp. 11–13, 2004.

[15] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative adversarial networks: An overview,” *IEEE signal processing magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[16] A. Aggarwal, M. Mittal, and G. Battineni, “Generative adversarial network: An overview of theory and applications,” *International Journal of Information Management Data Insights*, vol. 1, no. 1, p. 100004, 2021.

[17] I. J. Goodfellow, “On distinguishability criteria for estimating generative models,” *arXiv preprint arXiv:1412.6515*, 2014.

[18] D. Nankani and R. D. Baruah, “Investigating deep convolution conditional gans for electrocardiogram generation,” in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–8.

[19] H. Ye, L. Liang, G. Y. Li, and B.-H. Juang, “Deep learning-based end-to-end wireless communication systems with conditional gans as unknown channels,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3133–3143, 2020.

[20] J. Luo, Y. Xu, C. Tang, and J. Lv, “Learning inverse mapping by autoencoder based generative adversarial nets,” in *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part II 24*. Springer, 2017, pp. 207–216.

[21] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.

[22] S. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, “Impact of fully connected layers on performance of convolutional neural networks for image classification,” *Neurocomputing*, vol. 378, pp. 112–119, 2020.

[23] S. Akbar, M. Peikari, S. Salama, S. Nofech-Mozes, and A. Martel, “Transitioning between convolutional and fully connected layers in neural networks,” in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*. Springer, 2017, pp. 143–150.

[24] D. A. Van Dyk and X.-L. Meng, “The art of data augmentation,” *Journal of Computational and Graphical Statistics*, vol. 10, no. 1, pp. 1–50, 2001.

[25] G. E. Hinton and S. Roweis, “Stochastic neighbor embedding,” *Advances in neural information processing systems*, vol. 15, 2002.

[26] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne.” *Journal of machine learning research*, vol. 9, no. 11, 2008.

[27] M. Wattenberg, F. Viégas, and I. Johnson, “How to use t-sne effectively,” *Distill*, vol. 1, no. 10, p. e2, 2016.

[28] G. C. Linderman and S. Steinerberger, “Clustering with t-sne, provably,” *SIAM Journal on Mathematics of Data Science*, vol. 1, no. 2, pp. 313–332, 2019.

[29] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[30] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization.” *Journal of machine learning research*, vol. 12, no. 7, 2011.

[31] H. Hippius and G. Neundörfer, “The discovery of alzheimer’s disease,” *Dialogues in clinical neuroscience*, 2022.

[32] M. M. Weber, “Aloys alzheimer, a coworker of emil kraepelin,” *Journal of psychiatric research*, vol. 31, no. 6, pp. 635–643, 1997.

[33] G. Cipriani, C. Dolciotti, L. Picchi, and U. Bonuccelli, “Alzheimer and his disease: a brief history,” *Neurological Sciences*, vol. 32, pp. 275–279, 2011.

[34] J. L. Cummings and G. Cole, “Alzheimer disease,” *Jama*, vol. 287, no. 18, pp. 2335–2338, 2002.

[35] H. Yang, H. Xu, Q. Li, Y. Jin, W. Jiang, J. Wang, Y. Wu, W. Li, C. Yang, X. Li *et al.*, “Study of brain morphology change in alzheimer’s disease and amnestic mild cognitive impairment compared with normal controls,” *General psychiatry*, vol. 32, no. 2, 2019.

[36] L. S. Hemmy, E. J. Linskens, P. C. Silverman, M. A. Miller, K. M. Talley, B. C. Taylor, J. M. Ouellette, N. L. Greer, T. J. Wilt, M. Butler *et al.*, “Brief cognitive tests for distinguishing clinical alzheimer-type dementia from mild cognitive impairment or normal cognition in older adults with suspected cognitive impairment: a systematic review,” *Annals of internal medicine*, vol. 172, no. 10, pp. 678–687, 2020.

[37] R. C. Petersen, “Mild cognitive impairment,” *CONTINUUM: Lifelong Learning in Neurology*, vol. 22, no. 2 Dementia, p. 404, 2016.

[38] A. M. Sanford, “Mild cognitive impairment,” *Clinics in geriatric medicine*, vol. 33, no. 3, pp. 325–337, 2017.

