IJERT
ISSN : 2278 - 0181

THESIS

Thesis ID : IJERTTH0025

Addressing data scarcity and class imbalance in Alzheimer's

Luke Chugh

Northumbria University, Newcastle

Published By

International Journal of
Engineering Research and Technology
(www.ijert.org)

Northumbria
University
NEWCASTLE

Dissertation

KF7029
MSc Computer Science
and Digital Technologies Project

Student Name: Luke Chugh

Student ID: W21054410

Supervisor Name: Dr. Ossama Alshabrawy

Second Supervisor Name: Dr. Yilun Shang

MSc Programme: Data Science

Dissertation Title: Addressing data scarcity and class imbalance in Alzheimer's

Month and Year of Submission: 04/23

Declaration

I declare the following:

(1) That the material contained in this dissertation is the end result of my own work and that due acknowledgement has been given in the bibliography and references to **ALL** sources be they printed, electronic or personal.

(2) The Word Count (university allows 10% extra number of words) of this Dissertation is: ***13,142***

(3) That unless this dissertation has been confirmed as confidential, I agree to sections of the dissertation being placed on the eLearning Portal (Blackboard), if deemed appropriate, to allow future students the opportunity to see examples of past dissertations. I understand that if displayed on the eLearning Portal it would be made available for no longer than five years and that students would be able to print off copies or download.

(4) I agree to my dissertation being submitted to a plagiarism detection service, where it will be stored in a database and compared against work submitted from this or any other Department or from other institutions using the service.

In the event of the service detecting a high degree of similarity between content within the service this will be reported back to my supervisor and second supervisor,who may decide to undertake further investigation that may ultimately lead to disciplinary actions, should instances of plagiarism be detected.

(5) I have read the Northumbria University Policy Statement on Ethics in Research and I confirm that ethical issues have been considered, evaluated and appropriatelyaddressed in this research.

SIGNED:

Luke

DATE: 28/04/23

Abstract

Background: Alzheimer's disease (AD) is a type of dementia that progressively damages brain cells, resulting in memory and thinking deficits, loss of basic abilities, and ultimately death. While there is no cure for AD, its onset diagnosis can prevent it from becoming too severe and improve patient's life. Recent advancements in computer vision have found to be impactful, however the datasets for AD are limited and heavily imbalanced. Conventional approaches for addressing these issues are appropriate but not optimal and GANs based approaches are expected to outperform them. However, the predominantly used DC-GAN is prone to suffer from "Mode Collapse". This issue was resolved by using WGANs-GP.

Methods: This research uses both experimental and simulation methods and utilises WGAN-GP for generating synthetic MRIs for addressing the issue of dataset scarcity for AD and class imbalance in the Kaggle Alzheimer's Dataset of axial MRIs. The quality of these synthesised MRIs was accessed using metrics like FID Score, SSIM, PSNR, SD and Seaborn's Distplot. The effectiveness of this approach was evaluated by comparing its results with those of the conventional approaches of oversampling, SMOTE, image augmentation and cost sensitive learning to validate the hypothesis of this research.

Results: The synthetic MRIs were found to be as good as the original ones both in terms of quality and diversity as they resulted in mean FID Score of 0.13, mean SSIM of 0.97, mean PSNR of 32 dB, and mean Sharpness Difference (SD) of 0.04 (over all the minority classes) which are all close to their ideal values. This approach resulted in an overall increase of 11.77% in Balanced Accuracy, 15% increase in Matthew's Correlation Coefficient (MCC). The study also found a 91.4% improvement in the performance on minority classes, at the expense of a 1% reduction in performance on the majority class. Finally, the results of this approach were compared with those of similar researches which used DC-GANs to generate synthetic MRIs for addressing class imbalance in Kaggle Alzheimer's Dataset and it was found that both the approach and the final product produced in this study are superior than the other researches.

Conclusions: The study found that using WGAN-GP to address class imbalance in the Kaggle Alzheimer's Dataset was not only successful but also superior to conventional approaches. The final research product was made publicly available on Kaggle which addressed the issue of dataset scarcity for AD. Additionally, this approach can be extended to automate the diagnosis of AD using all three parts of MRI in 3D and can also be applied to other medical problems facing similar issues.

Contents

1. Introduction

The most prevalent form of dementia is Alzheimer's Disease (AD) [1,2], which demands extensive medical care. It disrupts fundamental cognitive abilities, causing harm to brain cells, memory loss and hinders thinking skills [3,4]. The condition progressively deteriorates from very mild to severe, making it challenging for an individual to perform daily tasks without assistance [5]. The lack of a cure for AD presents the biggest challenge for specialists in this field [6]. However, early detection and the use of available treatments can help to alleviate symptoms or slow their progression [7].

Upon reviewing and examining existing studies, several common trends and shortcomings have been identified. One of the most notable trends is the rapid growth in the detection and prediction of AD using machine learning techniques. However, there are also significant gaps, including an imbalance of events with attributes, scarcity of dataset to train deep learning models, class imbalance, overtraining and lack of external validation [8]. Despite these gaps, well designed and validated studies have shown that machine learning methods can enhance the accuracy of AD prediction compared to traditional statistical methods [9]. Amongst all these gaps, two of the most prominent ones are dataset scarcity and class imbalance [10]. Dataset scarcity, which often occurs due to confidentiality of medical data, can result in overfitting, inconsistent accuracy, lack of representativeness, and unreliable performance metrics [10]. On the other hand, class imbalance which can occur due to skewed distribution of people suffering from each stage of AD can cause the classifiers to be biased towards the majority class, leading to incorrect classifications such as identifying a person with early symptoms as "No Alzheimer's" [10].

The focus of this research is to generate synthetic MRI scans for AD using Generative Adversarial Networks (GANs) on Kaggle Alzheimer's dataset. This is expected to address both the class imbalance problem and dataset scarcity. It is also important to note that this research is limited to axial part of MRIs. If other parts including coronal and sagittal views were included, then classifiers are expected to be more robust and effective.

1.1 Hypothesis

"The hypothesis of this project is that Synthetic MRIs would be of sufficient quality for deep learning models to discover new patterns, resulting in overall improved performance and particularly significant improvement for the minority classes which would effectively address the challenge of class imbalance and would give better results than conventional approaches for solving class imbalance."

1.2 Research Question

"What is the impact and effectiveness of using Synthetic MRIs generated by Generative Adversarial Networks (GANs) on the performance of Deep Learning (DL) models in detecting Alzheimer's Disease (AD), particularly in terms of resolving the class imbalance problem and scarcity of medical data for Alzheimer's?"

1.3 Aim

Deep Learning models require substantial amounts of data to train on, in order to perform well and produce credible results, however, medical problems usually suffer from scarcity of data. There are currently only four publicly available datasets for Alzheimer's i.e., "ADNI" [11], "OASIS" [12], "EPAD" [13] and "Kaggle Alzheimer's Dataset" [14]. All of these datasets have severe class imbalance.

This research only focusses on Kaggle Alzheimer's Dataset and aims to address the challenges of scarcity of dataset and class imbalance issue by implementing GANs to generate synthetic MRI scans for its each minority class.

1.4 Project Objectives

This research has the following S.M.A.R.T. objectives

1. ***Exploring the current approaches to deep learning for detecting and solving the class imbalance problem in the diagnosis of Alzheimer's Disease. Having a thorough understanding of Alzheimer's is crucial for making informed decisions and interpretations. Additionally, having a clear comprehension of algorithms and techniques will help with the practical side of conducting research in this field.***
    - Undertake a literature review around different machine learning techniques and approaches for detecting Alzheimer's
    - Undertake a literature review around different techniques to solve class imbalance problem for Alzheimer's
    - Combine the literature from the above two areas to identify the limitations in other researcher's approaches to support the rationale behind approach for this study.

2. ***Generate synthetic images for each minority class in the Kaggle Alzheimer's dataset and evaluate the quality of the synthetic images. This will provide the synthetic dataset which would be the final product of this research on which pretrained models/CNNs will be evaluated and compared.***
    - Identify the appropriate type of GAN to generate synthetic images for each minority class by examining the limitations in current methods as discovered through the literature review.
    - Split the Kaggle Alzheimer's dataset into train and test sets and implement the appropriate GAN to generate synthetic images for each minority class separately only using the images in the train dataset and not the test dataset.
    - Determine the appropriate qualitative and quantitative methods/metrics to assess the quality of synthetic images with respect to real images for each minority class.

3. ***Develop, evaluate and compare several deep learning models for classifying Alzheimer's MRIs into their respective classes. The best performing classifier will be used to test the hypothesis of this research.***
    - Identify the appropriate metrics for evaluating the performance of pretrained models/custom CNN for severely imbalanced datasets.
    - Train, evaluate and compare multiple pretrained models/custom CNN using the appropriate quantitative metrics.
    - With the test data remaining fixed, evaluate the hypothesis of this research by comparing the performance of the best performing pre-trained model/custom CNN. This comparison will be made by analyzing the model's performance when it was trained solely on real MRIs versus when it was trained on the bigger balanced dataset of real and synthetic MRIs.

4. ***Evaluate the results and approach of this research project.***
    - Evaluate the results and effectiveness of the approach for this project in comparison to previous research on machine learning techniques to solve the class imbalance problem.
    - Discuss GRAD-CAMs [15] using the best classifier so far, to identify which parts of the MRIs did the model focused on, in order to make its classification decisions.
    - Conduct a self-reflection of the research project process to identify what aspects were effective and what areas could be improved.
    - Determine avenues for future work and acknowledge the limitations of the current study.

5. ***Publish the repository and the final product (Real + Synthetic MRIs combined dataset) of this research.***
    - Organize all files into appropriate folders and publish all the work for this research study as a repository on either GitHub or OneDrive, taking data management and reproducibility into consideration.
    - Publish the final product, the combined dataset of real and synthetic MRIs, as open source on Kaggle and accompany it with an adequate description of the data, being cognizant of ethical, legal, social, professional, and security concerns relevant to this research study.

6. ***Present the results of the research study in a dissertation and viva***
    - Create a draft of the dissertation for the supervisors to review, in compliance with the KF7029 guidelines.
    - Prepare for the presentation and viva for KF7029 MSc Computer Science and Digital Technologies project.

1.5 Research Approach

This study adopts a deductive (top-down) research approach, as it hypothesizes that utilizing a multiclass, well-balanced dataset of real and synthetic MRIs will result in an improved overall performance and effectively address the class imbalance problem, particularly in terms of performance on the minority classes. This research approach includes both experimental and simulation components. In the experimental component, various pretrained models/custom CNN are evaluated and compared based on their performance on a fixed test dataset. The best classifier will be used to test the research hypothesis. The simulation component involves using WGAN-GP to generate new data from Gaussian noise and uncover patterns in the input dataset to produce synthetic MRI scans. If the hypothesis is supported, this finding would suggest that synthetic MRIs are an effective simulation of real ones. The code is available at: [Link]

The effectiveness of this research approach will be compared to conventional approaches to validate its superiority. Finally, the best performing classifier so far, which overcame the issue of class imbalance, will be used to implement GRAD-CAMs to visualize, which parts of the image did the classifier focussed on, in order to make its classification decisions. If this research approach is successful then the final product of this research would be made publicly available by publishing it on Kaggle. This will address the issue of scarcity of medical dataset for Alzheimer’s.

1.6 Structure of The Dissertation

The structure of the remaining dissertation is as follows: it begins with a systematic and critical review of the literature related to this study. This review encompasses a comprehensive examination of the current methods for the early detection of Alzheimer's disease, as well as the solutions proposed to tackle the class imbalance problem. The review also evaluates the benefits and limitations of these methods with the aim of drawing meaningful conclusions and insights that can inform the current study.

The following chapter presents the principal research approach and offers an in-depth discussion on the detailed design and methodology for the generation of synthetic images. It also examines the machine learning techniques used to address the class imbalance problem through the utilization of synthetic MRIs and conducts a comparison with conventional approaches to evaluate their effectiveness.

The next chapter presents the major findings of the study and provides a thorough analysis of these results. The subsequent chapter conducts an evaluation of the primary outcomes of the study and includes the researcher's personal reflection on their journey. Finally, the last chapter summarizes the key findings of the study, proposes directions for future research, and acknowledges the limitations of the current study.

2. Literature Review

2.1 Background:

Alzheimer's disease (AD) is the most common form of dementia [1,2] accounting for 60-80% of all forms of dementia [15,16] that progressively kills brain cells, resulting in deficits in memory and thinking capacity, and eventually leading to loss of basic tasks [3,4]. The pathogenesis of AD is caused by the accumulation and overproduction of amyloid-β (Aβ) plaques and hyperphosphorylation of tau protein [17], disrupting the nucleocytoplasmic transport between neurons, leading to cell death, and causing memory and learning loss. The symptoms develop slowly and gradually become severe with time [5]. Although old age (over 65) [18] is the primary risk factor for AD, but it is not exclusively age-related and is more prevalent in women than in men [19]. Recent data from the World Alzheimer's Association indicates that over 4.7 million individuals aged over 65 in the United States have been diagnosed with Alzheimer's disease [18]. This number is projected to increase to 152 million people by 2050, with people developing AD every 3 seconds [20]. The estimated annual cost of AD is expected to reach $1 trillion, with predictions indicating that this figure will double by 2030 [21].

Alzheimer's disease can be categorized into the following stages [22]:

1. Very Mild Impairment: This stage is characterized by memory loss, which is common as people age. However, for some individuals, this may lead to AD. Those with very mild impairment may experience cognitive difficulties that impact their daily lives, such as forgetfulness, uncertainty, changes in personality, getting lost, and difficulties with routine tasks.
2. Mild Impairment (MCI): Patients at this stage require extra care and support, and their everyday lifestyle becomes more complex. Symptoms are similar to very mild impairment, but more severe. Patients may need assistance with basic activities such as combing their hair, and may exhibit significant personality changes such as becoming paranoid or irritable without reason. Sleep disorders are also common.
3. Moderate or Severe Impairment: Symptoms may worsen during this stage. Patients may lose the ability to communicate, and may require full-time treatment. They may also lose bladder control and be unable to perform basic actions such as keeping their head up or sitting in a chair. It usually takes 4 years to transition from mild to severe Alzheimer’s.

There is no cure for AD [5] which makes its early diagnosis crucial as it facilitates prompt treatment, minimizes medical care expenses, enables accurate diagnosis using advanced diagnostic techniques, prevents the disease from becoming too severe and improves the patient's quality of life [18]. Traditional methods of AD diagnosis are dependent on manual feature extraction, which is both time-consuming and subjective [22]. In contrast, the application of ML techniques, especially those utilizing CNNs, can automate feature extraction and enhance efficiency in AD diagnosis [23]. The complexity of multivariate heterogeneous data in AD diagnosis, coupled with the fact that the detection of AD has become a diagnostic indicator for other forms of dementia, creates a challenging task of manually comparing, visualizing, and analysing the data [24,25]. Therefore, ML plays a crucial role in AD diagnosis by enabling the management of the vast amount of data and enhancing diagnostic accuracy [26].

Biomarkers are measurable indicators of a biological state or condition that can be used to diagnose, monitor, or predict disease progression [27]. There are five types of biomarkers for detecting AD: neuroimaging of β-amyloid protein deposition, cerebrospinal fluid (CSF), photon emission tomography (PET), single photon emission computerised tomography (SPECT) and Magnetic Resonance Images (MRIs) [4,27]. However, data on all modalities for individual subjects is insufficient for reasonable classification. CSF biomarkers are promising for early AD diagnosis, but are costly and invasive with potentially painful lumber punctures [28,29]. PET can cause patients to experience distress and exposure to ionizing radiation [15]. Contrarily, MRIs have imaging flexibility, tissue contrast, no ionizing radiation, and can provide useful information on brain anatomy [5]. MRIs are typically 3D, but can be converted to 2D by dividing into axial, coronal, and sagittal parts.

There are currently only four open source verified MRI datasets available for AD which are ADNI [10], OASIS [11], EPAD [12] and the Kaggle Alzheimer’s dataset (axial MRIs only) [13]. After reviewing various studies, the main gaps in the ongoing research have been discovered including an imbalance of events with attributes, class imbalance, overfitting (high model parameters), and a lack of external testing and validation due to scarcity of datasets (due to patient-doctor confidentiality) of the same biomarkers used for training [29].

NOTE: Read the preliminary section in the appendix for understanding the basics of CNNs and their evaluation metrics (Appendix A1-A4) and GANs and their evaluation metrics (Appendix B1-B3) before proceeding to the next section

2.2 Related Works:

This section reviews conventional and generative approaches employed by several researchers to tackle the issue of class imbalance and scarcity of datasets for AD.

2.2.1 Conventional Approaches:

Lu et al. [30] developed a new deep neural network with a multistage technique for AD identification, achieving 82.4% accuracy for MCI prediction and 94.23% sensitivity for Alzheimer's disease and those patients later got exposed to AD in three years. Gupta et al. [31] proposed a diagnosis method combining coronal, axial and sagittal features from MRI scans, achieving 96.42% accuracy for AD classification. Ahmed et al. [32] proposed an ensemble CNN model with 90.05% accuracy for AD diagnosis using left and right hippocampus area in MRI scans to prevent overfitting. Mehmood et al. [33] extracted Grey Matter (GM) using tissue segmentation and achieved 98.73% accuracy in classifying AD vs NC and 83.72 % accuracy in classifying EMCI vs LMCI patients. Wang et al. [34] used a 3D ensemble model convolutional network to classify AD and MCI. The model consisted of 3D-DenseNets [35] optimized with a probability-based fusion approach, achieving a classification accuracy of 97.52% with the ADNI dataset. Janghel and Rathore [36] used pretrained VGG16 [37] to extract AD features from ADNI database, and then used SVM [38]. Linear Discriminate [39], K means clustering [40], and decision tree algorithms [41] and achieved 99.95% accuracy with random forests in functional MRI images and 73.46% average accuracy in PET images. Ge et al. [42] proposed a 3D multiscale deep learning architecture that achieved a test accuracy of 93.53% on a subject-segregated, random brain scan-partitioned dataset with an average accuracy of 87.24%. Different biomarkers are used to identify group differences in Alzheimer's disease, but they are not designed for individual classification. Due to the scarcity of datasets for a single type of biomarker, Zhang et al. [43] proposed a method that combines MRI, PET, and CSF biomarkers to differentiate between healthy and AD participants. Using a baseline dataset of 202 instances, they achieved a 93.2% classification accuracy with 10-fold cross-validation using all three modalities. The authors claimed that multimodal classification is more robust and consistently improves accuracy compared to individual modalities, which resulted in 86.5% accuracy.

All of these researchers performed binary classification using different datasets (mostly ADNI) and different modalities. It is easier to get higher accuracies with binary classification as it is easier for the models to learn patterns and differentiate between just 2 classes, but the problem with binary classification is that the models are prone to classify a person with early symptoms of AD as “Not Impaired” which is highly undesirable for diagnosis of Alzheimer’s at an onset stage. Thus, multiclass classification is important. However, limited availability of data for individual subjects on all modalities and class imbalance still remained significant challenges.

If the dataset is small, heavily imbalanced and if the model is too deep, then while backpropagation, the optimiser may not be able to update the parameters properly as the gradients might become too small when they reach the upper layers. This might result in overfitting or make the models more biased towards majority classes as the model may memorize the examples rather than learning the underlying patterns. Oversampling by randomly duplicating instances from the minority class until it reaches a desired ratio with the majority class can address class imbalance and can also tackle overfitting by increasing the size of the training set and reducing the model's sensitivity to noise in the data.

Eva et al. [44] attempted to address class imbalance in the Kaggle Alzheimer's Dataset by randomly oversampling minority classes. They achieved 60.67% and 60.75% accuracy using ResNet18 [45] and ShuffleNetV2 [46], respectively, with oversampling, compared to 50% and 50.27% without oversampling. Meanwhile, Ahmed et al. [47] used the DAD-Net [47] model and ADASYN [48] oversampling technique, which uses a weighted distribution to generate more synthetic instances near the boundary of the minority class and the majority class, to achieve 99.2% accuracy and address class imbalance in the same dataset. Similarly, Murugan et al. [49] also addressed class imbalance by implementing SMOTE [50] which uses KNN [51] and interpolates the nearest neighbouring images in the minority class to generate similar images, and achieved 99% accuracy with the Kaggle Alzheimer’s dataset.

One of the reasons for such significant improvement was that the number of images in the minority classes were so less that when they were oversampled and divided into train and test sets, the classifier was evaluated on the exactly same minority class images on which it was trained. This is cheating, rather they should have kept the test dataset separate since the beginning. While oversampling can improve accuracy by preventing overfitting, it may not be the best solution for class imbalance as the same minority class images are duplicated, preventing the model from learning any new patterns. This means that oversampling does not enhance the model's ability to generalize well to new patient data.

Oktavian et al. [52] utilized the ResNet-18 [45] architecture with mish activation function and a weighted loss function to address the issue of class imbalance in ADNI dataset of 306 MRIs. This resulted in increase in accuracy from 69.1% to 88.3%. Liu et al. [53] also tried something similar by using a type of hard negative mining technique that samples difficult cases during the training phase which resulted in the increase in accuracy from 72% to 89.7% with the same dataset. Although weighted loss or cost-sensitive learning can improve the model's performance on the minority class by assigning higher weights and penalising misclassifications on the minority classes more, it does not increase the diversity of images in the dataset. This approach does not add new patterns or variations to the dataset, which does not improve the model's robustness or generalization ability on future patient’s MRIs, nor does it address the issue of dataset scarcity. Therefore, cost-sensitive learning is not a scalable solution for addressing both class imbalance and dataset scarcity.