[39] S. Gauthier, B. Reisberg, M. Zaudig, R. C. Petersen, K. Ritchie, K. Broich, S. Belleville, H. Brodaty, D. Bennett, H. Chertkow *et al.*, “Mild cognitive impairment,” *The lancet*, vol. 367, no. 9518, pp. 1262–1270, 2006.

[40] W. van der Flier PhD, “Philip scheltens, bart de strooper, miia kivipelto, henne holstege, gael chételat, charlotte e teunissen, jeffrey cummings, wiesje m van der flier,” *Lancet*, vol. 397, pp. 1577–90, 2021.

[41] M. G. Ulep, S. K. Saraon, and S. McLea, “Alzheimer disease,” *The Journal for Nurse Practitioners*, vol. 14, no. 3, pp. 129–135, 2018.

[42] B. Dubois, N. Villain, G. B. Frisoni, G. D. Rabinovici, M. Sabbagh, S. Cappa, A. Bejanin, S. Bombois, S. Epelbaum, M. Teichmann *et al.*, “Clinical diagnosis of alzheimer’s disease: recommendations of the international working group,” *The Lancet Neurology*, vol. 20, no. 6, pp. 484–496, 2021.

[43] G. Katti, S. A. Ara, and A. Shireen, “Magnetic resonance imaging (mri)–a review,” *International journal of dental clinics*, vol. 3, no. 1, pp. 65–70, 2011.

[44] C. Westbrook and J. Talbot, *MRI in Practice.* John Wiley & Sons, 2018.

[45] E. G. Kehoe, J. P. McNulty, P. G. Mullins, and A. L. Bokde, “Advances in mri biomarkers for the diagnosis of alzheimer’s disease,” *Biomarkers in medicine*, vol. 8, no. 9, pp. 1151–1169, 2014.

[46] J. Appel, E. Potter, Q. Shen, G. Pantol, M. T. Greig, D. Loewenstein, and R. Duara, “A comparative analysis of structural brain mri in the diagnosis of alzheimer’s disease,” *Behavioural neurology*, vol. 21, no. 1-2, pp. 13–19, 2009.

[47] N. Yamanakkanavar, J. Y. Choi, and B. Lee, "Mri segmentation and classification of human brain using deep learning for diagnosis of alzheimer's disease: a survey," *Sensors*, vol. 20, no. 11, p. 3243, 2020.

[48] X. Tang, F. Cai, D.-X. Ding, L.-L. Zhang, X.-Y. Cai, and Q. Fang, "Magnetic resonance imaging relaxation time in alzheimer's disease," *Brain research bulletin*, vol. 140, pp. 176–189, 2018.

[49] S. Otake, T. Taoka, M. Maeda, and W. Yuh, "A guide to identification and selection of axial planes in magnetic resonance imaging of the brain. neuroradiol j 2018; 31: 336–44."

[50] R. Cuingnet, E. Gerardin, J. Tessieras, G. Auzias, S. Lehéricy, M.-O. Habert, M. Chupin, H. Benali, O. Colliot, A. D. N. Initiative *et al.*, "Automatic classification of patients with alzheimer's disease from structural mri: a comparison of ten methods using the adni database," *neuroimage*, vol. 56, no. 2, pp. 766–781, 2011.

[51] W. Yang, R. L. Lui, J.-H. Gao, T. F. Chan, S.-T. Yau, R. A. Sperling, and X. Huang, "Independent component analysis-based classification of alzheimer's disease mri data," *Journal of Alzheimer's disease*, vol. 24, no. 4, pp. 775–783, 2011.

[52] S. Campos, L. Pizarro, C. Valle, K. R. Gray, D. Rueckert, and H. Allende, "Evaluating imputation techniques for missing data in adni: a patient classification study," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 20th Iberoamerican Congress, CIARP 2015, Montevideo, Uruguay, November 9-12, 2015, Proceedings 20.* Springer, 2015, pp. 3–10.

[53] M. Shahbaz, S. Ali, A. Guergachi, A. Niazi, and A. Umer, "Classification of alzheimer's disease using machine learning techniques." in *Data*, 2019, pp. 296–303.

[54] S. Sarraf and G. Tofighi, "Classification of alzheimer's disease using fmri data and deep learning convolutional neural networks," *arXiv preprint arXiv:1603.08631*, 2016.

[55] S. Korolev, A. Safiullin, M. Belyaev, and Y. Dodonova, "Residual and plain convolutional neural networks for 3d brain mri classification," in *2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017).* IEEE, 2017, pp. 835–838.

[56] S. Basaia, F. Agosta, L. Wagner, E. Canu, G. Magnani, R. Santangelo, M. Filippi, A. D. N. Initiative *et al.*, "Automated classification of alzheimer's disease and mild cognitive impairment using a single mri and deep neural networks," *NeuroImage: Clinical*, vol. 21, p. 101645, 2019.

[57] D. Cheng and M. Liu, "Cnns based multi-modality classification for ad diagnosis," in *2017 10th international congress on image and signal processing, biomedical engineering and informatics (CISP-BMEI)*. IEEE, 2017, pp. 1–5.

[58] B.-K. Choi, N. Madusanka, H.-K. Choi, J.-H. So, C.-H. Kim, H.-G. Park, S. Bhattacharjee, and D. Prakash, "Convolutional neural network-based mr image analysis for alzheimer's disease classification," *Current Medical Imaging*, vol. 16, no. 1, pp. 27–35, 2020.

[59] R. Prajapati, U. Khatri, and G. R. Kwon, "An efficient deep neural network binary classifier for alzheimer's disease classification," in *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2021, pp. 231–234.

[60] B. C. Simon, D. Baskar, and V. Jayanthi, "Alzheimer's disease classification using deep convolutional neural network," in *2019 9th international conference on advances in computing and communication (ICACC)*. IEEE, 2019, pp. 204–208.

[61] F. Konidaris, T. Tagaris, M. Sdraka, and A. Stafylopatis, "Generative adversarial networks as an advanced data augmentation technique for mri data." in *VISIGRAPP (5: VISAPP)*, 2019, pp. 48–59.

[62] H.-C. Shin, N. A. Tenenholtz, J. K. Rogers, C. G. Schwarz, M. L. Senjem, J. L. Gunter, K. P. Andriole, and M. Michalski, "Medical image synthesis for data augmentation and anonymization using generative adversarial networks," in *Simulation and Synthesis in Medical Imaging: Third International Workshop, SASHIMI 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Proceedings 3*. Springer, 2018, pp. 1–11.

[63] C. F. Baumgartner, L. M. Koch, K. C. Tezcan, J. X. Ang, and E. Konukoglu, "Visual feature attribution using wasserstein gans," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8309–8319.

[64] M. Sajjad, F. Ramzan, M. U. G. Khan, A. Rehman, M. Kolivand, S. M. Fati, and S. A. Bahaj, "Deep convolutional generative adversarial network for alzheimer's disease classification using positron emission tomography (pet) and synthetic data augmentation," *Microscopy Research and Technique*, vol. 84, no. 12, pp. 3023–3034, 2021.

[65] J. Islam and Y. Zhang, "Gan-based synthetic brain pet image generation," *Brain informatics*, vol. 7, pp. 1–12, 2020.

[66] D. Mukherkjee, P. Saha, D. Kaplun, A. Sinitca, and R. Sarkar, "Brain tumor image generation using an aggregation of gan models with style transfer," *Scientific Reports*, vol. 12, no. 1, p. 9141, 2022.

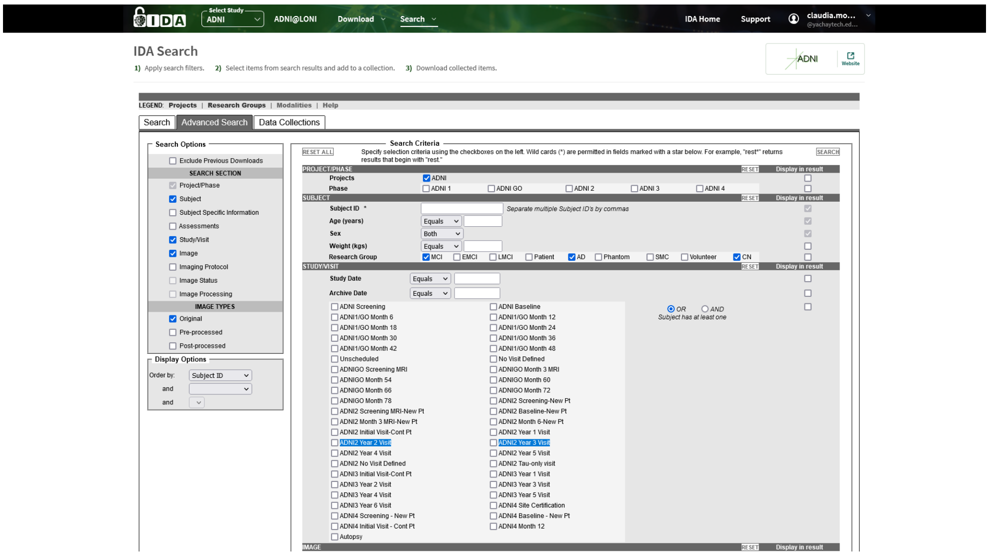
[67] C. K. Chong and E. T. W. Ho, "Synthesis of 3d mri brain images with shape and texture generative adversarial deep neural networks," *IEEE Access*, vol. 9, pp. 64 747–64 760, 2021.