Helaly et al. [54] and Afzal et al. [55] tackled class imbalance issues in AD MRI datasets using different image augmentation techniques. Helaly et al. augmented and oversampled the ADNI dataset to 48,000 images by rotating and flipping the MRIs and built custom 2D and 3D CNN models achieving 93.61% and 97% accuracies respectively. Whereas, Afzal et al. overcame the imbalance in the OASIS dataset using their own augmentation method, which involved randomly cropping (left, right, top, bottom and whole) and rotating images (90, 180 and 270 degrees), resulting in 98.41% and 95.11% accuracies for their 2D and 3D CNN models, respectively. This indicated that combining all three MRI parts (3D) may not always outperform 2D, and it varies depending on the dataset involved and the techniques applied. While augmentation is a reliable technique for preventing overfitting and enhancing diversity, it may not always be suitable for addressing class imbalance or data scarcity because it doesn’t create entirely new images but only creates variations of existing images that can sometimes be unrealistic and implausible, causing confusion in the model and leading to poor performance.

While these approaches primarily relied on accuracy to evaluate their effectiveness, accuracy may not be suitable for highly imbalanced datasets as it could be dominated by good performance on the majority class.

2.2.2 GANs Based Approaches:

GANs [56] based approaches for Alzheimer’s Disease are mainly used for image denoising, image segmentation, modalities transfer, image enhancement, MRI aging and synthetic image generation/augmentation [57]. Amongst these approaches DC-GANs [58] has been the predominantly used GAN for addressing class imbalance and scarcity of datasets for AD [57].

Mukherjee et al. [59] and Hu et al. [60] used DC-GANs to generate synthetic images and solve the class imbalance problem in Alzheimer's datasets. Mukherjee et al. used the DC-GANs to create new MRI scans, which they used to oversample the minority classes in the Kaggle Alzheimer's dataset. This resulted in a 4% increase in accuracy when using the ResNet50 [45] model. However, they could not train their DC-GANs for more than 200 epochs due to the issue of mode collapse (explained in Appendix B1), which could have led to even better results. They also did not train their DC-GANs separately for each class, which caused mixed features in the resulting images that could confuse their classifier. Furthermore, they did not analyse the quality of the synthetic MRIs before feeding to the ResNet50 model, so it was unclear whether it was the poor quality of the synthetic MRIs or the classifier which limited the effectiveness of their approach.

In contrast, Hu et al. used DC-GANs to synthesize PET images to address class imbalance in the ADNI dataset, which had only 200 PET images. They trained their DC-GANs for 400 epochs and used Maximum mean discrepancy [61] and Structural Similarity Index Measure [61] (SSIM) to measure the similarity and diversity of their synthetic images. This resulted in a 7% increase in accuracy using the DenseNet [37] model. However, their results were not significantly improved because they used only 200 images and trained for only 400 epochs due to the issue of mode collapse in DC-GANs.

Islam et al. [62] also used DC-GANs with the ADNI dataset of 411 PET scans from 479 patients. They trained their DC-GANs separately for each class for 500 epochs and used Peak Signal to Noise Ratio [61] and SSIM [61] to analyse the quality of their synthetic PET scans. Their approach led to a 10% increase in accuracy using a custom CNN.

These GAN-based approaches suggest that GANs are expected to be superior to conventional methods because they can generate entirely new images that capture the full range of variations in minority classes, unlike conventional approaches that interpolate or oversample existing images. GANs can not only address class imbalance but also the scarcity of medical datasets in various medical problems. However, none of the approaches evaluated their effectiveness by comparing their results to conventional methods on their datasets or published their synthetic datasets to address the scarcity of datasets.

This research aims to address the issue of class imbalance and scarcity of datasets for AD by oversampling the minority classes in the train set of Kaggle Alzheimer's dataset with synthetic MRIs generated using WGAN-GP [63] which overcomes the issue of "Mode Collapse" and will be discussed in detail in the methodology chapter.

3. Methodology

3.1 Introduction:

This chapter details the methods used to tackle class imbalance in the Kaggle Alzheimer's Dataset, including the use of an experimental and simulation-based research approach. It also explains the process of synthesizing MRI scans and verifying the research hypothesis. Finally, Ethics related issues were also discussed. The code is available at [Link]

3.2 Justification and Rationale:

This project has incorporated insights gleaned from the extant literature to identify aspects that can be implemented for synthesizing the final product of this research. This chapter encompasses the fundamental framework behind generating the synthetic dataset as well as testing the hypothesis of this research study. This is being done by following a deductive (top down) research approach which has both experimental and simulative components. This research is deductive because it hypothesizes that the bigger balanced multiclass dataset of real and synthetic MRIs will result in improved overall performance and also much better performance on the minority classes. Additionally, each algorithm and method used in machine learning is based on specific assumptions about the characteristics of data, rendering them inapplicable to other types of data. Therefore, the results of this approach will be finally compared with the conventional approaches of addressing class imbalance which will validate the effectiveness of this approach.

The experimental component in this research approach involves evaluating and comparing various pretrained models and a custom CNN to determine the best classifier for testing the research hypothesis and also for comparing the results of this approach with the conventional approaches to access its effectiveness. The simulation component involves using WGAN-GP [63] to generate new data and uncover patterns in the input dataset to produce synthetic MRI scans. The purpose of this simulation component is to test whether synthetic MRIs are an effective simulation of real ones. Using synthetic data is particularly useful in cases where the availability of real data is limited, or the process of collecting real data is expensive or time-consuming. Therefore, by combining experimental and simulation methodologies, the researchers can leverage the strengths of each approach and obtain more comprehensive insights into the problem of interest. The experimental component allows the researchers to establish the validity and reliability of the best classifier, while the simulation component enables them to test their hypothesis using synthetic data that mimics the properties of real data.

3.3 Data selection and Description:

The Kaggle Alzheimer's Dataset [13] contains axial MRI scans of the brain in four classes: "No Impairment," "Very Mild Impairment," "Mild Impairment," and "Moderate Impairment". The number of subjects in each class were 100, 70, 28, and 2, respectively. The dataset includes only axial MRI scans and no patient-specific information. This dataset was chosen for its severe class imbalance in the minority classes which made it ideal for hypothesis of this study, with 3,200 images in "No Impairment," 2,240 in "Very Mild Impairment," 896 in "Mild Impairment," and 64 in "Moderate Impairment." Each image initially was of 176 x 208 pixels, but was resized and rescaled to 128 x 128 pixels for model training. Sample images from each class are illustrated below:

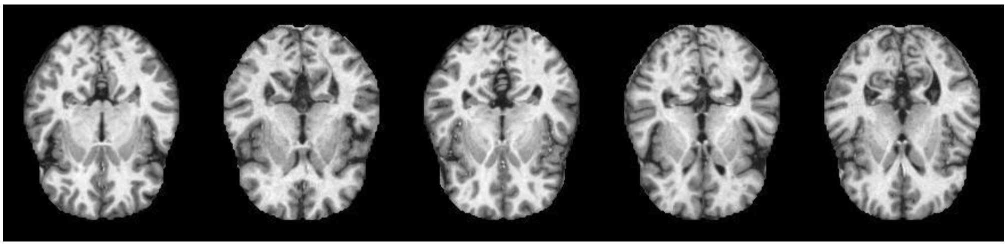


Fig 3.3(1) Sample Images for "No Impairment"

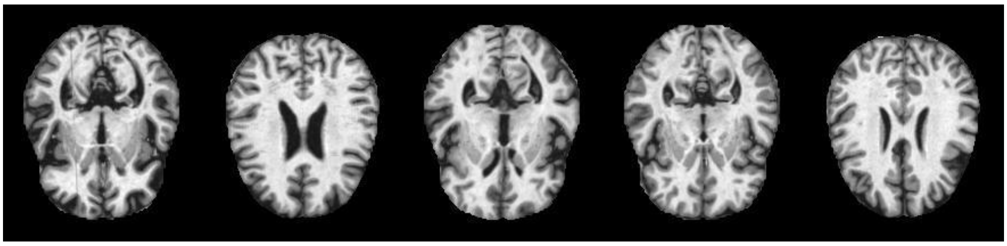


Fig 3.3(2) Sample Images for "Very Mild Impairment"

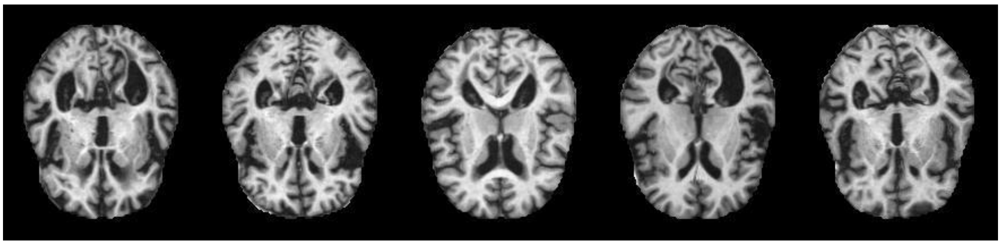


Fig 3.3(3) Sample Images for “Mild Impairment”

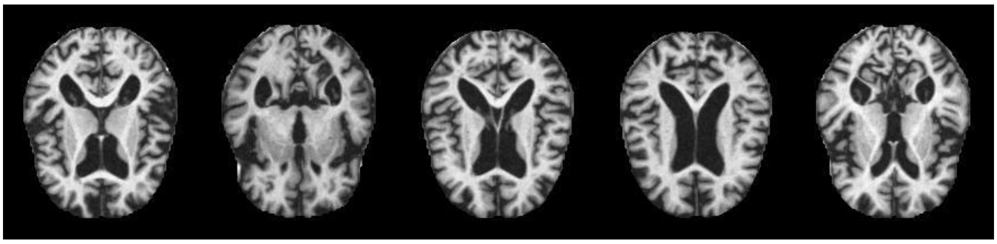


Fig 3.3(4) Sample Images for “Moderate Impairment”

3.4 Requirements and Reproducibility:

The deep learning models employed in this project were built using Keras API [64] of TensorFlow library [65] and were computationally expensive which required the use of high-end GPUs for their training. To fulfil this requirement, the Kaggle’s Tesla P100 (16 GB) GPU and Google Colab Pro Plus’s Tesla A100 (40 GB) GPU were utilized for synthesizing and classifying the MRIs.

To ensure reproducibility of this research approach, the seed for the kernel, TensorFlow, and Image Data Generator [64] were fixed. Since the dataset involved in this research was small and heavily imbalanced, it was resulting in inconsistent performance metrics (such as accuracy), each time the models were re-trained even if the seed was fixed. To mitigate this and make the results reproducible, after the models had been trained, they were saved in form of “.h5” file formats.

3.5 Rationale Behind WGAN-GP [63]:

As discussed previously both in Appendix B2 and in the literature review, DC-GANs [36] just like GANs suffers from “Mode Collapse” and is very sensitive to hyperparameters and model architecture. WGANs-GP overcomes the issue of mode collapse and is less sensitive to model architecture and hyperparameter configurations because it uses Wasserstein distance as the loss function. Wasserstein distance is a more stable measure of the difference between generated and real data distributions than Binary Cross Entropy used in DC-GANs. It calculates the minimum cost of moving mass to convert data distribution from $\mathbb{P}_g$ to $\mathbb{P}_r$. The Wasserstein distance for the real data distribution $\mathbb{P}_r$ and the generated data distribution $\mathbb{P}_g$ is mathematically defined as the greatest lower bound (infimum) for any transport plan

$$\boldsymbol{W}(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{x,y\sim\gamma}[\|\boldsymbol{x} - \boldsymbol{y}\|]$$

Where $\boldsymbol{\Pi}(\mathbb{P}_r, \mathbb{P}_g)$ denotes the set of all joint distributions $\boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{y})$ whose marginals are respectively $\mathbb{P}_r$ and $\mathbb{P}_g$. Using Kantorovich-Rubinstein duality $\boldsymbol{W}(\mathbb{P}_r, \mathbb{P}_g)$ is equivalent to:

$$\boldsymbol{W}(\mathbb{P}_r, \mathbb{P}_g) = \sup_{\|f\|_{L\leq 1}} \mathbb{E}_{x\sim\mathbb{P}_r}[\boldsymbol{f}(\boldsymbol{x})] - \mathbb{E}_{x\sim\mathbb{P}_g}[\boldsymbol{f}(\boldsymbol{x})]$$

where the supremum (maximize) is taken over all 1-Lipschitz functions. If this equation is multiplied by (-1) then the loss function (L) will be to find the infimum (minimize) which is what optimizers can work with.

To ensure f is a 1-Lipschitz function, WGAN uses a simple weight clipping approach that restricts the maximum weight value in the discriminator/critic to a certain range determined by hyperparameter c. However, weight clipping can sometimes limit the model's capacity and hinder its ability to model complex functions, causing WGAN to generate poor samples or fail to converge. Large clipping parameters slow the rate at which weights reach their limit, making it harder to train the critic. On the other hand, small clipping parameters, especially in the presence of many layers or without batch normalization, can result in vanishing gradients. To resolve this issue, gradient penalty was introduced in WGAN-GP. A function f is 1-Lipschitz if its gradients have a norm of 1 everywhere. Rather than using weight clipping, WGAN-GP penalizes the model when the gradient norm deviates from its target norm value of 1.

$$\boldsymbol{Generator's\ Loss} = \frac{1}{m}\sum_{i=1}^{i=m} -D_w\left(G_\theta(z)\right)$$

$$\boldsymbol{Critic's\ Loss\ with\ G.P.} = \underbrace{\mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_g}[D_w(\hat{x})] - \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D_w(x)]}_{Critic's\ Loss\ in\ WGAN} + \lambda \underbrace{\mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_g}[(\|\nabla_{\hat{x}}D_w(\hat{x})\|_2 - 1)^2]}_{Gradient\ Penalty}$$

Where $\hat{x} = G_\theta(z)$, $z \sim p(z)$ (the input to the generator sampled from random noise of distribution p) and D_w is the set of 1-Lipschitz functions. In contrast to DC-GANs, which uses a discriminator to classify generated images as real or fake, WGAN-GP replaces the discriminator with a critic that performs regression to score "realness" or "fakeness" of an image, instead of classification. For this reason, the critic in WGANs-GP does not use a "sigmoid" activation function in its last layer and assigns a score close to -1 for real images and close to 1 for fake images (as opposed to 1 and 0, respectively, in DC-GANs).

In DC-GANs, the generator learns less when the discriminator excels at distinguishing real and fake images since the discriminator's loss becomes saturated, leading to small gradients for the generator's parameter updates which can cause slow convergence or even non-convergence of the training. Conversely, WGANs-GP prevents the generator from producing samples far from the real data distribution as the critic becomes better at identifying real and generated data which is why critic is trained five times for each generator's update, and gradient penalty is applied only to the critic's loss. This ensures the critic's gradient is continuous, which is critical for training stability. The generator's loss is based only on the critic's evaluation of generated samples and thus does not require gradient penalty

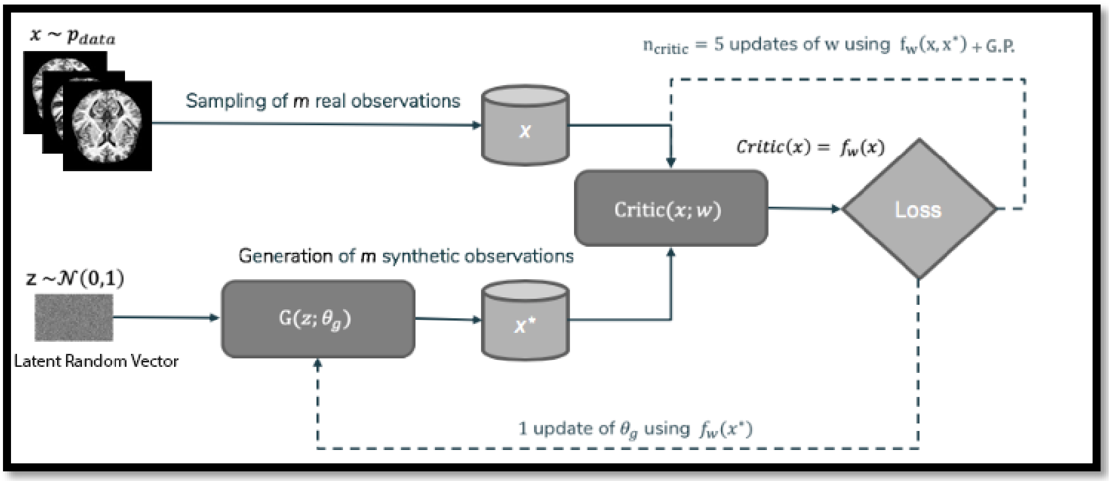
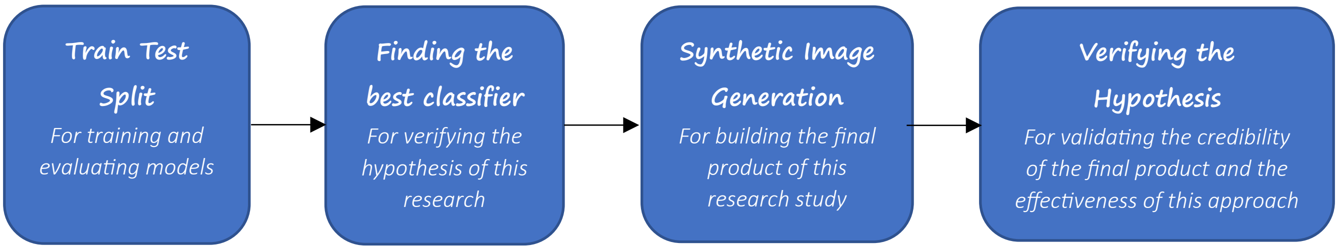

$x \sim p_{data}$
n_{critic} = 5 updates of w using $f_w(x, x^*)$ + G.P.
Sampling of m real observations
x
$Critic(x) = f_w(x)$
$Critic(x; w)$
Loss
Generation of m synthetic observations
$z \sim \mathcal{N}(0,1)$
$G(z; \theta_g)$
x^*
Latent Random Vector
1 update of θ_g using $f_w(x^*)$

Fig 3.5(1) WGAN-GP Network

The benefit of using WGANs-GP over DC-GAN is that they have a more stable training process and are less sensitive to model architecture and hyperparameter configurations. Moreover, unlike Wasserstein Distance, Binary Cross Entropy in DC-GANs has nothing to do with the quality of images produced at each epoch. The Wasserstein distance in WGANs-GP produces better results and stable training compared to DC-GANs, preventing mode collapse and generating high-quality, diverse data that captures the real data distribution.

3.6 Methodology Pipeline:

A pipeline automates the creation of a final product, such as a machine learning model or a dataset [66]. This study used a four-step pipeline. First the dataset was split into train and test sets for model training and evaluation respectively. Then multiple pre-trained models and a custom CNN were trained, evaluated and compared to find the best classifier. Subsequently, synthetic images were generated using WGAN-GP and evaluated for synthesizing the final product and finally the best classifier was used to verify the study's hypothesis, ensuring the credibility of the final product and the effectiveness of this methodology.



Train Test Split
For training and evaluating models

Finding the best classifier
For verifying the hypothesis of this research

Synthetic Image Generation
For building the final product of this research study

Verifying the Hypothesis
For validating the credibility of the final product and the effectiveness of this approach

Fig 3.6(1) Methodology Workflow

3.7 Train Test Split:

In machine learning, the test train split technique divides a dataset into a training set for model training and a testing set for performance evaluation on unseen data [67]. The split is usually random with a certain percentage assigned to each set. It helps to detect overfitting, underfitting, and if the performance is unsatisfactory, the model can be fine-tuned, retrained and re-evaluated. In this project, the Kaggle Alzheimer's dataset was split into 80% for training and 20% for testing, randomly for each class.