[68] ADNI. (2005) Alzheimer's Disease Neuroimaging Initiative (ADNI). [Online]. Available: https://adni.loni.usc.edu/data-samples/access-data/

Appendices

.1 Appendix 1 - Advanced Search for ADNI dataset.

This appendix shows in detail the search filters available in the Image and Data Archive (IDA) portal [1] for downloading data from the ADNI dataset. From these filters it is possible to control the type of data obtained from the original dataset. From these filters you can create a collection of images and download them later. You can access Open Use studies with an IDA account, and can apply to access other studies.



IDA Search

1) Apply search filters. 2) Select items from search results and add to a collection. 3) Download collected items.

Search Options

☐ Exclude Previous Downloads

SEARCH SECTION

☑ Project/Phase
☑ Subject
☐ Subject Specific Information
☐ Assessments
☑ Study/Visit
☑ Image
☐ Imaging Protocol
☐ Image Status
☐ Image Processing

IMAGE TYPES

☑ Original
☐ Pre-processed
☐ Post-processed

Display Options

Order by: Subject ID
and
and

Search Criteria

Specify selection criteria using the checkboxes on the left. Wild cards (*) are permitted in fields marked with a star below. For example, "rest*" returns results that begin with "rest."

PROJECT/PHASE

Projects: ☑ ADNI
Phase: ☐ ADNI 1 ☐ ADNI GO ☐ ADNI 2 ☐ ADNI 3 ☐ ADNI 4

SUBJECT

Subject ID *: Separate multiple Subject ID's by commas
Age (years): Equals
Sex: Both
Weight (kgs): Equals
Research Group: ☑ MCI ☐ EMCI ☐ LMCI ☐ Patient ☑ AD ☐ Phantom ☐ SMC ☐ Volunteer ☑ CN

STUDY/VISIT

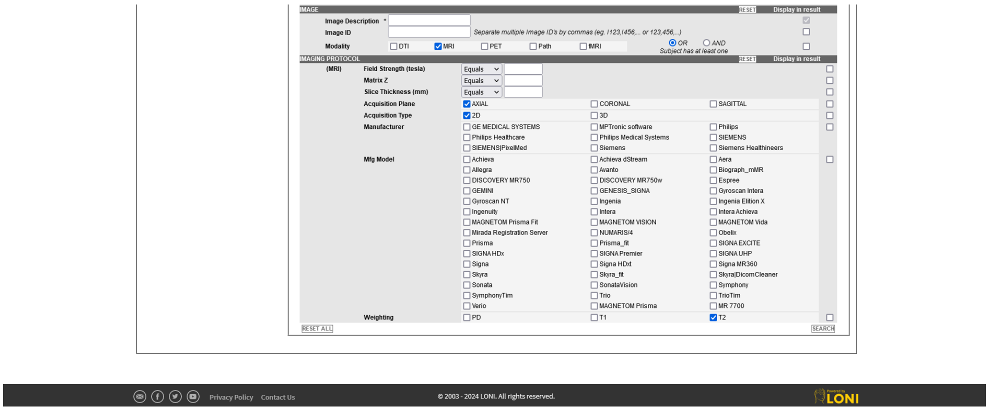
Study Date: Equals
Archive Date: Equals

☐ ADNI Screening ☐ ADNI Baseline
☐ ADNI1/GO Month 6 ☐ ADNI1/GO Month 12
☐ ADNI1/GO Month 18 ☐ ADNI1/GO Month 24
☐ ADNI1/GO Month 30 ☐ ADNI1/GO Month 36
☐ ADNI1/GO Month 42 ☐ ADNI1/GO Month 48
☐ Unscheduled ☐ No Visit Defined
☐ ADNIGO Screening MRI ☐ ADNIGO Month 3 MRI
☐ ADNIGO Month 54 ☐ ADNIGO Month 60
☐ ADNIGO Month 66 ☐ ADNIGO Month 72
☐ ADNIGO Month 78 ☐ ADNI2 Screening-New Pt
☐ ADNI2 Screening MRI-New Pt ☐ ADNI2 Baseline-New Pt
☐ ADNI2 Month 3 MRI-New Pt ☐ ADNI2 Month 6-New Pt
☐ ADNI2 Initial Visit-Cont Pt ☐ ADNI2 Year 1 Visit
☐ ADNI2 Year 2 Visit ☐ ADNI2 Year 3 Visit
☐ ADNI2 Year 4 Visit ☐ ADNI2 Year 5 Visit
☐ ADNI2 No Visit Defined ☐ ADNI2 Tau-only visit
☐ ADNI3 Initial Visit-Cont Pt ☐ ADNI3 Year 1 Visit
☐ ADNI3 Year 2 Visit ☐ ADNI3 Year 3 Visit
☐ ADNI3 Year 4 Visit ☐ ADNI3 Year 5 Visit
☐ ADNI3 Year 6 Visit ☐ ADNI4 Site Certification
☐ ADNI4 Screening - New Pt ☐ ADNI4 Baseline - New Pt
☐ ADNI4 Initial Visit - Cont Pt ☐ ADNI4 Month 12
☐ Autopsy

● OR ○ AND
Subject has at least one

IMAGE

Figure 1: First part of the advanced search section for the ADNI dataset. It contains the following: project phase, research group (CN, MCI, or AD), and type of image (Original).



IMAGE

Image Description *
Image ID: Separate multiple Image ID's by commas (eg. I123,I456,... or 123,456,...)
Modality: ☐ DTI ☑ MRI ☐ PET ☐ Path ☐ fMRI
● OR ○ AND
Subject has at least one

IMAGING PROTOCOL

(MRI)
Field Strength (tesla): Equals
Matrix Z: Equals
Slice Thickness (mm): Equals
Acquisition Plane: ☑ AXIAL ☐ CORONAL ☐ SAGITTAL
Acquisition Type: ☑ 2D ☐ 3D
Manufacturer:
☐ GE MEDICAL SYSTEMS ☐ MPTronic software ☐ Philips
☐ Philips Healthcare ☐ Philips Medical Systems ☐ SIEMENS
☐ SIEMENS|PixelMed ☐ Siemens ☐ Siemens Healthineers
Mfg Model:
☐ Achieva ☐ Achieva dStream ☐ Aera
☐ Allegra ☐ Avanto ☐ Biograph_mMR
☐ DISCOVERY MR750 ☐ DISCOVERY MR750w ☐ Espree
☐ GEMINI ☐ GENESIS_SIGNA ☐ Gyroscan Intera
☐ Gyroscan NT ☐ Ingenia ☐ Ingenia Elition X
☐ Ingenuity ☐ Intera ☐ Intera Achieva
☐ MAGNETOM Prisma Fit ☐ MAGNETOM VISION ☐ MAGNETOM Vida
☐ Mirada Registration Server ☐ NUMARIS/4 ☐ Obelix
☐ Prisma ☐ Prisma_fit ☐ SIGNA EXCITE
☐ SIGNA HDx ☐ SIGNA Premier ☐ SIGNA UHP
☐ Signa ☐ Signa HDxt ☐ Signa MR360
☐ Skyra ☐ Skyra_fit ☐ Skyra|DicomCleaner
☐ Sonata ☐ SonataVision ☐ Symphony
☐ SymphonyTim ☐ Trio ☐ TrioTim
☐ Verio ☐ MAGNETOM Prisma ☐ MR 7700
Weighting: ☐ PD ☐ T1 ☑ T2

© 2003 - 2024 LONI. All rights reserved.