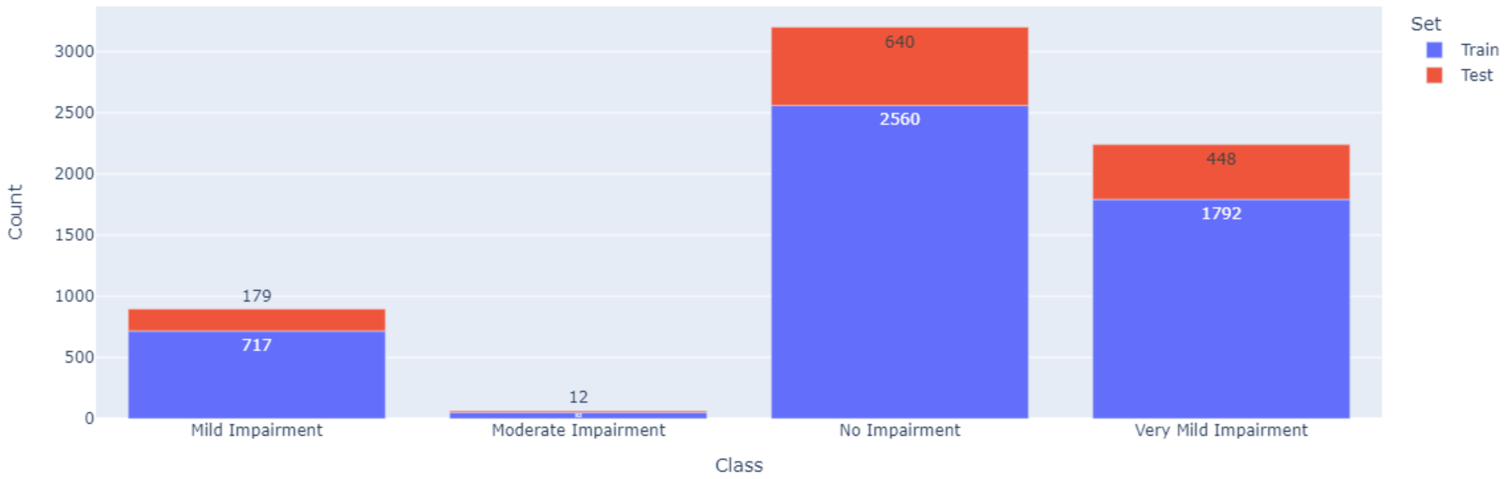


Fig 3.7(1) Distribution of images in each class after train-test split

3.8 Finding The Best Classifier:

This section involves the following steps for identifying the best performing classifier, which will be later used for testing the hypothesis of this research:

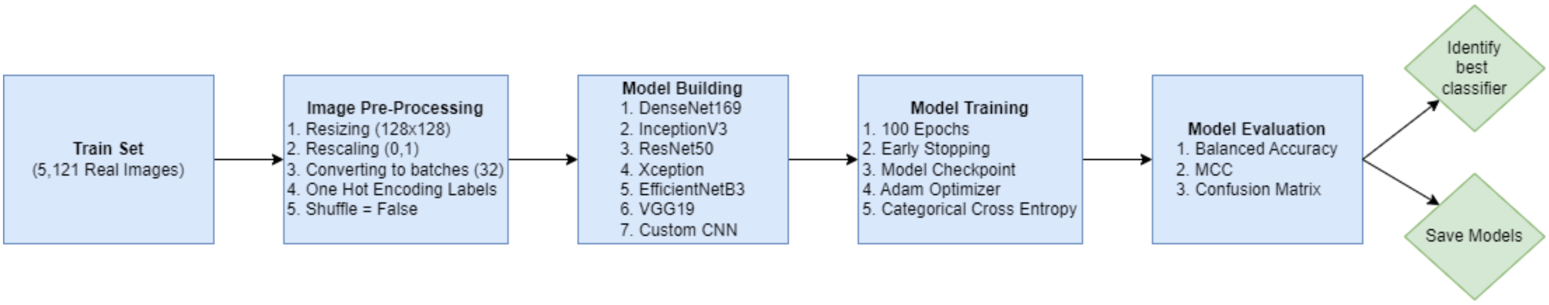


Fig 3.8(1) Workflow for finding the best classifier

3.8.1 Image Pre-Processing:

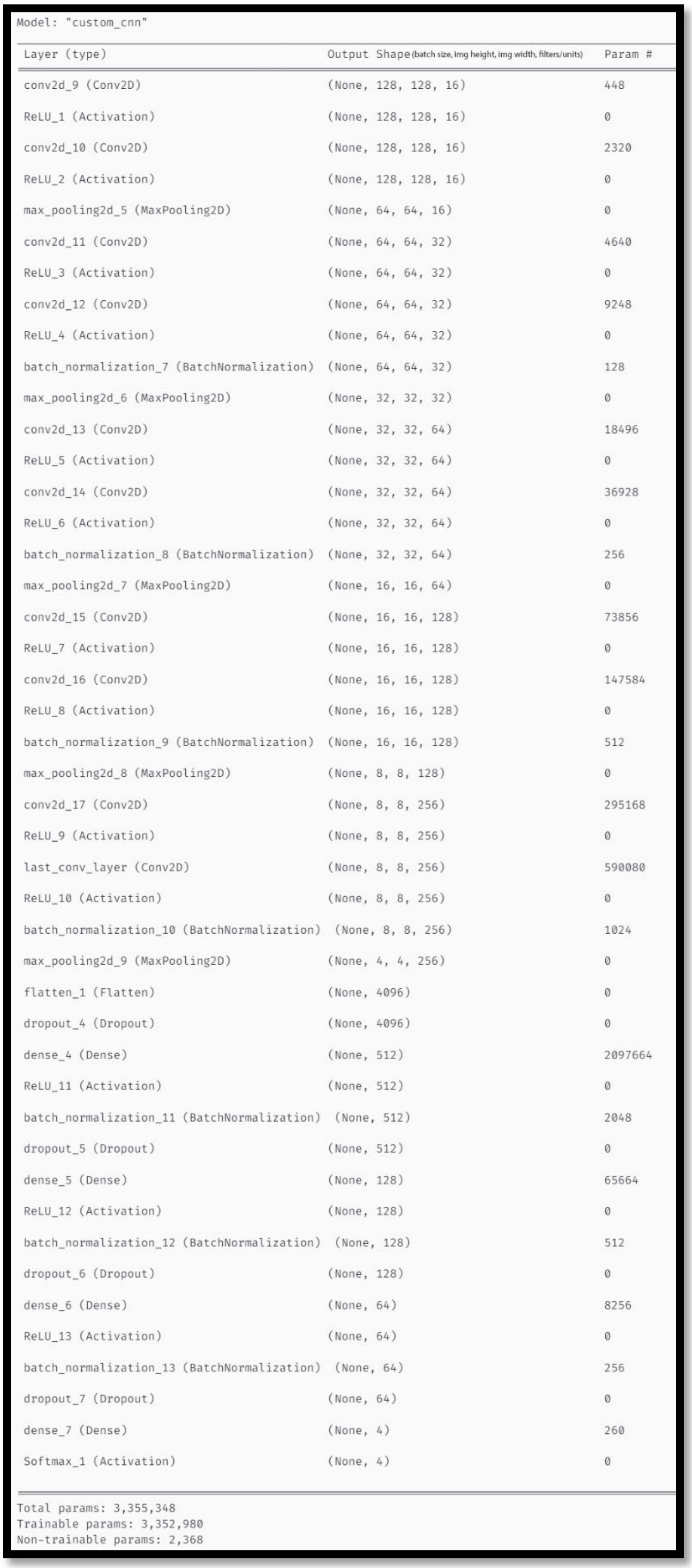
1. Using Keras's Image Data Generator class [64], "***train_generator***" and "***test_generator***" objects were created to pre-process images in train and test sets separately.
2. The parameter "***target_size***" was set to ***(128,128)*** for resizing images to 128x128 pixels which ensured that all input images will have the same dimensions. This was important because neural networks require fixed-sized input images.
3. The parameter "***rescale***" was set to ***(0,1)*** to avoid the issue of exploding gradients and improving performance as many activation functions are designed to work between the values of 0 and 1.
4. The parameter "***color_mode***" was set to "***rgb***" so that the grayscale MRIs are read as 3 channel RGB images. This was important as pretrained models expect a RGB image as input.
5. The parameter "***class_mode***" was set to "***categorical***" to convert the class labels into one hot encoded vector which is necessary for softmax activation function used for multiclass classification.
6. The parameter "***batch_size***" was set to ***32*** which converted images to batches of 32 image arrays. This was important for CNNs because it allows for efficient processing of large datasets, parallel computing using GPUs, and regularization of the training process.
7. The parameter "***shuffle***" was set to "***False***" so that each time the multiple models are trained or evaluated on train and test sets, the images in those sets are arranged in the exactly same order which will result in fair evaluation and comparison.

3.8.2 Model Building:

Pre-trained models such as DenseNet169 [35], InceptionV3 [68], ResNet50 [45], Xception [69], EfficientNetB3 [70] and VGG19 [37] were built using the following steps:

1. Each of these pre-trained models were imported from Keras with "*ImageNet*" weights as the base model, with their top layers removed by setting the parameter "***include_top***" = ***False***.
2. Then all the layers of the base model were set to be *non-trainable* so that the optimizer won't update the weights of these layers and they will retain the "*ImageNet*" weights
3. Then a dense layer (output layer) with 4 units was added with "*softmax*" activation function to output the class probabilities for each of these 4 classes.

Then custom CNN was build using the following architecture:



```
Model: "custom_cnn"
________________________________________________________________________________________________________
 Layer (type)                                Output Shape(batch size, img height, img width, filters/units)   Param #
========================================================================================================
 conv2d_9 (Conv2D)                           (None, 128, 128, 16)         448

 ReLU_1 (Activation)                         (None, 128, 128, 16)         0

 conv2d_10 (Conv2D)                          (None, 128, 128, 16)         2320

 ReLU_2 (Activation)                         (None, 128, 128, 16)         0

 max_pooling2d_5 (MaxPooling2D)              (None, 64, 64, 16)           0

 conv2d_11 (Conv2D)                          (None, 64, 64, 32)           4640

 ReLU_3 (Activation)                         (None, 64, 64, 32)           0

 conv2d_12 (Conv2D)                          (None, 64, 64, 32)           9248

 ReLU_4 (Activation)                         (None, 64, 64, 32)           0

 batch_normalization_7 (BatchNormalization)  (None, 64, 64, 32)           128

 max_pooling2d_6 (MaxPooling2D)              (None, 32, 32, 32)           0

 conv2d_13 (Conv2D)                          (None, 32, 32, 64)           18496

 ReLU_5 (Activation)                         (None, 32, 32, 64)           0

 conv2d_14 (Conv2D)                          (None, 32, 32, 64)           36928

 ReLU_6 (Activation)                         (None, 32, 32, 64)           0

 batch_normalization_8 (BatchNormalization)  (None, 32, 32, 64)           256

 max_pooling2d_7 (MaxPooling2D)              (None, 16, 16, 64)           0

 conv2d_15 (Conv2D)                          (None, 16, 16, 128)          73856

 ReLU_7 (Activation)                         (None, 16, 16, 128)          0

 conv2d_16 (Conv2D)                          (None, 16, 16, 128)          147584

 ReLU_8 (Activation)                         (None, 16, 16, 128)          0

 batch_normalization_9 (BatchNormalization)  (None, 16, 16, 128)          512

 max_pooling2d_8 (MaxPooling2D)              (None, 8, 8, 128)            0

 conv2d_17 (Conv2D)                          (None, 8, 8, 256)            295168

 ReLU_9 (Activation)                         (None, 8, 8, 256)            0

 last_conv_layer (Conv2D)                    (None, 8, 8, 256)            590080

 ReLU_10 (Activation)                        (None, 8, 8, 256)            0

 batch_normalization_10 (BatchNormalization)  (None, 8, 8, 256)           1024

 max_pooling2d_9 (MaxPooling2D)              (None, 4, 4, 256)            0

 flatten_1 (Flatten)                         (None, 4096)                 0

 dropout_4 (Dropout)                         (None, 4096)                 0

 dense_4 (Dense)                             (None, 512)                  2097664

 ReLU_11 (Activation)                        (None, 512)                  0

 batch_normalization_11 (BatchNormalization)  (None, 512)                 2048

 dropout_5 (Dropout)                         (None, 512)                  0

 dense_5 (Dense)                             (None, 128)                  65664

 ReLU_12 (Activation)                        (None, 128)                  0

 batch_normalization_12 (BatchNormalization)  (None, 128)                 512

 dropout_6 (Dropout)                         (None, 128)                  0

 dense_6 (Dense)                             (None, 64)                   8256

 ReLU_13 (Activation)                        (None, 64)                   0

 batch_normalization_13 (BatchNormalization)  (None, 64)                  256

 dropout_7 (Dropout)                         (None, 64)                   0

 dense_7 (Dense)                             (None, 4)                    260

 Softmax_1 (Activation)                      (None, 4)                    0

========================================================================================================
Total params: 3,355,348
Trainable params: 3,352,980
Non-trainable params: 2,368
```

Fig 3.8.2(1) Architecture of custom CNN

3.8.3 Model Training:

1. All of the pretrained models and custom CNN were compiled with Adam [71] as the optimizer with learning rate of 0.001, "Accuracy" and "F1 Score" as the metrics that will be computed after each epoch and "Categorical Cross Entropy" as the loss function since multiclass classification was being performed.
2. For callbacks, early stopping with patience = 10 and model checkpoint were implemented, both of which monitored validation loss. Early stopping was implemented to prevent the models from overfitting and patience was set to 10 because it's a thumb rule in machine learning that patience should be 10% of the total epochs [72]. Model checkpoint was implemented so that at the end of training each model will be loaded with those weights for which its validation loss was the minimum.
3. Finally, all of the pretrained models and the custom CNN were trained for 100 epochs by updating their weights using the training data and validating their performance after each epoch on the test data.

3.8.4 Model Evaluation:

After the models had been trained, they were evaluated on the fixed test dataset. Their resulting predictions (class probabilities from the output layer's softmax activation function) were converted into class labels using the "*numpy.argmax*" function. After this step, using these class labels, the overall performance of each of these classifiers were compared by computing Balanced Accuracy [73] and Matthews Correlation Coefficient [74] which revealed the best performing classifier. Finally, the models were saved in "*.h5*" format for the sake of reproducibility of results.

3.9 Synthetic Image Generation:

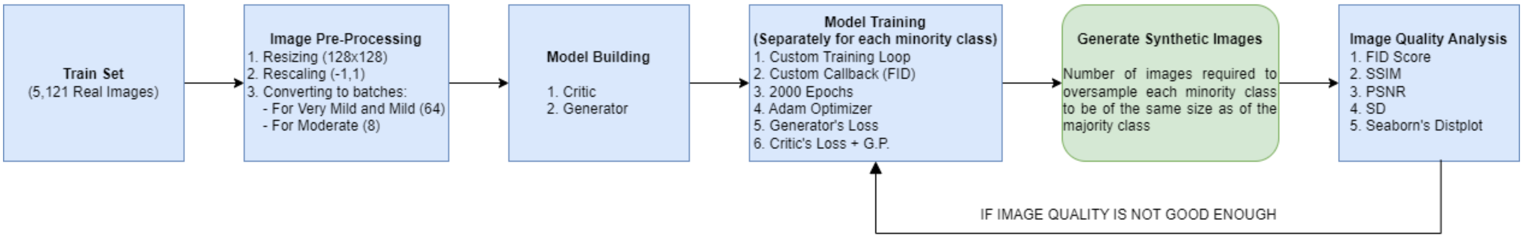


Fig 3.9(1) Steps for producing the synthetic MRIs dataset

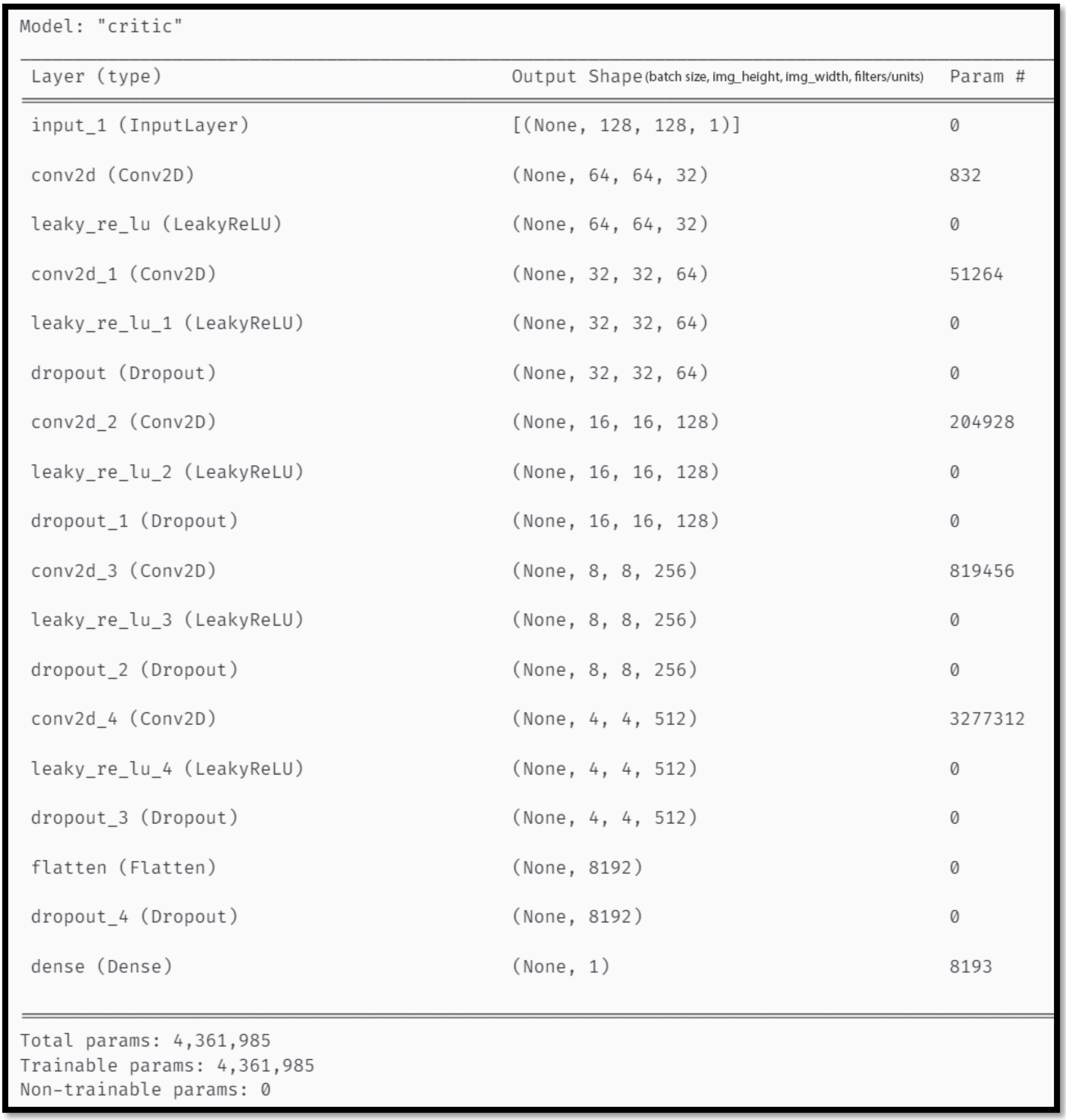
3.9.1 Image Pre-Processing:

Images for each class were resized to 128 x 128 pixels and rescaled to a range of (-1,1) to ensure that the output of the generator (synthetic images) and the input to the critic (real and synthetic images) are in the same range, because the generator typically outputs values between -1 and 1 using a "tanh" activation function in its output layer. If the real images are not of the same scale as of the synthetic images, the critic may have difficulty learning to distinguish between real and synthetic/fake images, and WGAN-GP may not converge or produce high-quality results.

3.9.2 Building WGAN-GP:

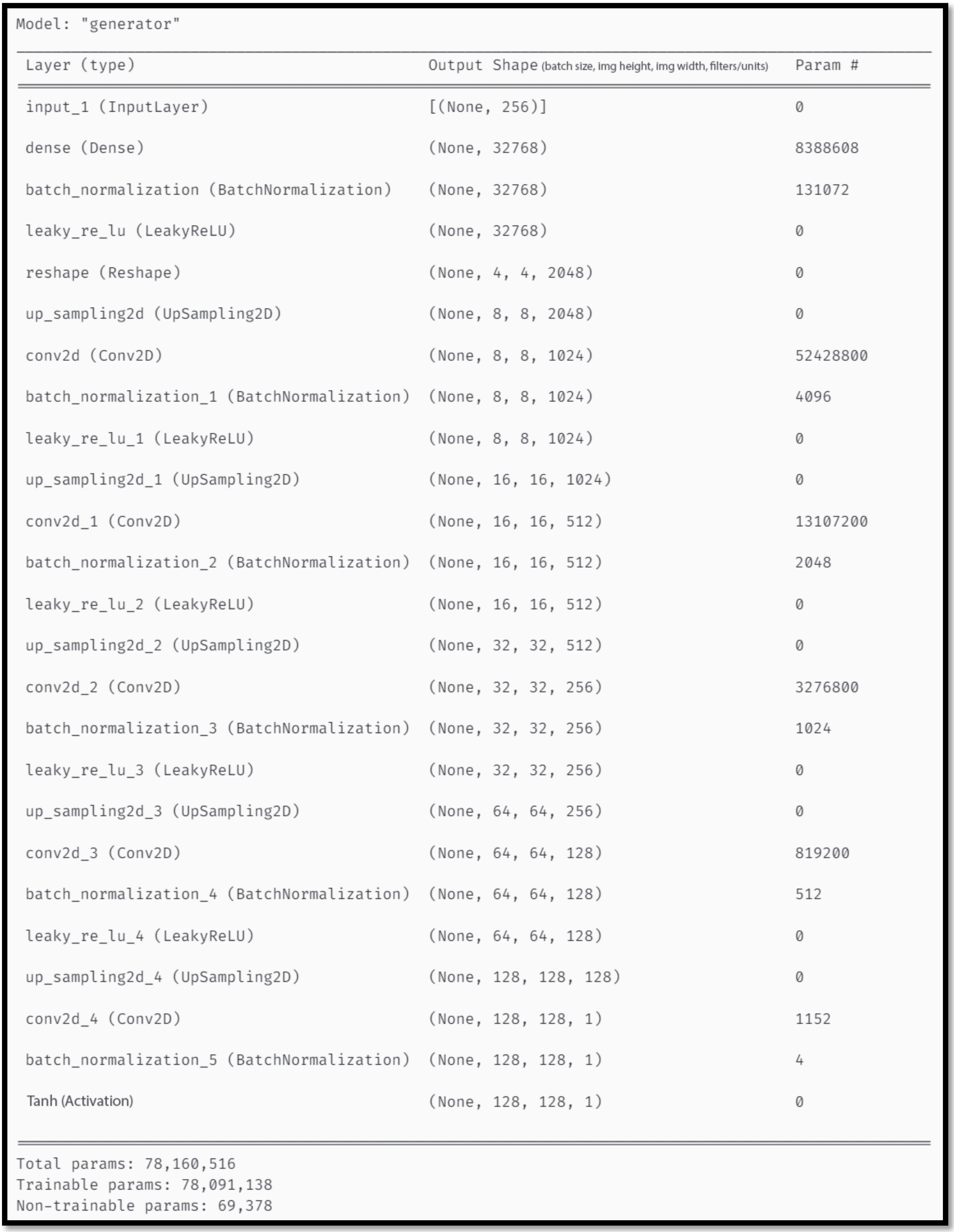
The critic and generator were built as per the proposed guidelines in the WGAN-GP paper [63]:

- Batch Normalisation [75] was used in the generator but not in the critic. Although, batch normalisation can stabilise training, it was omitted in the critic because it changes the form of critic's problem from mapping a single input to single output, to mapping an entire batch of inputs to a batch of outputs. This means that critic's gradient is calculated over entire batch of inputs and not for individual inputs due to which the penalized training objective which penalizes the norm of the critic's gradient with respect to each input independently is no longer valid. Instead, batch normalisation layer was replaced with dropout (30%) in critic which prevents overfitting and improves generalization
- Adam (lr = 0.0001, ß$_1$ = 0, ß$_2$ = 0.9) was used as the optimizer because it converges faster and reduces the chances of mode collapse [63].
- In gradient penalty term, lambda = 10 was used as it was found to work well in all their experiments with different datasets and ImageNet CNNs
- Second derivative of the activation function is needed to update the discriminator's parameters. Non-smooth functions like ReLU cause undefined gradients, making it hard to optimize. Therefore, Leaky ReLU with a slope of 0.2 was used as it allows non-zero gradients, making it easier to train WGAN with gradient penalties.
- The "Tanh" activation function is used in the last layer of the generator to scale the output to the range of [-1, 1] which generates a bigger range of data than 0 to 1, making it easier for the critic (Wasserstein distance) to distinguish between real and fake samples. Additionally, "Tanh" is symmetric and differentiable, which helps in reducing the likelihood of vanishing gradients during training.
- "Sigmoid" activation was omitted in the last layer of critic as it is no longer a classifier in WGAN-GP and it assigns a score close to -1 for real images and score close to 1 for fake images (instead of 1 and 0 respectively)



```
Model: "critic"
_________________________________________________________________________________________________
 Layer (type)                Output Shape(batch size, img_height, img_width, filters/units)  Param #
=================================================================================================
 input_1 (InputLayer)        [(None, 128, 128, 1)]     0

 conv2d (Conv2D)             (None, 64, 64, 32)        832

 leaky_re_lu (LeakyReLU)     (None, 64, 64, 32)        0

 conv2d_1 (Conv2D)           (None, 32, 32, 64)        51264

 leaky_re_lu_1 (LeakyReLU)   (None, 32, 32, 64)        0

 dropout (Dropout)           (None, 32, 32, 64)        0

 conv2d_2 (Conv2D)           (None, 16, 16, 128)       204928

 leaky_re_lu_2 (LeakyReLU)   (None, 16, 16, 128)       0

 dropout_1 (Dropout)         (None, 16, 16, 128)       0

 conv2d_3 (Conv2D)           (None, 8, 8, 256)         819456

 leaky_re_lu_3 (LeakyReLU)   (None, 8, 8, 256)         0

 dropout_2 (Dropout)         (None, 8, 8, 256)         0

 conv2d_4 (Conv2D)           (None, 4, 4, 512)         3277312

 leaky_re_lu_4 (LeakyReLU)   (None, 4, 4, 512)         0

 dropout_3 (Dropout)         (None, 4, 4, 512)         0

 flatten (Flatten)           (None, 8192)              0

 dropout_4 (Dropout)         (None, 8192)              0

 dense (Dense)               (None, 1)                 8193

=================================================================================================
Total params: 4,361,985
Trainable params: 4,361,985
Non-trainable params: 0
```

Fig 3.9.2(1) Critic’s Architecture



```
Model: "generator"
_________________________________________________________________________________________________
 Layer (type)                             Output Shape (batch size, img height, img width, filters/units)   Param #
=================================================================================================
 input_1 (InputLayer)                     [(None, 256)]              0

 dense (Dense)                            (None, 32768)              8388608

 batch_normalization (BatchNormalization) (None, 32768)              131072

 leaky_re_lu (LeakyReLU)                  (None, 32768)              0

 reshape (Reshape)                        (None, 4, 4, 2048)         0

 up_sampling2d (UpSampling2D)             (None, 8, 8, 2048)         0

 conv2d (Conv2D)                          (None, 8, 8, 1024)         52428800

 batch_normalization_1 (BatchNormalization) (None, 8, 8, 1024)       4096

 leaky_re_lu_1 (LeakyReLU)                (None, 8, 8, 1024)         0

 up_sampling2d_1 (UpSampling2D)           (None, 16, 16, 1024)       0

 conv2d_1 (Conv2D)                        (None, 16, 16, 512)        13107200

 batch_normalization_2 (BatchNormalization) (None, 16, 16, 512)      2048

 leaky_re_lu_2 (LeakyReLU)                (None, 16, 16, 512)        0

 up_sampling2d_2 (UpSampling2D)           (None, 32, 32, 512)        0

 conv2d_2 (Conv2D)                        (None, 32, 32, 256)        3276800

 batch_normalization_3 (BatchNormalization) (None, 32, 32, 256)      1024

 leaky_re_lu_3 (LeakyReLU)                (None, 32, 32, 256)        0

 up_sampling2d_3 (UpSampling2D)           (None, 64, 64, 256)        0

 conv2d_3 (Conv2D)                        (None, 64, 64, 128)        819200

 batch_normalization_4 (BatchNormalization) (None, 64, 64, 128)      512

 leaky_re_lu_4 (LeakyReLU)                (None, 64, 64, 128)        0

 up_sampling2d_4 (UpSampling2D)           (None, 128, 128, 128)      0

 conv2d_4 (Conv2D)                        (None, 128, 128, 1)        1152

 batch_normalization_5 (BatchNormalization) (None, 128, 128, 1)      4

 Tanh (Activation)                        (None, 128, 128, 1)        0

=================================================================================================
Total params: 78,160,516
Trainable params: 78,091,138
Non-trainable params: 69,378
```

Fig 3.9.2(2) Generator's Architecture

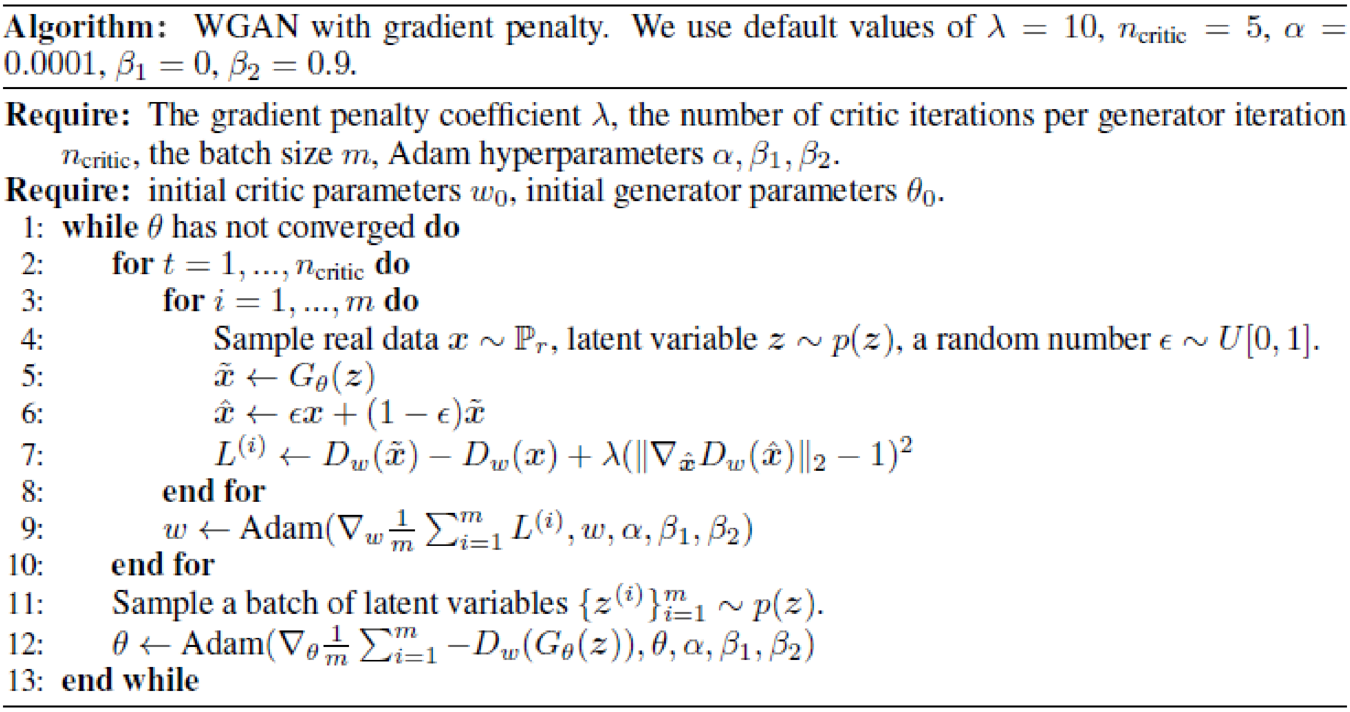
3.9.3 Training WGAN-GP:

WGAN-GP was trained separately for each minority class (only using the train images) so that it can learn the distribution of images and focus on unique characteristics in each class separately and does not mix features corresponding to different classes in the synthetic images. WGAN-GP was not trained for majority class i.e., “No Impairment” (2560 train images) because the goal was to oversample the minority classes to be of the same size as of the majority class using synthetic images.

Batch size of 64 was used for “Very Mild Impairment” (1792 train images) and “Mild Impairment” (717 train images) because it balances the trade-off between training stability and computational efficiency. With larger batch size, the training could be faster, but it may also lead to less accurate gradients due to the averaging of noise and the potential for vanishing gradients, thus, resulting in underfitting. Batch size of 8 was used for “Moderate Impairment” as it only had 52 train images. With a smaller batch size, the generator and critic networks were updated more frequently and with more variation in the input data, which would lead to a more robust and generalizable model.

The generator is supposed to synthesize MRIs of dimension 128 x 128 due to which Latent Random Vector (Gaussian Noise) of dimension 256 was chosen. This gave generator much more information to work with, allowing it to capture more complex features and generate high quality images. The generator and critic were compiled with Adam optimizer (lr = 0.0001, ß$_1$ = 0, ß$_2$ = 0.9) and their respective loss functions.

Since, both of them learn in an adversarial manner, they had to be trained using a custom training loop and a custom callback where the custom training loop trains the discriminator first but for each step of the generator, the critic has to be trained for 5 steps as explained previously in section 3.5 where the gradient penalty term was added to the critic’s loss but not to the generator’s loss.



```
**Algorithm:** WGAN with gradient penalty. We use default values of $\lambda = 10$, $n_{\text{critic}} = 5$, $\alpha = 0.0001$, $\beta_1 = 0$, $\beta_2 = 0.9$.
**Require:** The gradient penalty coefficient $\lambda$, the number of critic iterations per generator iteration $n_{\text{critic}}$, the batch size $m$, Adam hyperparameters $\alpha, \beta_1, \beta_2$.
**Require:** initial critic parameters $w_0$, initial generator parameters $\theta_0$.
while $\theta$ has not converged do
    for $t = 1, ..., n_{\text{critic}}$ do
        for $i = 1, ..., m$ do
            Sample real data $x \sim \mathbb{P}_r$, latent variable $z \sim p(z)$, a random number $\epsilon \sim U[0, 1]$.
            $\tilde{x} \leftarrow G_\theta(z)$
            $\hat{x} \leftarrow \epsilon x + (1 - \epsilon)\tilde{x}$
            $L^{(i)} \leftarrow D_w(\tilde{x}) - D_w(x) + \lambda(\|\nabla_{\hat{x}} D_w(\hat{x})\|_2 - 1)^2$
        end for
        $w \leftarrow \text{Adam}(\nabla_w \frac{1}{m} \sum_{i=1}^{m} L^{(i)}, w, \alpha, \beta_1, \beta_2)$
    end for
    Sample a batch of latent variables $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$.
    $\theta \leftarrow \text{Adam}(\nabla_\theta \frac{1}{m} \sum_{i=1}^{m} -D_w(G_\theta(z)), \theta, \alpha, \beta_1, \beta_2)$
end while
```

Fig 3.9.3(1) Algorithm for training WGAN-GP [63]

The custom callback does the following:

1. At the end of each epoch, it computes the FID Score [76] between a batch of randomly selected real images and a batch of synthetic images generated by the current generator for that epoch. Then it prints the FID Score for that epoch and appends current FID Score in the “***fid_scores***” list
2. If the FID Score for that epoch is less than 5 then it would save the current generator with the “***class_name”,*** “***current_epoch***” and “***current_fid***” in its name as a “.h5” file.

After training WGAN-GP for 2,000 epochs for each class, the latest epoch which corresponded to the minimum FID Score was computed and the corresponding generator i.e., the best generator so far throughout the training was loaded, to generate and visualize 64 synthetic images for that class. Finally, at the end of the training, FID Scores vs Epochs was plotted and WGAN-GP Loss vs Epochs were plotted by retrieving “***fid_scores***” list and WGAN-GP’s loss history from the custom callback.

3.9.4 Synthetic Image Quality Analysis:

Assessing the quality of synthetic images was crucial in order to ensure the validity of hypothesis verification. Without evaluating the quality of synthetic images, it is impossible to determine whether the hypothesis has failed due to incorrect assumptions or due to the low quality of images. To evaluate the quality of synthetic images; using the best generator for each class, the required number of synthetic images for matching the size of the majority class were generated and saved as ".*jpg*" files in their corresponding folders. Then FID score [76], SSIM [61], PSNR [61], SD [61], and Seaborn's Distplot [77] were computed between real and synthetic images for each class, and the results were documented and tabulated. If these metrics indicated that the quality is not good enough as discussed previously in Appendix B3, for some class, then WGAN-GP was re-trained for that particular class only.

3.10 Verifying The Hypothesis:

The synthetic images for each class were merged with the real ones to oversample and match the size of the majority class. The resulting distribution is illustrated in the following figure:

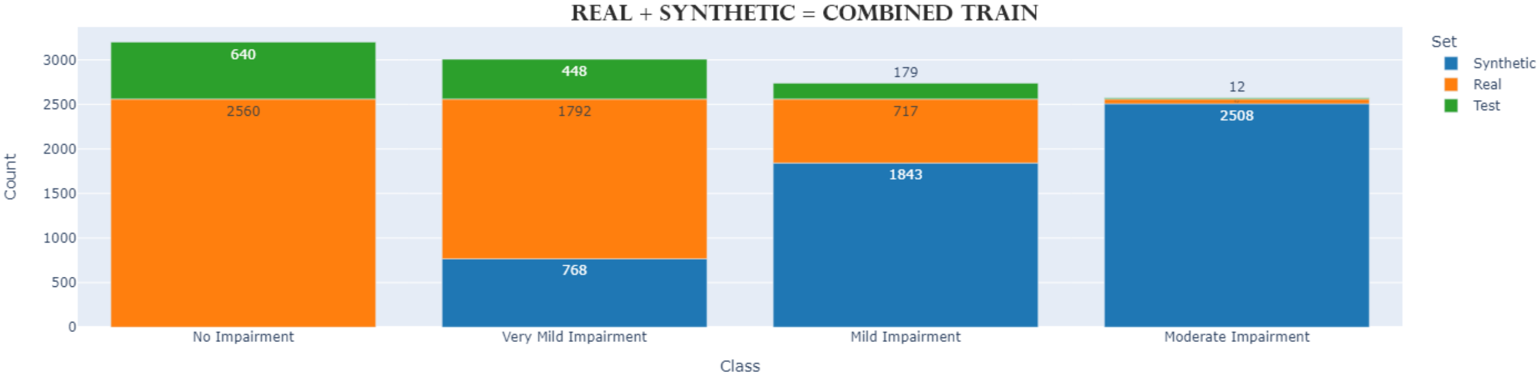


Fig 3.10(1) Image distribution after combining synthetic and real MRIs in train set (Real Moderate Impairment = 52)

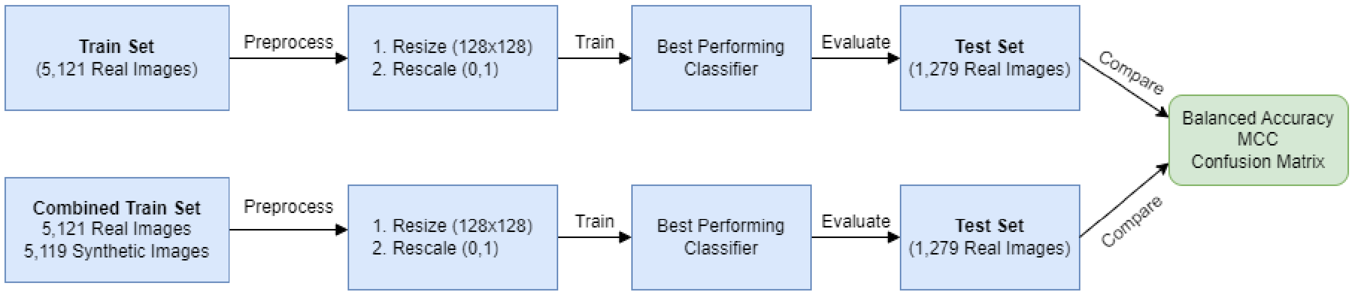


Fig 3.10(2) Workflow for Hypothesis Testing

The best performing classifier from section 3.8 was trained on this combined train dataset and was evaluated on the test dataset which had been kept fixed since the train-test split. Balanced Accuracy and Matthews Correlation Coefficient of this classifier were then compared with those of the best classifier from section 3.8 to assess overall changes in performance. Additionally, the performance on the minority classes was accessed using confusion matrices. These evaluations verified the hypothesis of this research.

If the overall performance and the performance on the minority classes is found to be improved then the combined dataset will be published on Kaggle. To further validate the effectiveness of this approach, the results were compared with those of conventional approaches, including Oversampling, Image Augmentation, SMOTE and cost-sensitive learning/class weighting. All of the models produced in this section were also saved as ".h5" files. Finally, the results for both testing the hypothesis and validating the effectiveness of this approach were tabulated and documented. Following parameters were used in Image Data Generator for Augmentation:

Parameters	Range/Value
rotation_range	20
brightness_range	[0.8,1.2]
zoom_range	[0.99, 1.01]
fill_mode	nearest
horizontal_flip	TRUE

Table 3.10(3) Parameters for Image Augmentation

4. Results and Discussion

4.1 Results for finding the best classifier

The overall performance of seven different models for classifying Alzheimer's MRIs was evaluated based on their balanced accuracies (BA) [73] and Matthew's Correlation Coefficients (MCC) [74]. The models compared were Custom CNN, DenseNet169 [35], VGG19 [37], Xception [69], InceptionV3 [68], EfficientNetB3 [70], and ResNet50 [45]. Table 4.1(1) summarizes their main results and these were the main highlights:

- Among these models, the Custom CNN achieved the highest balanced accuracy of 87.04% and MCC of 82.61%, outperforming all other models. The DenseNet169 model achieved the second-highest balanced accuracy of 82.13%, but its MCC was comparatively lower at 67.11%.
- The VGG19 model obtained a balanced accuracy of 74.45% and an MCC of 58.85%, while the Xception model showed a balanced accuracy of 64.33% and an MCC of 56.02%. The InceptionV3 model had a balanced accuracy of 59.66% and an MCC of 45.81%.
- The performance of the EfficientNetB3 and ResNet50 models was found to be inferior to that of the other models, with a balanced accuracy of 31.48% and 28.31%, respectively, and MCCs of 13.21% and 15.52%, respectively.

Overall, the Custom CNN model outperformed all pretrained models and demonstrated the best performance in terms of both BA and MCC, while the EfficientNetB3 and ResNet50 models showed the poorest performance.

Model	Balanced Accuracy (%)	Matthews Correlation Coefficient (%)
Custom CNN	87.04	82.61
DenseNet169	82.13	67.11
VGG19	74.45	58.85
Xception	64.33	56.02
InceptionV3	59.66	45.81
EfficientNetB3	31.48	13.21
ResNet50	28.31	15.52

Table 4.1(1)

4.2 Discussion for finding the best classifier

The results indicated that the Custom CNN model outperformed all pretrained models. While transfer learning is often suggested as a viable approach for dealing with limited dataset sizes, allowing pre-trained models to be fine-tuned for better generalization on smaller datasets, this study did not find this to be the case. Even though the DenseNet169 model showed decent performance which was similar to that of custom CNN in terms of BA (82.13% vs 87.04% respectively), its MCC was significantly lower than that of custom CNN (67.11% vs 82.61% respectively). There can be two possible reasons for this:

1. The dataset was heavily imbalanced and even though Balanced Accuracy (BA) focusses on both positives as well as the negatives (unlike the F1 Score), it can still result in a high accuracy by simply having a higher score for the majority class. MCC, on the other hand, took into account the true positives, true negatives, false positives, and false negatives of all four classes and provided a more balanced measure of performance which made it a more sensitive metric as small differences in performance can be more pronounced in MCC than in BA.
2. All of the pretrained models used "ImageNet" weights and were too complex for this dataset due to which they could not generalise well on MRI data and ended up overfitting. On the other hand, the custom CNN was not a very complex architecture and was trained from scratch to only learn patterns specific to this use case.

4.3 Results for analysing the quality of synthetic images

In this study, the minority classes in the Kaggle Alzheimer's dataset i.e., "Very Mild Impairment (VMI)", "Mild Impairment (MI)" and "Moderate Impairment (MoI)" were oversampled with synthetic images generated by WGANs-GP to be of the same size as of the majority class "No Impairment" (2560). The quality of these images was evaluated and analysed using metrics like FID Score (the closer to 0 the better) [76], SSIM index (the closer to 1 the better) [61], PSNR (ideally between 30 dB to 50 dB) [61] and SD (the closer to 0 the better) [61]. The results are summarized in Table 4.3(1).