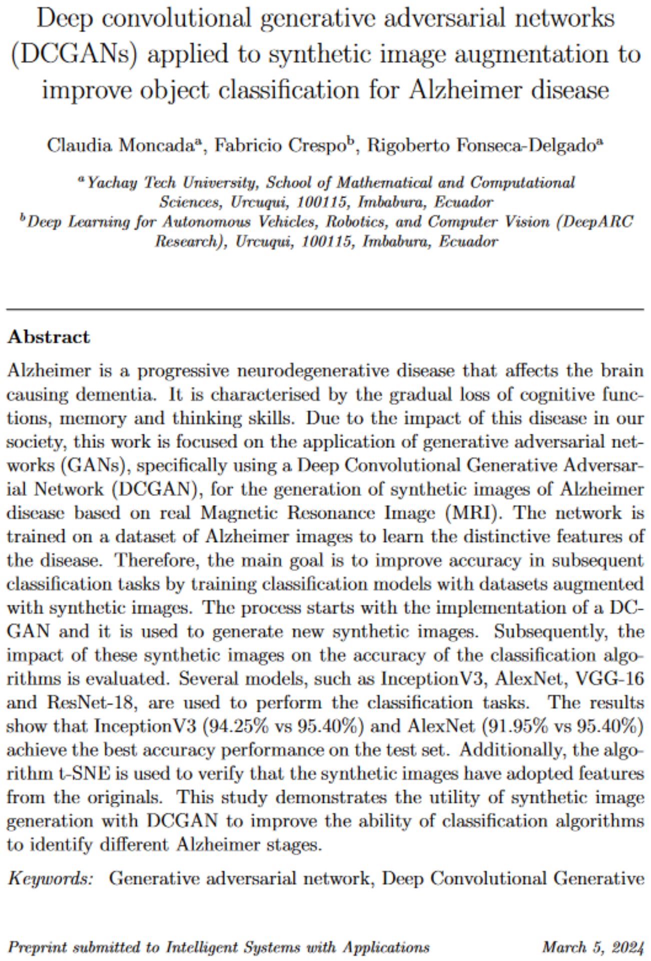
Figure 2: Second part of the advanced search section for the ADNI dataset. It contains the following: image modality (MRI), acquisition plane (Axial), acquisition type (2D) and weighting (T2).

---

[1] `https://ida.loni.usc.edu/login.jsp?project=ADNI`

.2 Appendix 2 - List of Papers

Based on the results obtained for this work, it was decided to write a scientific article. Due to the nature of the results and the methodology used, it was referred to the Journal Intelligent Systems with Applications. The Figure 3 shows the first page of the article submitted for review.



Deep convolutional generative adversarial networks (DCGANs) applied to synthetic image augmentation to improve object classification for Alzheimer disease

Claudia Moncada[a], Fabricio Crespo[b], Rigoberto Fonseca-Delgado[a]

[a] *Yachay Tech University, School of Mathematical and Computational Sciences, Urcuqui, 100115, Imbabura, Ecuador*
[b] *Deep Learning for Autonomous Vehicles, Robotics, and Computer Vision (DeepARC Research), Urcuqui, 100115, Imbabura, Ecuador*

Abstract

Alzheimer is a progressive neurodegenerative disease that affects the brain causing dementia. It is characterised by the gradual loss of cognitive functions, memory and thinking skills. Due to the impact of this disease in our society, this work is focused on the application of generative adversarial networks (GANs), specifically using a Deep Convolutional Generative Adversarial Network (DCGAN), for the generation of synthetic images of Alzheimer disease based on real Magnetic Resonance Image (MRI). The network is trained on a dataset of Alzheimer images to learn the distinctive features of the disease. Therefore, the main goal is to improve accuracy in subsequent classification tasks by training classification models with datasets augmented with synthetic images. The process starts with the implementation of a DCGAN and it is used to generate new synthetic images. Subsequently, the impact of these synthetic images on the accuracy of the classification algorithms is evaluated. Several models, such as InceptionV3, AlexNet, VGG-16 and ResNet-18, are used to perform the classification tasks. The results show that InceptionV3 (94.25% vs 95.40%) and AlexNet (91.95% vs 95.40%) achieve the best accuracy performance on the test set. Additionally, the algorithm t-SNE is used to verify that the synthetic images have adopted features from the originals. This study demonstrates the utility of synthetic image generation with DCGAN to improve the ability of classification algorithms to identify different Alzheimer stages.

Keywords: Generative adversarial network, Deep Convolutional Generative

Preprint submitted to Intelligent Systems with Applications *March 5, 2024*

Figure 3: First page of the paper submitted to Intelligent System with Applications Journal.