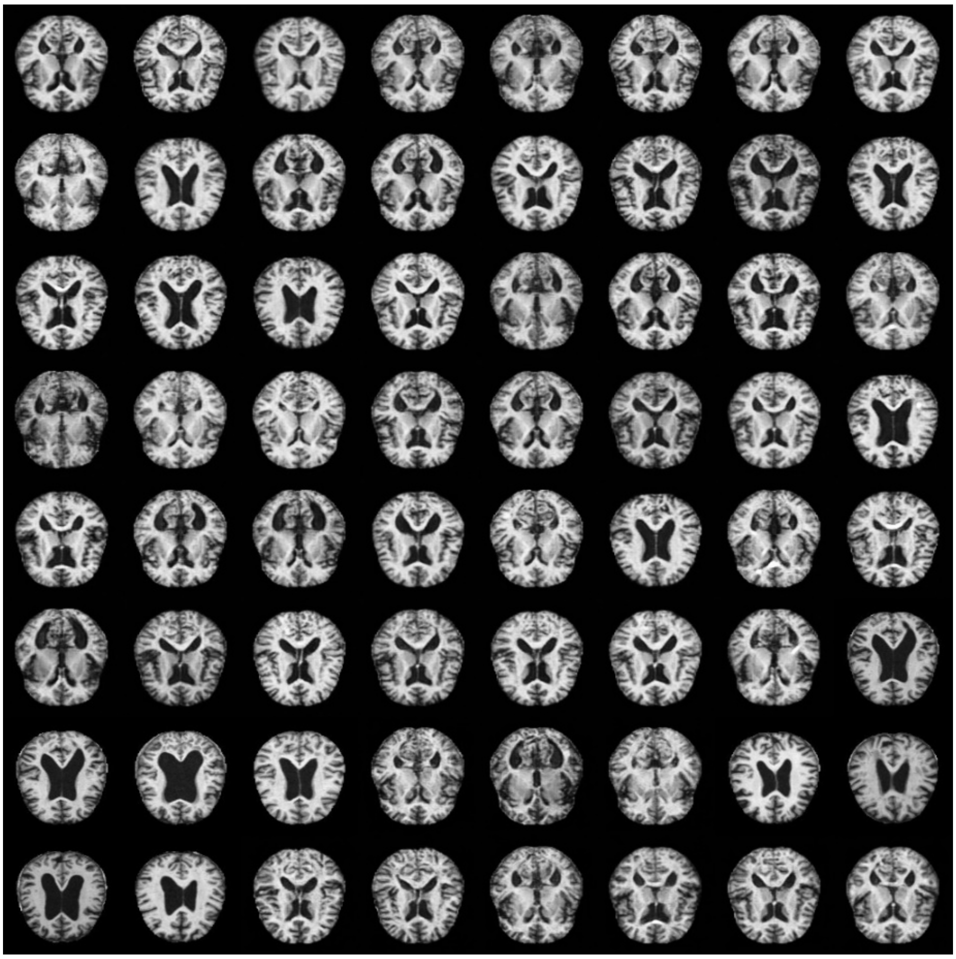


Fig 4.3(1): Samples of synthetic Very Mild Impairment MRIs

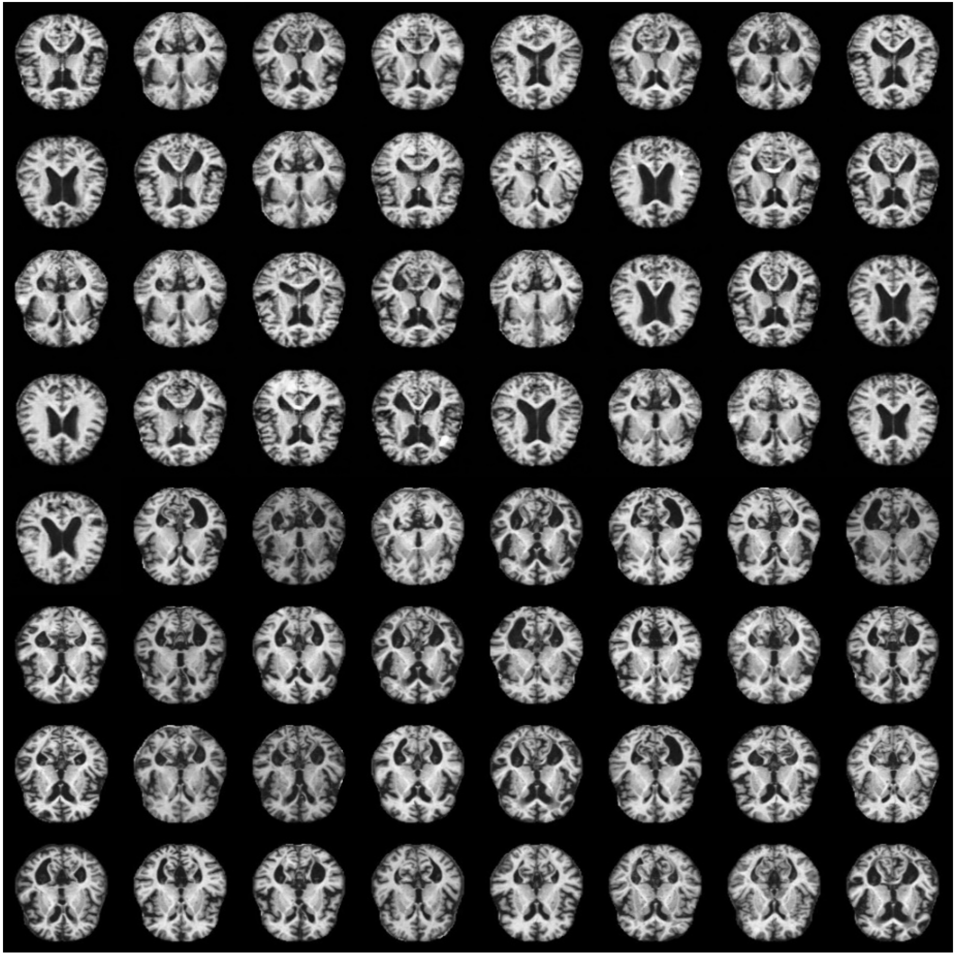


Fig 4.3(2): Samples of synthetic Mild Impairment MRIs

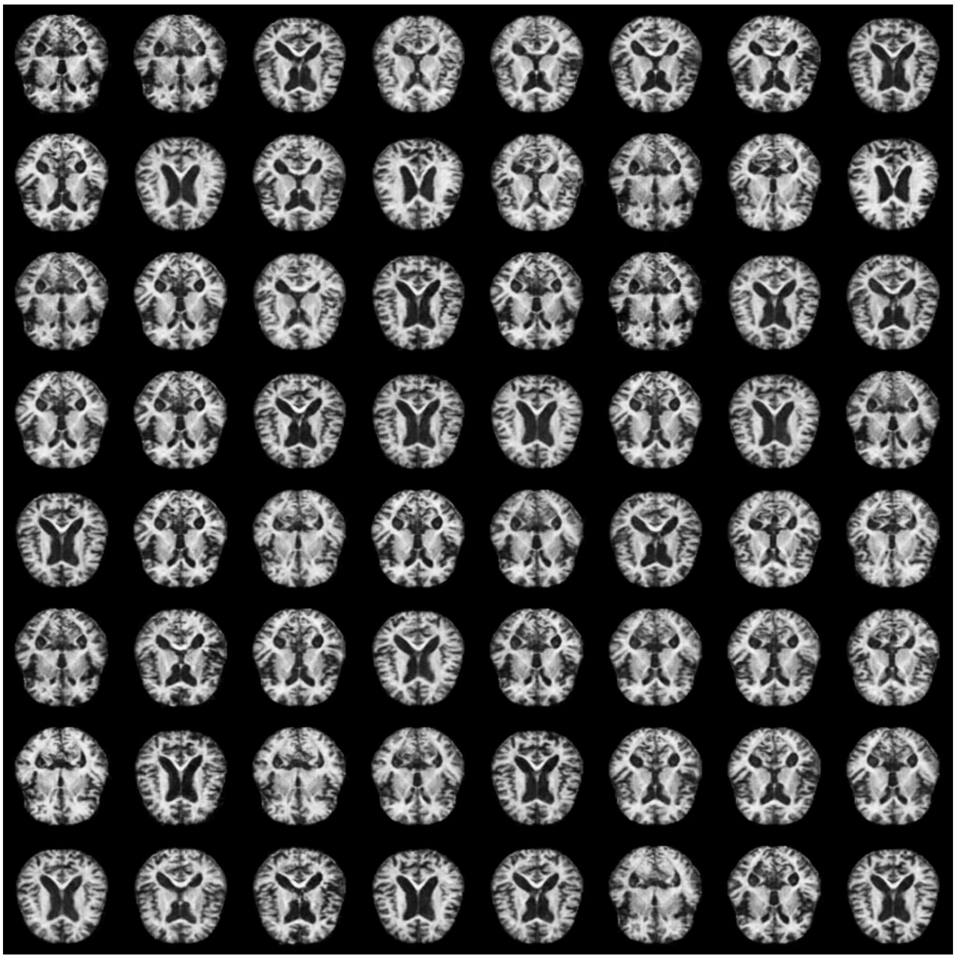


Fig 4.3(3): Samples of synthetic Moderate Impairment MRIs

	FID	SSIM	PSNR	SD	Synthetic MRIs	Real MRIs	Real + Synthetic
Very Mild Impairment (VMI)	0.07	0.97	33.39	0.05	768	1792	2560
Mild Impairment (MI)	0.15	0.96	29.93	0.07	1843	717	2560
Moderate Impairment (MoI)	0.17	0.98	32.59	0.00	2508	52	2560

Table 4.3(1)

Unlike the FID Score which can be computed over the entire batch of all real and synthetic images at once, the SSIM, PSNR an SD could only be computed between an image and its reference image. Since the size of real MRIs was not the same as of the synthetic ones, there was not a reference real MRI for each and every synthetic MRI. Therefore, SSIM, PSNR an SD were computed between the mean of real and the mean of synthetic MRIs for each class.

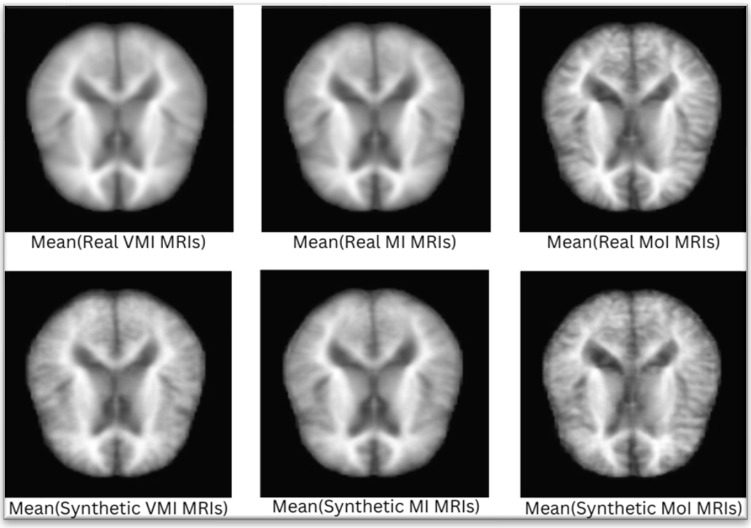


Fig 4.3(4) Mean of Real vs Synthetic MRIs for each minority class.

Additionally, Seaborn's Distplot was used to perform a qualitative analysis and comparison of the distribution of pixels, including range, shape, and central tendency, between real and synthetic MRIs for each minority class. The subsequent figures explicitly illustrate that the distribution of real and synthetic MRIs for each minority class is overlapping (purple colour indicates overlap as blue + red = purple):

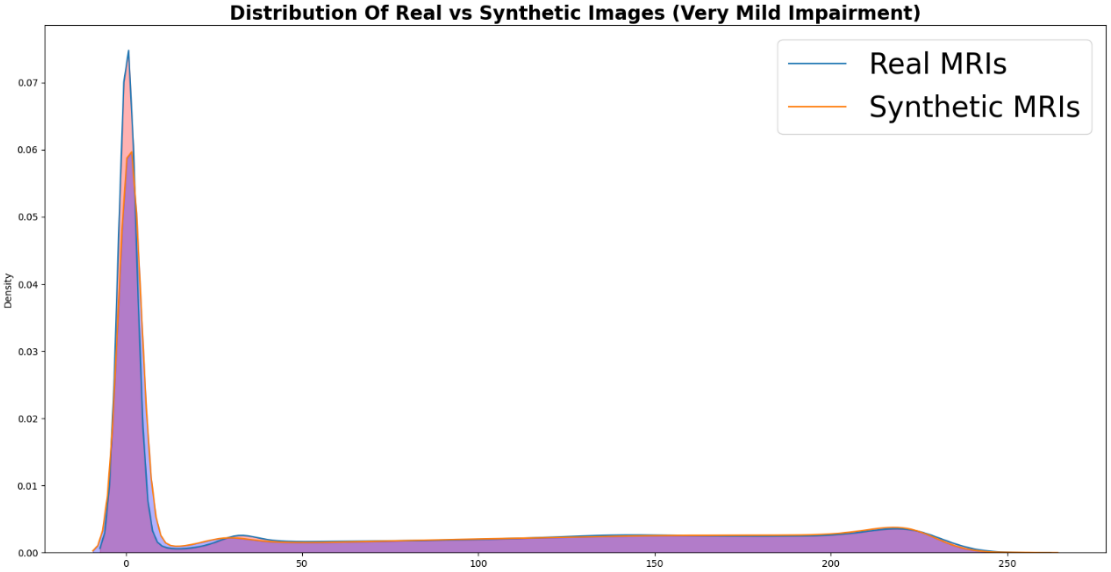


Fig 4.3(5) Distribution of pixels in Real and Synthetic Images for VMI class

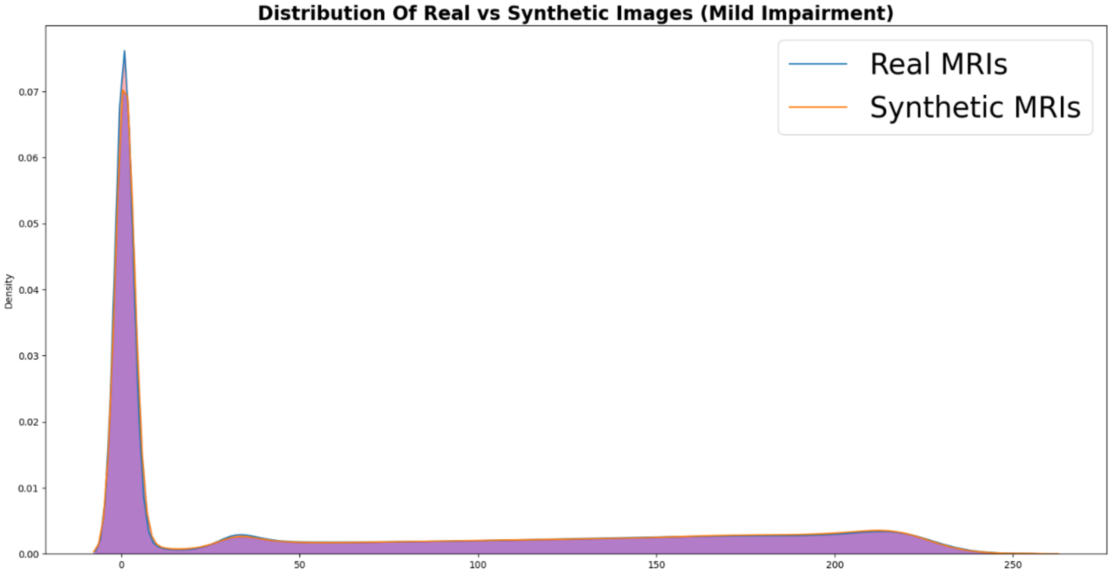


Fig 4.3(6) Distribution of pixels in Real and Synthetic Images for MI class

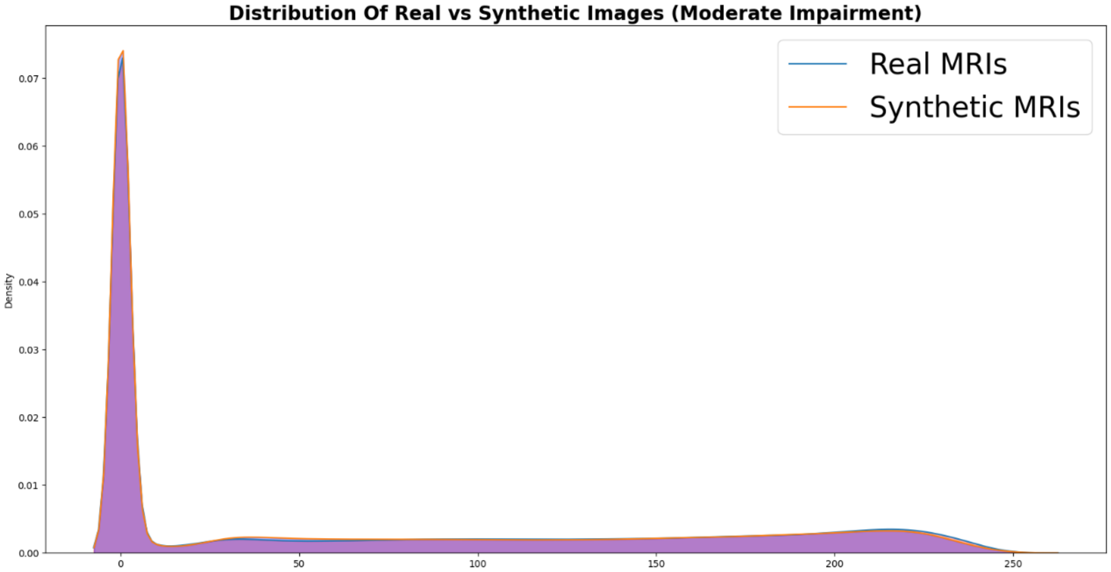


Fig 4.3(7) Distribution of pixels in Real and Synthetic Images for MoI class

4.4 Discussion for analysing the quality of synthetic images

This section appreciates the quality of the final product of this research i.e., the synthetic Alzheimer’s MRIs dataset. FID Score is a statistical metric which is influenced by the number of images it is computed on. This is because it uses mean and covariances of feature embeddings of images, both of which are influenced by the number of samples (higher the number of samples the more precise are the means and covariances). This is why during training of WGAN-GP, since the FID Score was calculated only for a batch of 64 images for “Very Mild Impairment” and “Mild Impairment” class and a batch of 8 images for “Moderate Impairment” class it resulted in FID Score of 4-5 but when the FID Score was computed over hundreds and thousands of images as shown in Table 4.3(1), it resulted in a mean FID Score of 0.13 (over all three minority classes) which is very close to zero (ideal value). This indicates that both the quality and diversity of synthetic MRIs were as good as the real ones.

The set of mean real and mean synthetic MRIs for each minority class as shown in figure 4.3(1) look very similar to each other. These set of images were used to calculate SSIM, PSNR and SD. The mean SSIM of 0.97 indicate that the real and synthetic MRIs are perceptually very similar to each other in terms of luminance, contrast and overall structure. The mean PSNR of 32 dB which is well between its ideal range of 30 dB and 50 dB indicates that the synthetic MRIs were as good as the real ones and the mean Sharpness Difference of 0.04 also indicates that there was almost no loss in sharpness during generation of synthetic images.

Finally, the results of all these metrics over the entire batch justifies that the synthetic MRIs were as good as the real ones but not completely identical which proves that they are indeed new images which satisfies the goal of synthetic image generation. Furthermore, Seaborn’s Distplots in figures 4.3(5), 4.3(6) and 4.3(7) clearly illustrate that the distribution of synthetic MRIs for each minority class was almost completely overlapping with the distribution of Real MRIs for those particular classes which means that the synthetic MRIs were as diverse as the images in the original Kaggle Alzheimer’s dataset. This explains why the mean FID Score over all three minority classes was so close to its ideal value of 0.

Now, since the quality of synthetic dataset is found to be good enough. This allows the researcher to verify the hypothesis in a fair and transparent manner.

4.5 Verifying the hypothesis

Since Custom CNN was found to be the best so far amongst all other classifiers, it was used to verify the hypothesis of this research. The performance of Custom CNN was compared when it was trained on Train Set of 5,121 MRIs vs Combined Train Set of 5,121 Real and 5,119 Synthetic MRIs. The results for overall performance in terms of Balanced Accuracy and Matthew’s Correlation Coefficient are summarised in Table 4.5(1).

Custom CNN		Balanced Accuracy (%)	MCC (%)
	(5121 Real MRIs) Train Set	87.04	82.61
	(5121 Real + 5119 Synthetic MRIs) Train Set	98.81	97.56

Table 4.5(1)

These results clearly show that there was 11.77% increase in Balanced Accuracy and approximately 15% increase in MCC both of which clearly indicate significant improvement in overall performance.

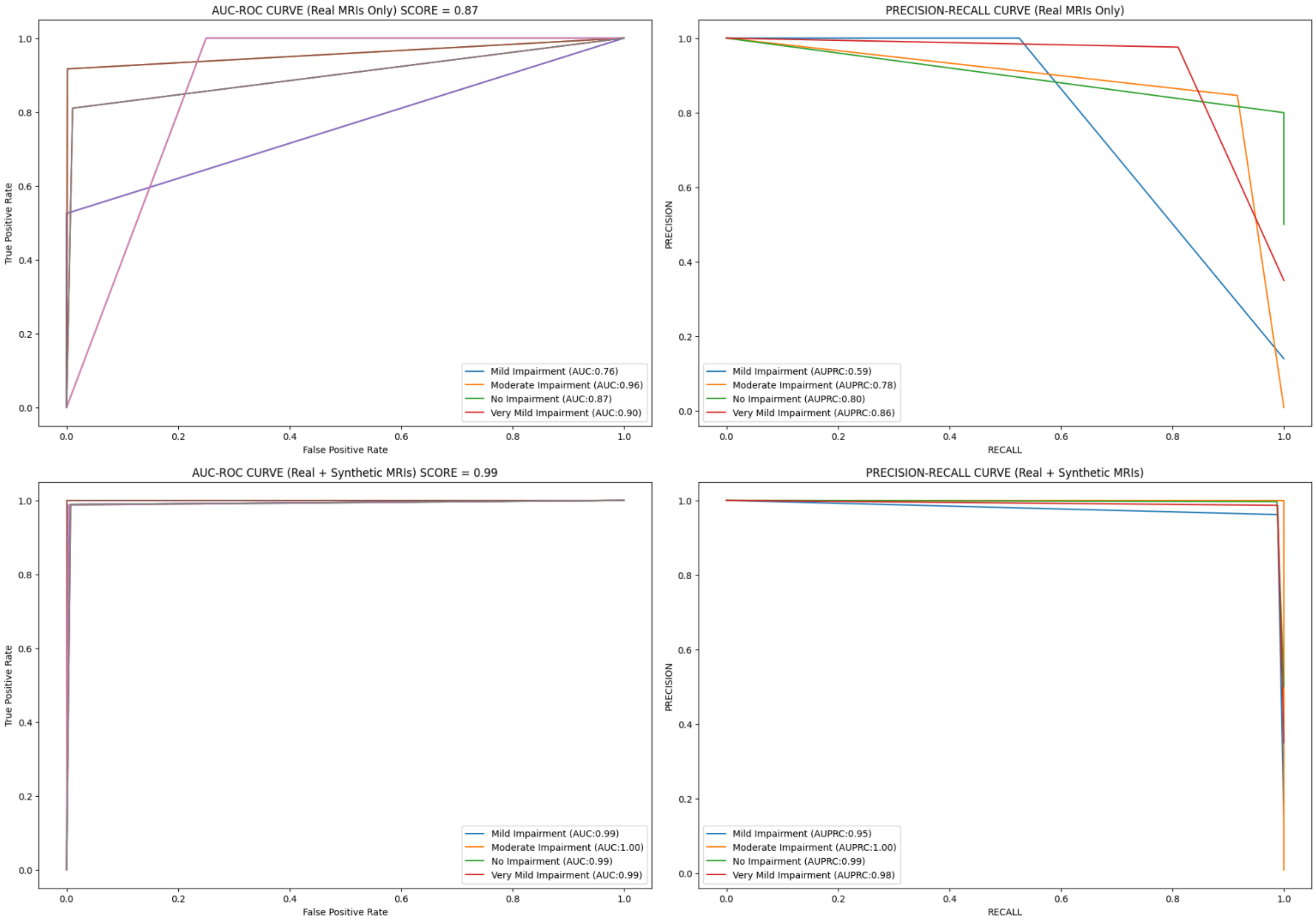


Fig 4.5(1) AUC-ROC Curves and PR Curves with AUPRC for both Real and Real + Synthetic MRIs

AUC-ROC Curves:
Custom CNN when trained on Real MRIs only, resulted in AUC scores of 0.87, 0.9, 0.76 and 0.96 for No Impairment, Very Mild Impairment, Mild Impairment and Moderate Impairment respectively and an overall ROC-AUC score of 0.87. Whereas, Custom CNN when trained on combined train set of Real and Synthetic MRIs resulted in AUC scores of 0.99, 1.0, 0.99 and 0.99 for No Impairment, Very Mild Impairment, Mild Impairment and Moderate Impairment respectively and an overall ROC-AUC Score of 0.99.

PR Curves:
Custom CNN when trained on Real MRIs only, resulted in AUPRC scores of 0.8, 0.86, 0.59 and 0.78 for No Impairment, Very Mild Impairment, Mild Impairment and Moderate Impairment respectively. Whereas, Custom CNN when trained on combined train set of Real and Synthetic MRIs resulted in AUPRC scores of 0.99, 0.98, 0.95 and 1.0 for No Impairment, Very Mild Impairment, Mild Impairment and Moderate Impairment respectively.

This verifies, as discussed previously in Appendix A4, that PR Curves are more sensitive to class imbalance than AUC-ROC curves. But both of these curves don't explicitly represent the number of misclassifications on each minority class and the overall improvement in performance on the minority classes which is why the performance on the minority classes was compared using Confusion Matrix which is illustrated in Figure 4.5(2) and Figure 4.5(3).

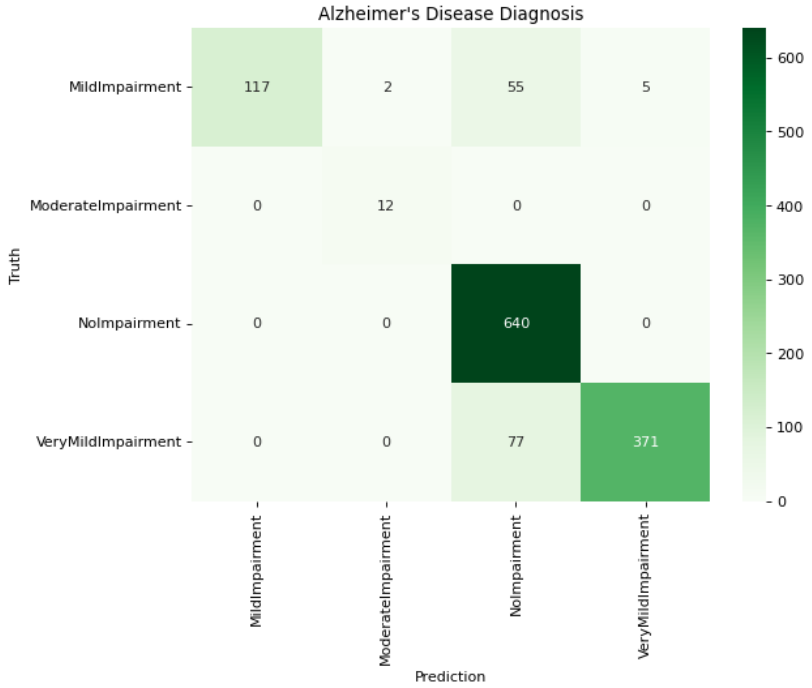


Fig 4.5(2) Confusion Matrix for Custom CNN trained only on Real MRIs

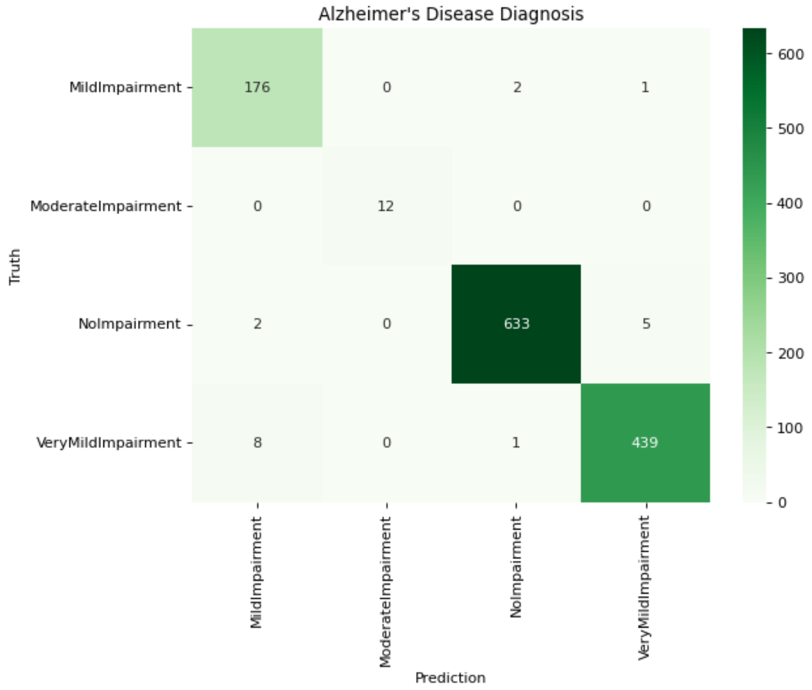


Fig 4.5(3) Confusion Matrix for Custom CNN trained on Real + Synthetic MRIs

These confusion matrices above clearly illustrate that when custom CNN was trained just using the real MRIs it made no False Negatives (misclassifications) on the “No Impairment” (Majority Class), 77 False Negatives on “Very Mild Impairment”, 62 False Negatives on “Mild Impairment” and no False Negatives on “Moderate Impairment”. But when the same custom CNN was trained on the combined train set of real and synthetic MRIs, it made 7 False Negatives on “No Impairment”, 9 False Negatives on “Very Mild Impairment”, 3 False Negatives on “Mild Impairment” and no False Negatives on “Moderate Impairment” classes. This indicates 88.31% improvement on “Very Mild Impairment” class and 95.45% improvement on the “Mild Impairment” class. This means an overall 91.4% improvement on the minority classes at the cost of 1% reduction in performance on the majority class. Therefore, the overall 11.77% increase in Balanced Accuracy, 15% increase in MCC and 91.4% improvement on the minority classes proves that the hypothesis is correct and this approach has successfully addressed the class imbalance problem in the Kaggle Alzheimer’s dataset.

4.6 Discussion for verifying the hypothesis

In section 4.5 of the present research, the results clearly demonstrate that the hypothesis is correct and the research has been successful, with an overall improvement of 11.77% in Balanced Accuracy and an 15% improvement in Matthew's Correlation Coefficient. As previously discussed in the same section, the confusion matrices depicted in figure 4.5(2) and fig 4.5(3) indicate a 91.4% improvement in the minority classes, at the expense of a 1% decrease in performance on the majority class, i.e. "No Impairment". However, this is not an issue since misclassifying a person with "Very Mild Impairment" or "Mild Impairment" as "No Impairment" is much more undesirable than misclassifying someone with "No Impairment" as Mild or Very Mild Impaired. The credit goes to WGAN-GP for overcoming mode collapse and generating synthetic images which not only successfully imitated the real MRIs in terms of quality and diversity, but also provided the custom CNN with new patterns to learn from. This resulted in improved overall performance and effectively addressed the class imbalance problem in the Kaggle Alzheimer's Dataset. Thus, the credibility of the synthetic dataset has been established, which gives a green signal for it to be published [Link].

4.7 Analysing the effectiveness of this methodology

To further analyse the effectiveness of this approach, the results of this approach were compared with the results of conventional approaches of Oversampling, Image Augmentation, SMOTE, and Cost Sensitive Learning or Class Weighting. The results of these experiments are summarized in Fig 4.7(1) and the main highlights are the following:

- The Synthetic Images approach achieved the best performance, with a balanced accuracy of 98.81% and a Matthew’s correlation coefficient of 97.56%.
- SMOTE also performed well, with a balanced accuracy of 93.85% and a Matthews correlation coefficient of 90.72%.
- Regular Oversampling and Class Weighting or Cost Sensitive Learning had similar performances, with balanced accuracies of 93.73% and 94.29%, respectively, but lower Matthew’s correlation coefficients of 85.18% and 84.05%, respectively.
- The Augmentation approach had the lowest performance among the evaluated approaches, with a balanced accuracy of 52.88% and a Matthews correlation coefficient of 44.16%.

These results show that machine learning approaches can significantly improve the performance of classification models on imbalanced datasets. Overall, the evaluation results suggest that the Synthetic Images approach is the most effective and superior approach followed by SMOTE for addressing class imbalance for this use case. Although not as effective as the prior two, Regular Oversampling and Class Weighting can also provide reasonable results. However, Augmentation was not found suitable for this dataset, at least not without careful selection of the augmentation techniques and parameters. In conclusion, the choice of the most suitable approach depends on the specific characteristics of the dataset and the classification task at hand.

Approach	Balanced Accuracy (%)	Matthews Correlation Coefficient (%)
Synthetic Images	98.81	97.56
SMOTE	93.85	90.72
Regular Oversampling	93.73	85.18
Class Weighting / Cost Sensitive Learning	94.29	84.05
Image Augmentation	52.88	44.16

Fig 4.7(1) Comparison of our approach with conventional approaches

4.8 Discussion for analysing the effectiveness of this methodology

In Section 4.7, the study concluded that oversampling with synthetic images generated by WGAN-GP was the most effective method for addressing class imbalance. While SMOTE generates synthetic samples by interpolating between existing samples of the minority class and has been effective, it has limitations such as generating samples that are too similar to existing ones, leading to overfitting and reduced generalization performance, and not capturing the underlying data distribution if the minority class is complex or has high variability. In contrast, synthetic images generated by WGAN-GP were more diverse and realistic, capturing the underlying data distribution of the minority class. This increased diversity enabled the model to learn more robustly which can generalise the new patient’s unseen data better, thus resulting in improved performance. Generating synthetic MRIs is also better than any other technique because it produces samples that are not direct copies of existing ones, increasing variation and diversity in the train set.

Although regular oversampling improved overall performance, it did not perform as well as synthetic images and SMOTE because it did not improve image diversity, as it simply replicated existing samples, which reduced its ability to generalise. Class weighting, which assigned higher weights and penalized mistakes on minority class samples more, improved overall performance similar to regular oversampling but could not perform as well as synthetic images and SMOTE because it did not create new samples or capture the underlying data distribution of the minority class.

Image augmentation parameters, as described in methodology chapter, was the least effective approach amongst all approaches and resulted in even poorer performance than the baseline custom CNN model, possibly because the augmented images introduced too much noise or variability in the training data, making it more difficult for the model to learn underlying patterns and generalize. Additionally, the augmentation techniques used may not have been appropriate for the specific dataset or task, introducing unrealistic or irrelevant variations that negatively impacted the model's performance. Finally, it is possible that the model architecture or hyperparameters were not suitable for the augmented data and could not effectively learn from the augmented samples.

4.9 Comparison of results with other GAN based approaches

Roy et al. [83] utilized federated GAN-based biomedical image augmentation and classification to tackle class imbalance in the Kaggle Alzheimer's dataset. Their approach achieved a significant improvement in accuracy, increasing it from 86% to 97.81%. For image generation, they used DC-GAN, and for classification, they used VGG16, both with fed-focal loss for imbalanced data instead of binary cross entropy. Similarly, Jain et al. [84] improved accuracy from 74% to 87% on the same dataset using their own custom CNN and DC-GANs. Datta et al. [85] also implemented DC-GANs to address class imbalance in the Kaggle Alzheimer's dataset and improved the balanced accuracy from 57.5% to 79%. Lastly, Mukherjee et al. [59] employed DC-GANs using the same dataset and improved accuracy from 69% to 82% with ResNet50. By achieving an increase in balanced accuracy from 87% to 98.81%, this approach exhibits the superiority of both its methodology and final product over other DC-GAN-based methods that have been utilized on the same dataset. While many of these other approaches have reported their classification results, they have neglected to include any analysis of image quality within their research.

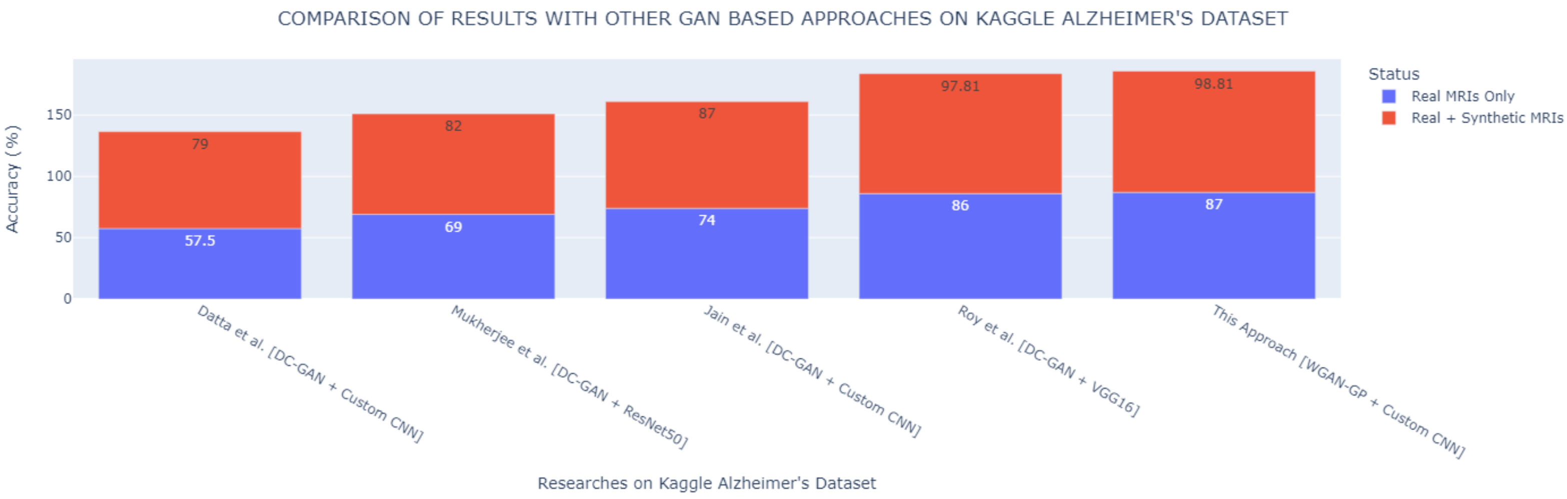


Fig 4.9(1) Comparison of results with other GAN based approaches on Kaggle Alzheimer’s Dataset

4.10 GRAD-CAMs [14] (A cherry on top)

Since, the research approach has been successful in addressing class imbalance and the final product has been found credible, the best performing classifier so far i.e., custom CNN with BA of 98.8% and MCC of 97.56% was employed to implement grad cams.

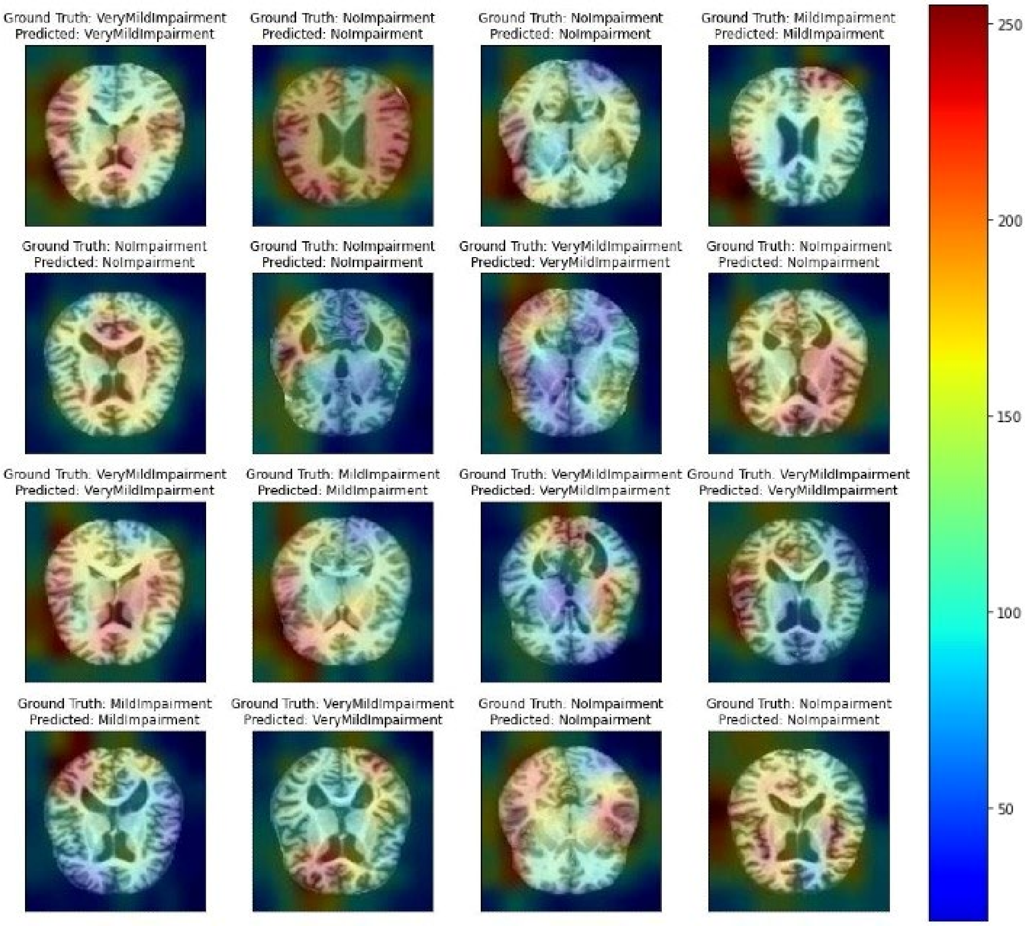

Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: MildImpairment
Predicted: MildImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: MildImpairment
Predicted: MildImpairment
Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: MildImpairment
Predicted: MildImpairment
Ground Truth: VeryMildImpairment
Predicted: VeryMildImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
Ground Truth: NoImpairment
Predicted: NoImpairment
250
200
150
100
50

Fig 4.10(1) GRAD CAMs

This was done by extracting the gradients from the last convolutional layer, for each image, which were then overlapped on top of that particular image in form of a heatmap to visualize which parts of the image did the classifier focussed on, in order to make its classification decisions.

Figure 4.10(1) clearly illustrates the regions responsible for classification in a hierarchy using a colour-bar in which “red” indicates the region where the classifier focused the most followed by “orange”, “yellow”, “green” and finally “blue” indicating the regions where it focussed the least. These images were classified with 100% accuracy which can also be seen by the ground truth and the predicted label on top of each image.

Researchers and medical professionals can benefit from using grad-cams (explainable AI) to identify the indicators of Alzheimer’s for each stage. Additionally, in the future, if an application is developed that automates the diagnosis of Alzheimer’s disease, clinicians could then utilize the grad-cam feature in the application to interpret and confirm the classification decisions for each person's MRI.

5. Evaluation

5.1 Introduction

This chapter assesses the results of the research study and presents a personal reflection of the researcher's experience throughout the process.

5.2 Evaluation of main research approach and findings

The ongoing research has revealed that the biggest issue in diagnosing Alzheimer's is the scarcity of datasets and class imbalance in those datasets. The literature review explains that many researchers have attempted binary classification, but this is not appropriate for medical datasets, making multiclass classification essential.

To address the issue of class imbalance, many researchers have utilized machine learning techniques, such as oversampling, SMOTE, cost-sensitive learning, and image augmentation. In addition, transfer learning has been proposed as a potential solution. Some researchers have also used generative algorithms, such as GANs [56], Cycle GANs [78], and DC-GANs [58], to generate synthetic images for either enhancing image quality or synthesizing Deep Fake images to solve class imbalance. Many of the studies reviewed did not provide clear information on whether they trained their GAN models for each individual class or not, nor did they specify whether images from the test dataset were included in training their GAN. These studies also neglected to use metrics that are specifically designed for imbalanced datasets, nor did they compare their results to those of conventional approaches. Additionally, most of the studies utilized DC-GANs for image synthesis, but only trained them on a small number of images and that too for less than 1,000 epochs. Lastly, some studies failed to report any measures of image quality for their synthetic images.

This study found that transfer learning was not a viable approach for classifying images in Kaggle Alzheimer's dataset because the "*ImageNet*" weights made it hard for pre-trained models to generalize on MRI scans for Alzheimer's, leading to overfitting. Instead, custom CNN outperformed all other pretrained models because it learned patterns specific to this use case.

The initial plan was to go with DC-GANs for synthesizing MRIs, but the study found that they suffer from mode collapse, which limited researchers to work with just a few hundred images, that too for limited epochs for which they can be trained. In contrast, WGAN-GP [63] overcomes this issue of mode collapse, providing a more stable training process thus being less sensitive to model architecture and hyperparameter configurations. Through experimentation, the study found that batch normalization layers in the generator and dropout layers in the critic can make the WGAN-GP converge faster and improve the quality of synthetic images.

Unlike most research studies that did not use synthetic image quality metrics, this study found appropriate metrics through literature, not only for analysing the quality of synthetic images but also for evaluating classifiers on heavily imbalanced datasets. The study found that Matthew's Correlation Coefficient provided a more balanced measure of performance by taking all four categories in the confusion matrix into consideration, making it a more sensitive metric as small differences in performance can be more pronounced in MCC than in Balanced Accuracy. Furthermore, the study found that FID score can be used not only to assess the quality and diversity of synthetic images but also to compare epochs with each other while training GAN, leading to identifying the epoch that corresponded to the best generator for that particular class.

However, training WGAN-GP posed many difficulties as it is a combination of two different architectures that have different loss functions and must be trained adversarially. Therefore, WGAN-GP had to be trained using a custom training loop, a custom callback, and a custom metric (FID). One of the major issues with training WGAN-GP was that it is computationally very expensive, and thus, it could not be trained on Kaggle GPU as Kaggle's kernel disconnects automatically after 12 hours of training. As a result, it had to be trained on Google Colab Pro+. Furthermore, saving the generator after each epoch was not feasible, as it would have accounted for 582 GB of space since each generator was of 298 MB. Therefore, to mitigate this, only those generators were saved for which FID Score was less than 5. This threshold was found appropriate for this use case by experimentation.

Finally, it was found that the research has been successful and it was also found that the oversampling with synthetic images is the most effective for addressing class imbalance problem as compared to oversampling, SMOTE, cost sensitive learning and image augmentation. Moreover, it was also found that image augmentation, instead of improving overall performance and solving class imbalance, ended up resulting in even poorer performance as compared to the baseline. This was because either because it introduced unrealistic or irrelevant variations that negatively impacted the model's performance in its ability to learn patterns and generalise or because custom CNN’s architecture or hyperparameters were not suitable for the augmented data and could not effectively learn from the augmented samples.

5.3 Ethical, Legal, Social, Professional and Security Issues

Ethical considerations are crucial in any research, weighing both benefits and drawbacks. While synthetic images generated by WGAN-GP were of high quality, researchers should be aware that conclusions derived from this synthetic dataset may not resemble the real world since these were not verified by a radiologist. This concern will be addressed by including a statement in the dataset description when the synthetic dataset is made public on Kaggle. Since, both Kaggle Alzheimer’s Dataset and the synthetic dataset did not resemble any human’s identity or their personal information, there were no privacy concerns. As the study did not involve human participation or personal information collection, it did not address social issues either. Furthermore, the study did not involve any unfair data use or processing and used an open-source dataset, therefore, there are no legal or security risks as no data is being collected from individuals and amyloid imaging is not restricted by GINA [78]

5.4 Personal Reflection

Using machine learning techniques to solve real-world problems is fascinating. Initially planning to use DC-GANs, research papers led to the conclusion that WGAN-GP should be used for synthetic image generation. Generating synthetic MRIs for Alzheimer's with WGAN-GP was both exciting and daunting, but provided ample opportunities to learn.

Over several months, the researcher immersed themselves in research papers, tried various approaches, and learned from the limitations of others' work. Countless hours were spent refining techniques, tweaking parameters, and analysing results. It was a steep learning curve, but the researcher persisted, knowing that the potential impact of this research could be significant. The project's success was a significant achievement because of its complexity and meticulousness. It involved writing and implementing a substantial amount of code from scratch, involving object-oriented programming.

Along the way, the researcher learned valuable lessons and skills that will help in future projects of this scale, including generating synthetic images which is very valuable as it leads to overcome the scarcity of medical data, increase accuracy and robustness of machine learning models, reduce research costs and time, that too without causing any privacy concerns. The researcher also learned the importance of understanding existing research limitations and creating something novel on top of it, as well as the value of persistence and project management.

In conclusion, the journey of learning a new skill and finding limitations in existing research was a challenging yet rewarding experience that helped the researcher grow as an individual. It left him with newfound confidence in his abilities and the knowledge that he can overcome challenges and succeed in future projects.

5.5 Conclusion

This project aimed at verifying the hypothesis by solving the class imbalance problem in the Kaggle Alzheimer’s Dataset [13] by oversampling minority classes using synthetic MRIs generated by WGAN-GP and evaluating the effectiveness of this approach by comparing its results with those of conventional approaches of addressing class imbalance.

The results of the study indicated that the Custom CNN model outperformed all pretrained models, achieving a Balanced Accuracy of 87.04% and Matthew’s Correlation Coefficient of 82.61%. The study also found that transfer learning may not always be a viable approach for dealing with limited dataset sizes that too with heavy class imbalance. This was because unlike the pretrained models, the custom CNN model was trained from scratch and it learned patterns specific to this use case which made it more effective in classifying Alzheimer's MRIs as compared to pretrained models. The complexity of pretrained models with “ImageNet” weights was found to be unsuitable for this dataset, as they resulted in overfitting. In contrast, the custom CNN used batch normalization layers, which prevented the weights from becoming too small and allowed the optimizer to update the weights properly, thereby preventing overfitting.

This study also concluded that the synthetic Alzheimer’s MRIs dataset generated using WGAN-GP was such high quality and diversity that they could be used as an effective substitute for real MRIs which overcomes the scarcity of medical data, increase accuracy and robustness of machine learning models, reduce costs and time, that too without causing any privacy concerns. The mean FID [73] Score of 0.13, a mean SSIM [61] of 0.97, a mean PSNR [61] of 32 dB, and a mean Sharpness Difference (SD) [61] of 0.04 proved that synthetic MRIs are of as good quality as the real ones and are indeed new images as all these results were very close to their ideal values. Furthermore, the Seaborn’s Distplot [77] also indicated that the diversity of the synthetic MRIs is very close to that of real MRIs.

This study also concluded that the research hypothesis was correct and successful, with an overall increase of 11.77% in Balanced Accuracy and an 15% increase in Matthew's Correlation Coefficient. The study also found a 91.4% improvement in the performance on minority classes, at the expense of a 1% reduction in performance in the majority class. This indicates that the custom CNN model developed in this study can effectively classify Alzheimer's MRIs, particularly in the minority classes.

Besides, this approach was found more effective than conventional approaches of addressing class imbalance and indicated that MCC is the best metric for evaluating performance on heavily imbalanced datasets as it provides a balanced measure of performance over all the classes which makes it a more sensitive metric to slight differences than Balanced Accuracy. Additionally, the study also found that image augmentation was not suitable for this use case as it introduced unrealistic or irrelevant variations that negatively impacted the model's performance in its ability to learn patterns and generalise.

Finally, this study introduced GRAD-CAMs [14], which could help clinicians make better diagnoses and interpret the indicators for Alzheimer's for each stage. In conclusion, the findings of this study have significant implications for the development of effective classifiers for Alzheimer's MRIs.

5.6 Limitations

While the results of the study showed promising improvements in the performance of the model, there are several limitations that must be considered. Custom CNN's (98.8% BA and 97.56% MCC) ability to generalize to other Alzheimer's MRI datasets with different characteristics and distributions may be limited. The approach employed in this project, which involves generating synthetic MRIs, can also be computationally expensive and time-consuming, making it difficult to scale to larger datasets or real-world scenarios. Furthermore, the generated data may not accurately represent the full range of variability seen in real patient data, which could potentially impact the quality of the diagnosis and treatment decisions based on the model's outputs. In addition, the use of deep learning models for medical diagnosis and treatment can be challenging to interpret, making it difficult for clinicians to understand how the model arrived at its decisions. Although, the study introduced GRAD-CAMs to help clinicians make better diagnoses and interpret indicators for Alzheimer's, there may be limitations to the interpretability of the model. Additionally, the dataset used in the study may have certain biases that are not representative of the broader population. Therefore, the model's performance may be limited when applied to more diverse datasets or patient populations. It is also important to note that the dataset used in the study only contained axial MRI scans, while a full 3D MRI scan consists of axial, coronal, and sagittal parts of the brain. The lack of data from these other planes could potentially limit the model's ability to accurately diagnose and classify Alzheimer's disease. Finally, the study did not evaluate the robustness of the model to various types of noise and adversarial attacks, which could potentially compromise the accuracy of the diagnosis and treatment decisions.

5.7 Future Work

The study has identified several potential areas for future work that could expand upon the findings of this research. To evaluate the robustness of the model to various types of noise and adversarial attacks, researchers could test the model's performance when presented with noisy or altered images, or purposely crafted adversarial examples intended to deceive the model. This would provide a better understanding of the model's limitations and potential ways to enhance its performance in real-world settings. Furthermore, future work could extend the approach to 3D MRI scans, which encompass axial, coronal, and sagittal parts of the brain, as the current study only employed axial MRIs. By doing so, researchers could assess the effectiveness of the approach on a wider range of data and improve the accuracy of the diagnosis. Additionally, instead of only using MRI scans, multimodal features such as Age, Gender, PET scans, MRI scans, and cerebrospinal fluid data could be combined to train the classifiers. This would make the model more robust and generalize better [43]. Furthermore, attention layers can be added in the generator of the WGAN-GP to focus more on the low-level semantics which will further enhance the synthetic MRIs. Finally, by addressing these limitations, the diagnosis of Alzheimer's disease could be automated, and a WebApp or API could be developed to aid clinicians in making better-informed diagnoses on a larger scale.

Appendix:

A1. Convolutional Neural Networks (CNNs) [79]

Neurons are the building blocks of neural networks. They receive input data ($\boldsymbol{x_i}$) and apply fixed weights ($\boldsymbol{w_i}$) to amplify or suppress the input based on its significance. The output then passes through a non-linear activation function $\boldsymbol{f(\sum w_i x_i)}$. These neurons are organized into layers to form a Neural Network, and stacking three or more of these layers creates Deep Neural Networks (DNNs) that can extract complex features from large, non-linear datasets for data classification. The optimal weight matrices for the neurons are determined using an optimization algorithm that minimizes the loss function with respect to input data. Training is conducted for a fixed number of epochs during which the loss function is gradually reduced towards zero.

Convolutional Neural Networks (CNNs) have revolutionized computer vision, achieving remarkable success in image recognition, classification, segmentation, and object detection. By passing images through a sequence of input layer, convolutional layers, activation functions, batch normalization layers, pooling layers, fully connected dense layers, and output layer, CNNs effectively extract valuable features from images, enabling more accurate and efficient analysis of visual data.

1. Input layer: It is responsible for pre-processing the input data into a 3D matrix compatible with subsequent layers. This includes reshaping the data based on the image's width, height, and colour channel depth without modifying the data, and passing it on to the following layers for feature extraction and classification. As the quality of the input data is crucial for the CNN's effectiveness, the input layer plays a critical role in ensuring that the data is correctly pre-processed for optimal performance.

2. Convolutional Layer: They are responsible for extracting features from the input image by applying learnable filters called convolutional kernels. These kernels highlight different patterns and produce feature maps. These layers recognize various shapes, edges, and textures in the image, making it possible for the network to learn complex representations of the input data. After feature extraction, the output of the convolutional layers is passed through activation and pooling layers before being flattened and fed into fully connected layers for classification and other tasks. This feature extraction process helps improving the accuracy for tasks such as image classification.

3. Activation functions: They introduce non-linearity into the output of the convolutional layers, allowing the network to model complex, non-linear relationships in the input data. This is important for tasks such as image classification. Different types of activation functions, such as ReLU, sigmoid, softmax, and tanh, can be used in a CNN depending on the requirements of the task and the network architecture.

4. Batch Normalisation Layer: It improves the stability and speed of training by normalizing input values of a layer, which adjusts and scales activations to have zero mean and unit variance. This mitigates vanishing and exploding gradients that can occur during training, which can impede the network's ability to learn and prolong convergence. By preventing these issues, batch normalization can improve overall performance. It can be applied before and after the activation function, but generally performs better when placed after the activation function [79].

5. Pooling layers: They down-sample the output of the convolutional layer, reducing the spatial dimensions and parameters in the network while preserving key information. This operation is typically achieved through either max pooling or average pooling, which respectively extract the maximum or average value within a region of the feature map. This layer enhances the computational efficiency of the network, reduces overfitting, and expands the receptive field of later layers.

6. Fully connected layers: They are used to perform the final classification of an image by connecting all neurons in the previous layer to every neuron in the next layer, allowing the network to learn complex, non-linear relationships between the extracted features and the target classes. The flatten operation is typically used to convert the output of the last convolutional and pooling layers into a one-dimensional array, which can be fed into a fully connected layer. This helps reduce the total number of parameters and computations needed in the model, allowing for faster training and inference times.

7. Output Layer: It produces the final network output, which can be a probability distribution across different classes for classification tasks. The number of neurons in the output layer depends on the specific task, with each neuron representing a class. For multiclass classification, the activation function used in the output layer is softmax, while for binary classification, it is sigmoid. These activation functions ensure that the output values are normalized and add up to one because of which these outputs can be perceived as class probabilities.

A2. Transfer Learning [80]:

Transfer learning is a technique used in Deep Learning when data or computational resources are limited. It involves fine-tuning a model pre-trained on a large dataset (usually "ImageNet") for a smaller dataset. This method is mainly used in computer vision and Natural Language Processing tasks to improve overall performance, where large labelled datasets are unavailable. Pre-trained Deep Convolutional Neural networks, including DenseNet169, InceptionV3, ResNet50, Xception, EfficientNetB3, and VGG19, were used in this project to extract features from MRI scans for image classification.

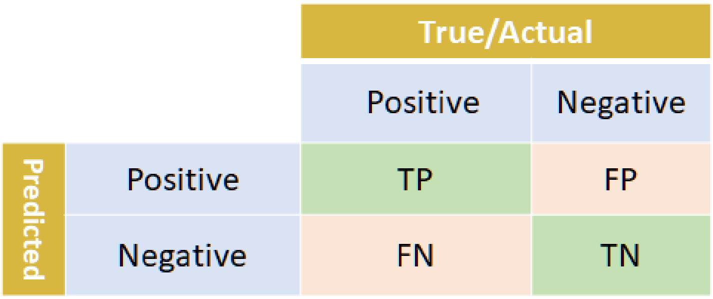
A3. Ensemble Learning [81]:

Ensemble learning uses different methods such as bagging, boosting, and stacking to combine models with the idea of leveraging their collective intelligence and suppressing undesirable characteristics like bias, variance and noise thus making the resulting model much more robust. Bagging trains multiple models independently on different subsets of the training data and then combines their predictions by averaging or voting. Boosting, on the other hand, trains multiple models sequentially and adjusts the weights of the training data to focus on the samples that are harder to classify. Finally, stacking involves training a meta-model to combine the predictions of multiple base models. Ensemble learning has been successful in various machine learning applications, including image classification, natural language processing, and recommendation systems. It can be applied to different types of models, such as decision trees, neural networks, and support vector machines

A4. Evaluation Metrics for Image Classification [82]:

Machine learning evaluation metrics provide quantitative measures of a model's performance. In CNNs, they are used to assess the effectiveness of the model for tasks such as image classification. Common metrics include accuracy, precision, recall, F1 score, Balanced Accuracy, Matthew's correlation coefficient (MCC) and confusion matrix. Higher values of these metrics indicate better classification performance. Here's a brief summary of each:

1. Confusion Matrix: It is a matrix that compares the predicted and actual class labels of a dataset to evaluate the performance of a classification model. The matrix has four cells representing the counts of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN). TP indicates when the model correctly predicts the positive class, while FP indicates when the model incorrectly predicts the positive class. FN represents the cases where the model incorrectly predicts the negative class, and TN represents the cases where the model correctly predicts the negative class. Moreover, it can be used to calculate other evaluation metrics such as accuracy, precision, recall, and F1 score.



		True/Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Fig A4(1) Confusion Matrix [82]

2. Accuracy: It is the ratio of correctly classified observations to the total number of predicted observations. It is calculated by dividing the sum of true negatives and true positives by the sum of all values in the confusion matrix. Accuracy is expressed as a percentage.

$$Accuracy = \frac{Number\ of\ correctly\ classified\ images}{Total\ Number\ of\ Images} \times 100$$

3. Precision: It indicates the confidence we have in our predictions. It is the ratio of True Positive Predicted observations and all positively predicted observations. It ranges from 0 to 1

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

4. Recall: It tells us about what proportion of actual positive observations the classifier could predict correctly. It ranges from 0 to 1.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

5. F1 Score: It is the harmonic mean of precision and recall and is used when both precision and recall are important for the given problem. It is a measure of the model's accuracy that considers both false positives and false negatives. A perfect F1 score is 1, indicating that the model has achieved both high precision and high recall. A score of 0 indicates poor performance.

$$\boldsymbol{F1\ Score} = \frac{\boldsymbol{2\ \times Precision\ \times Recall}}{\boldsymbol{Precision+Recall}}$$

6. Matthews Correlation Coefficient: It is a statistical measure that assesses the performance of a classification model by taking into account all four categories in the confusion matrix. It is called a "correlation coefficient" because it measures the correlation between the predicted and true labels and provides a balanced measure that is not affected by class imbalance. It ranges from -1 to +1, where +1 represents a perfect prediction, 0 represents a random prediction, and -1 represents a total disagreement between the prediction and the true labels.

$$\boldsymbol{MCC} = \frac{(\boldsymbol{TP} \times \boldsymbol{TN}) - (\boldsymbol{FP} \times \boldsymbol{FN})}{\sqrt{(\boldsymbol{TP+FP})(\boldsymbol{TP+FN})(\boldsymbol{TN+FP})(\boldsymbol{TN+FN})}}$$

7. Balanced Accuracy: It is the arithmetic mean of sensitivity (True Positive Rate or recall) and specificity (True Negative Rate).

$$\boldsymbol{Balanced\ Accuracy} = \frac{\boldsymbol{TPR} + \boldsymbol{TNR}}{\boldsymbol{2}}$$

8. AUC-ROC Curve: It is a graphical representation of the performance of a classification model. It plots the true positive rate (TPR) against the false positive rate (FPR) at different probability thresholds. The AUC-ROC curve measures the model's overall performance, and an AUC-ROC value of 0.5 indicates a random model, while a value of 1.0 indicates a perfect model. It is a widely used evaluation metric because it summarizes the model's performance across all possible probability thresholds and is not affected by the class distribution or the choice of the positive class threshold.
9. Precision Recall Curve: It is a graphical representation of the performance of a classification model that plots the precision against the recall at different probability thresholds. It measures the model's overall performance and is a useful evaluation metric, especially in cases of class imbalance. A perfect model has a PR curve that passes through the point (1,1), while a random model has a PR curve that is a straight line from the origin to the point (1,r), where r is the ratio of positive samples to the total number of samples.

Balanced Accuracy [73], MCC [74], Area Under Precision Recall Curve (AUPRC) and confusion matrix are the most appropriate metrics for this project because of the following reasons:

- F1 score may be misleading for heavily imbalanced datasets as it's biased towards the majority class due to equal weighting of precision and recall, regardless of class balance. Thus, a high F1 score does not necessarily indicate good performance on the minority class.
- F1 score is insensitive to misclassified true negatives and prioritizes positive instances more as both precision and recall are calculated based on the positive instances. When working on a heavily imbalanced dataset where positives are as important as the negatives, Balanced Accuracy does much better than F1 score because balanced accuracy shows a fast decrease when there is a decrease in the correct predictions of true negatives.
- Matthew's correlation coefficient takes into account all four categories of confusion matrix and scores high only if high percentage of both negative and positive instances are classified correctly, regardless of class balance or imbalance. It's unbiased towards any class and hence suitable for heavily imbalanced datasets.
- ROC curves plot the true positive rate (TPR) against the false positive rate (FPR) and only consider the positive class, ignoring the effects of the negative class. While the area under the curve (AUC) evaluates overall classification performance, it does not place greater emphasis on one class over the other, making it less effective for representing minority classes. On the other hand, precision-recall curves directly reflect class imbalance, and therefore provide a better means for highlighting differences between models for highly imbalanced datasets. The area under the precision-recall curve (AUPRC) combines the benefits of both the AUC-ROC and precision-recall curves by summarizing the model's performance across all possible probability thresholds, and is sensitive to the model's performance for the positive class (i.e., One vs Rest). Unlike the AUC, the AUPRC is not influenced by the class distribution or the choice of positive class threshold, making it an excellent metric for evaluating severely imbalanced datasets.

- Confusion matrix will enable the researcher to compare the performance on the minority classes for testing the hypothesis of this research.

B1. Generative Adversarial Networks (GANs) [56]

Generative Adversarial Networks (GANs) are a kind of neural network that can generate new data such as images, text, and audio, without being explicitly taught. GANs are comprised of two networks, a generator and a discriminator. The generator produces novel data samples, while the discriminator attempts to differentiate between the fake and real data. These networks are trained in an adversarial manner, where the generator aims to deceive the discriminator, while the discriminator tries to accurately classify the data.

During the training process of GANs, the generator network utilizes a random noise vector to generate new samples that are convincing enough to deceive the discriminator. The discriminator network takes both real and generated samples as input and produces a probability score that indicates whether the sample is real or generated. As GANs are composed of two networks with distinct objectives, they cannot be trained in the same way as traditional neural networks. Therefore, each training iteration of GANs is divided into two phases:

1. The initial phase of GAN training entails training the discriminator. This involves selecting a batch of real images from the training dataset and pairing them with an equivalent number of fake images produced by the generator. These images are labelled as 1 for real and 0 for fake, and the discriminator is trained for one step on this labelled batch using the binary cross-entropy loss. $\boldsymbol{\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{i=m}[logD\left(x^{(i)}\right) + \log\left(1 - D(G(z^{(i)}))\right)]}$. During this stage, it is critical to note that backpropagation only optimizes the discriminator's weights.

2. The second phase involves training the generator by generating another batch of fake samples and having the discriminator determine their authenticity. Unlike the first phase, no real images are included in this batch, and all samples are marked as real, causing the discriminator to believe the fakes are real. Notably, during this step, the discriminator's weights are frozen, and only the generator's weights are updated via backpropagation.

Although the generator is never shown real images, it gradually learns to generate credible fake samples. However, GAN training is a challenging task due to a phenomenon called "Mode Collapse," where the generator only produces samples similar to a specific subset of the authentic data, rather than a diverse array of samples. This occurs because the generator focuses too much on those images which it can so generate well that it can fool the discriminator each time and eventually forgets how to produce other images [59]. Mode collapse becomes more likely with more epochs [59] which is why many researchers could only synthesize synthetic MRIs using just 400-700 images and that too for 1000 epochs. Furthermore, since the generator and discriminator networks are in constantly pushing against each other, their parameters may fluctuate and become unstable. Thus, significant fine-tuning of hyperparameters may be necessary.

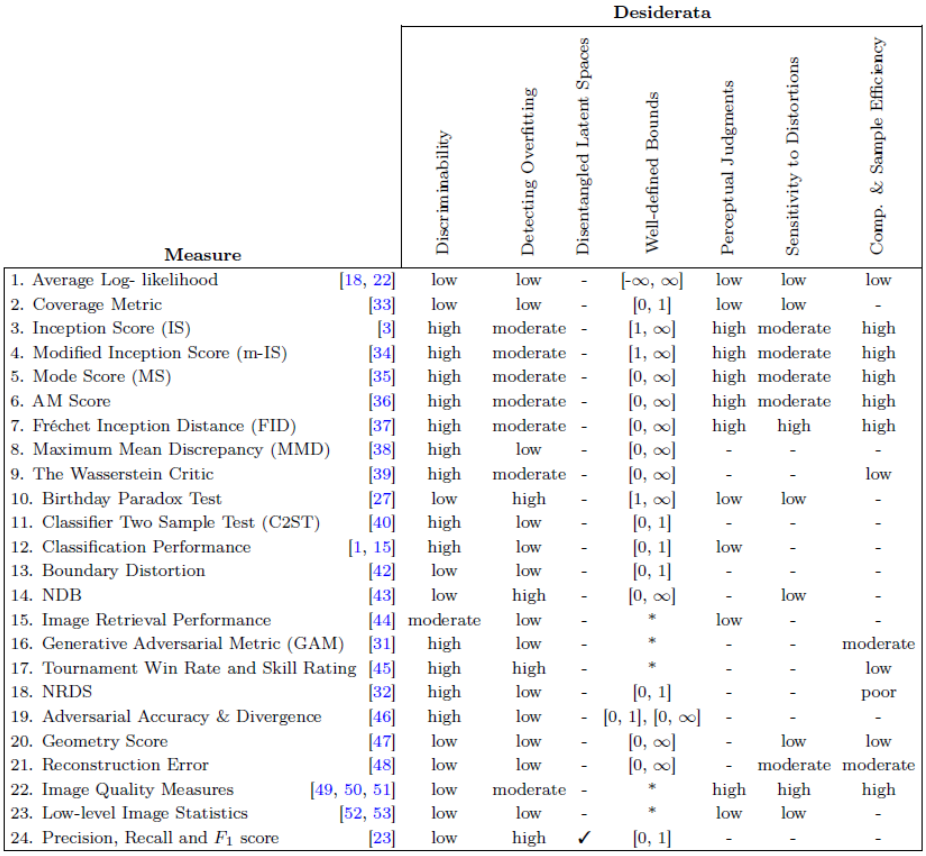
B2. Deep Convolutional Generative Adversarial Networks (DC-GANs) [58]

DC-GANs (Deep Convolutional GANs) employ convolutional layers in both the generator and discriminator networks to generate synthetic image data, and employ the same loss function as GANs. To ensure the stability of DC-GANs, the following are the key guidelines to be followed while building them:

- Replace any pooling layers with strided convolutions (in the discriminator) and transposed convolutions (in the generator).
- Use batch normalisation in both the generator and the discriminator, except in the generator's output layer and the discriminator's input layer.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in the generator for all layers except the output layer, which should use tanh
- Use leaky ReLU activation in the discriminator for all layers.

B3. Evaluation Metrics for Synthetic Image Generation [61]:

The issue of establishing objective metrics that accurately correlate with perceived quality measurement is still an open issue. In order to assess the quality of synthetic MRIs produced by GAN, suitable quantitative and qualitative evaluation techniques were employed. Various GAN evaluation methods were assessed and contrasted by the authors of [61]. Based on their findings, the FID Score and Image Quality Measures such as SSIM, Sharpness Difference, and PSNR were identified as the most effective methods. These are discussed below:



Measure		Desiderata						
		Discriminability	Detecting Overfitting	Disentangled Latent Spaces	Well-defined Bounds	Perceptual Judgments	Sensitivity to Distortions	Comp. & Sample Efficiency
1. Average Log- likelihood	[18, 22]	low	low	-	[-∞, ∞]	low	low	low
2. Coverage Metric	[33]	low	low	-	[0, 1]	low	low	-
3. Inception Score (IS)	[3]	high	moderate	-	[1, ∞]	high	moderate	high
4. Modified Inception Score (m-IS)	[34]	high	moderate	-	[1, ∞]	high	moderate	high
5. Mode Score (MS)	[35]	high	moderate	-	[0, ∞]	high	moderate	high
6. AM Score	[36]	high	moderate	-	[0, ∞]	high	moderate	high
7. Fréchet Inception Distance (FID)	[37]	high	moderate	-	[0, ∞]	high	high	high
8. Maximum Mean Discrepancy (MMD)	[38]	high	low	-	[0, ∞]	-	-	-
9. The Wasserstein Critic	[39]	high	moderate	-	[0, ∞]	-	-	low
10. Birthday Paradox Test	[27]	low	high	-	[1, ∞]	low	low	-
11. Classifier Two Sample Test (C2ST)	[40]	high	low	-	[0, 1]	-	-	-
12. Classification Performance	[1, 15]	high	low	-	[0, 1]	low	-	-
13. Boundary Distortion	[42]	low	low	-	[0, 1]	-	-	-
14. NDB	[43]	low	high	-	[0, ∞]	-	low	-
15. Image Retrieval Performance	[44]	moderate	low	-	*	low	-	-
16. Generative Adversarial Metric (GAM)	[31]	high	low	-	*	-	-	moderate
17. Tournament Win Rate and Skill Rating	[45]	high	high	-	*	-	-	low
18. NRDS	[32]	high	low	-	[0, 1]	-	-	poor
19. Adversarial Accuracy & Divergence	[46]	high	low	-	[0, 1], [0, ∞]	-	-	-
20. Geometry Score	[47]	low	low	-	[0, ∞]	-	low	low
21. Reconstruction Error	[48]	low	low	-	[0, ∞]	-	moderate	moderate
22. Image Quality Measures	[49, 50, 51]	low	moderate	-	*	high	high	high
23. Low-level Image Statistics	[52, 53]	low	low	-	*	low	low	-
24. Precision, Recall and F_1 score	[23]	low	high	✓	[0, 1]	-	-	-

Fig B3(1) Comparison of multiple GAN evaluation metrics [61]

1. Fréchet Inception Distance (FID) [76]: It extracts the feature embeddings from a batch of both real and synthetic images using a pretrained InceptionV3 model and computes the following:

$$FID = \left\|\mu_r - \mu_g\right\|^2 + T_r\left(\Sigma_r + \Sigma_g - 2\left(\Sigma_r\Sigma_g\right)^{1/2}\right)$$

   It is considered to be the most appropriate metric for evaluating performance of GANs as it not only compares the quality of synthetic images by computing the Euclidean distance (first term) between the mean of real and synthetic images but also compares the distribution of synthetic images with respect to the distribution of real images by computing the second term. This makes it sensitive to “mode collapse” (discussed in the next section) which is why it will give a good score (ideally 0) only if both quality and diversity of synthetic images are as good as the real ones. Additionally, unlike SSIM, PSNR and SD which compare an image to a reference image, FID can compare a whole batch of images with a reference batch of images which makes it ideal for comparing epochs during training of GAN. Being a statistical measure, it is sensitive to both number of images in the batch and the scale of those images. This is why there is no universally acceptable threshold for FID [76] but FID < 5 was found appropriate for this project through experimentation.

2. Structural Similarity Index (SSIM): SSIM is designed to measure the perceptual difference between pixels of two images by comparing their luminance, contrast, and structure. It produces a score between 0 and 1, where a score of 1 indicates that the two images are identical, and a score of 0 indicates that they are completely dissimilar. SSIM > 0.9 is considered to be an excellent reconstructed/synthetic image [63].

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

   Where x is the estimated MRI and y is the ground truth MRI, μ_x *and* μ_y are their averages, σ_x^2 *and* σ_y^2 are their variances and σ_{xy} is the covariance of x and y. C_1*and* C_2 are just constants used to stabilize the division with weak denominator.

3. Peak Signal to Noise Ratio (PSNR): It is used to measure the ratio between the maximum possible intensity value and the mean squared error of the synthetic and the real image. Ideally, PSNR should be between 30 dB and 50 dB [63] for 8 bit ".jpg" images. Its formula is given below where n is the number of pixels in the given image.

$$\boldsymbol{PSNR = 10\log_{10}\frac{(\max(y))^2}{\frac{1}{n}\sum_i^n(y_i-\hat{y})^2}}$$

4. Sharpness Difference (SD): It measures the loss of sharpness during image generation. Ideally the sharpness difference between a real and synthetic image should be 0 [63]. It is computed as:

$$\boldsymbol{SD(I,K) = 10\log_{10}\left(\frac{MAX_I^2}{\nabla S_{I,K}}\right)}$$

5. Seaborn's Distplot: It is a type of histogram that displays the distribution of a continuous variable in a dataset. It offers a fast overview of a variable's distribution, including its range, shape, and central tendency. Distplots are commonly used to compare and contrast the distributions of different variables or datasets. They can also be used to determine how similar is the distribution of real and synthetic images if they overlap.

References:

1. S. Liu, S. Liu, W. Cai, H. Che, S. Pujol, R. Kikinis, D. Feng, M. J. Fulham, and Adni, “Multimodal neuroimaging feature learning for multiclass diagnosis of Alzheimer's disease”, "IEEE Trans. Biomed. Eng., vol. 62, no. 4, pp. 1132-1140, Apr. 2015, doi: 10.1109/TBME.2014.2372011.
2. SP Singh, L Wang, S Gupta, H Goli, P Padmanabhan, B. Gulyás. 3D Deep learning on medical images: a review, 2020; pp. 1–13.
3. S. Przedborski, M. Vila, and V. Jackson-Lewis, “Series introduction: Neurodegeneration: What is it and where are we?” J. Clin. Invest., vol. 111, no. 1, pp. 3-10, Jan. 2003, doi: 10.1172/JCI200317522.
4. C.M. Clark, C. Davatzikos, A. Borthakur, A. Newberg, S. Leight, V.M.Y. Lee, J.Q. Trojanowski. Biomarkers for Early Detection of Alzheimer Pathology. Neuro-Signals, pp. 1-8, December 5, 2007
5. S. Afzal, M. Maqsood, F. Nazir, U. Khan, F. Aadil, K.M. Awan, I. Mehmood and O.Y. Song. “A Data Augmentation-Based Framework to Handle Class Imbalance Problem for Alzheimer's Stage Detection”. IEEE Access. Biomed Eng., vol 2, pp. 115528-115539, June 24 2019, DOI: 10.1109/ACCESS.2019.293278
6. S. Liu, W. Cai, H. Che, S. Pujol, D. Feng, M.J. Fulham and S. Liu. Multimodal Neuroimaging Feature Learning for Multiclass Diagnosis of Alzheimer’s Disease. IEEE Transactions Biomed Eng. Vol. 62, No. 4, April 2015. pp. 1-9.
7. S. Klöppel et al., "Accuracy of dementia diagnosis — a direct comparison between radiologists and a computerized method," Brain, vol. 131, no. 11, pp. 2969-2974, 2008.
8. A. Khan, M. Usman. Early Diagnosis of Alzheimer’s Disease using Machine Learning Techniques. IEEE Trans. Biomed. Eng. pp. 1-8.
9. N. Yamanakkanavar, J. Y. Choi, and B. Lee, "MRI segmentation and classification of the human brain using deep learning for diagnosis of Alzheimer’s disease: a survey," Sensors (Switzerland), vol. 20, no. 11, pp. 1-31, 2020.
10. Weiner, M.W.; Veitch, D.P.; Aisen, P.S.; Beckett, L.A.; Cairns, N.J.; Green, R.C.; Harvey, D.; Jack, C.R.; Jagust, W.; Liu, E. The Alzheimer’s Disease Neuroimaging Initiative: A review of papers published since its inception. Alzheimer’s Dement. 2013, 9, e111–e194.. https://adni.loni.usc.edu/data-samples/access-data/
11. Marcus, D. Wang, T. OASIS: Cross-sectional, MRI data in young, middle aged, nondemented, and demented, older adults. J. Cogn. Neurosci. 2017, 19, 1498–1507.
12. Molinuevo, J.L., Ritchie, C., Truyen, L., Satlin, A., Van der Geyten, S., Lovestone, S., Bayer, A. and Simmons, A., 2016. European Prevention of Alzheimer's Dementia (EPAD) initiative: A multinational dementia prevention research project. Alzheimer's & Dementia, 12(7), pp. P1229-P1230. Available at: https://ep-ad.org/open-access-data/data/. DOI: 10.1016/j.jalz.2016.06.2364
13. Sarvesh Dubey, 2017. Kaggle Alzheimer's Dataset. https://www.kaggle.com/tourist55/alzheimers-dataset-4-class-of-images
14. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 618-626, doi: 10.1109/ICCV.2017.74.
15. S. Murugan, C. Venkatesan, M.G. Sumithra, X.Z. Gao, B. Elakkiya, M. Akila and S. Manoharan. “DEMNET: A Deep Learning Model for Early Diagnosis of Alzheimer Diseases and Dementia From MR Images”, IEEE Access. pp. 90319-90329, June 18 2021, doi: 10.1109/ACCESS.2021.3090474.
16. T. Jo, K. Nho, AJ. Saykin. Deep learning in Alzheimer’s disease: diagnostic classification and prognostic prediction using neuroimaging data, Frontiers in Aging Neuroscience. 2019;11.
17. D. Schenk et al., "Immunization with amyloid-beta attenuates Alzheimer-disease like pathology in the PDAPP mouse," Nature, vol. 400, pp. 173-177, 1999.
18. J. Giorgio, S. M. Landau, W. J. Jagust, P. Tino, and Z. Kourtzi, “Modelling prognostic trajectories of cognitive decline due to Alzheimer's disease,” NeuroImage, Clin., vol. 26, Jan. 2020, Art. no. 102199, doi: 10.1016/j.nicl.2020.102199.
19. D. A. Butterfield and C. M. Lauderback, "Lipid peroxidation and protein oxidation in Alzheimer’s disease brain: potential causes and consequences involving amyloid beta-peptide-associated free radical oxidative stress," Free Rad. Biol. Med., vol. 32, pp. 1050-1060, 2002.
20. H.A. Helaly, M. Badaway, A.Y. Haikal. Deep Learning Approach for Early Detection of Alzheimer’s Disease. Cognitive Computation, 2022, pp. 1711-1727, doi: 10.1007/s12559-021-09946-2
21. C. Adelina, "The costs of dementia: advocacy, media, and stigma," Alzheimer's Dis. Int. World Alzheimer Rep., vol. 2, 2019, pp. 100-101, 2019.

22. A. Segato, A. Marzullo, F. Calimeri, and E. De Momi, "Artificial intelligence for brain diseases: a systematic review," APL Bioeng., vol. 4, no. 1, p. 041501, 2020.
23. N. Yamanakkanavar, J. Y. Choi, and B. Lee, "MRI segmentation and classification of the human brain using deep learning for diagnosis of Alzheimer's disease: a survey," Sensors (Switzerland), vol. 20, no. 11, pp. 1-31, 2020.
24. M. B. T. Noor, N. Z. Zenia, M. S. Kaiser, S. Al Mamun, and M. Mahmud, "Application of deep learning in detecting neurological disorders from magnetic resonance images: a survey on the detection of Alzheimer's disease, Parkinson's disease, and schizophrenia," Brain Informatics, vol. 7, no. 1, p. 11, 2020.
25. F. Li, L. Tran, K.-H. Thung, S. Ji, D. Shen, and J. Li, "A robust deep model for improved classification of AD/MCI patients," IEEE J. Biomed. Health informatics, vol. 19, no. 5, pp. 1610-1616, 2015.
26. A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," Comput. Intell. Neurosci., vol. 2018, pp. 1-13, 2018.
27. R.C. Barber, Biomarkers for Early Detection of Alzheimer's Disease. Neuro-Signals, vol. 110, September 2010. pp. 1-6
28. D. Zhang, "Multimodal classification of Alzheimer's disease and mild cognitive impairment," Neuroimage, vol. 55, no. 3, pp. 856-867, 2011.
29. A. Khan, M. Usman. Early Diagnosis of Alzheimer's Disease using Machine Learning Techniques. IEEE Trans. Biomed. Eng. pp. 1-8.
30. D. Lu, A. Disease Neuroimaging Initiative, K. Popuri, G. W. Ding, R. Balachandar, and M. F. Beg, "Multimodal and multiscale deep neural networks for the early diagnosis of Alzheimer's disease using structural MR and FDG-PET images" Sci. Rep., vol. 8, no. 1, pp. 1-13, Dec. 2018, doi: 10.1038/s41598-018-22871-z.
31. Y. Gupta, K. H. Lee, K. Y. Choi, J. J. Lee, B. C. Kim, and G. R. Kwon, ``Early diagnosis of Alzheimer's disease using combined features from voxel-based morphometry and cortical, subcortical, and hippocampus regions of MRI T1 brain images," PLoS ONE, vol. 14, no. 10, 2019, Art. no. e0222446, doi: 10.1371/journal.pone.0222446.
32. S. Ahmed, K. Y. Choi, J. J. Lee, B. C. Kim, G.-R. Kwon, K. H. Lee, and H. Y. Jung, ``Ensembles of patch-based classifiers for diagnosis of Alzheimer's diseases," IEEE Access, vol. 7, pp. 73373-73383, 2019, doi: 10.1109/ACCESS.2019.2920011.
33. A. Mehmood, S. Yang, Z. Feng, M. Wang, A. S. Ahmad, R. Khan, M. Maqsood, and M. Yaqub, "A transfer learning approach for early diagnosis of Alzheimer's disease on MRI images," Neuroscience, vol. 460, pp. 43-52, Apr. 2021, doi: 10.1016/j.neuroscience.2021.01.002.
34. H. Wang, Y. Shen, S. Wang, T. Xiao, L. Deng, X. Wang, and X. Zhao, "Ensemble of 3D densely connected convolutional network for diagnosis of mild cognitive impairment and Alzheimer's disease," Neurocomputing, vol. 333, pp. 145-156, Mar. 2019, doi: 10.1016/j.neucom.2018.12.018.
35. G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely connected convolutional networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700-4708.
36. R. R. Janghel and Y. K. Rathore, "Deep convolution neural network-based system for early diagnosis of Alzheimer's disease," IRBM, vol. 1, pp. 1-10, Jul. 2020, doi: 10.1016/j.irbm.2020.06.006.
37. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
38. C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.
39. Fu, Y.G. and Shen, R.M., 2008. Color image watermarking scheme based on linear discriminant analysis. Computer Standards & Interfaces, 30(3), pp.115-120.
40. J. Wu, "Advances in K-means clustering: a data mining thinking," Springer Science & Business Media, 2012.
41. X. Wu et al., "Top 10 algorithms in data mining," Knowledge and Information Systems, vol. 14, no. 1, pp. 1-37, 2008.
42. C. Ge, Q. Qu, I. Y.-H. Gu, and A. Store Jakola, ``Multiscale deep convolutional networks for characterization and detection of Alzheimer's disease using MR images," in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2019, pp. 789-793, doi: 10.1109/ICIP.2019.8803731.
43. D, Zhang, 2011. Multimodal classification of Alzheimer's disease and mild cognitive impairment. Neuroimage, 55(3), pp. 856-867.
44. E. Y. Puspaningrum, R. R. Wahid, R. P. Amaliyah and C. Nisa', "Alzheimer's Disease Stage Classification using Deep Convolutional Neural Networks on Oversampled Imbalance Data," 2020 6th Information Technology International Seminar (ITIS), Surabaya, Indonesia, 2020, pp. 57-62, doi: 10.1109/ITIS50118.2020.9321061.

45. 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
46. N. Ma, X. Zhang, H.T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 116-131.
47. G. Ahmed, M. Joo Er, M. Muhammad Sadiq Fareed, S. Zikria, S. Mahmood, J. He, M. Asad, S. Fizzad Jilani, M. Aslam. "DAD-Net: Classification of Alzheimer's Disease Using ADASYN Oversampling Technique and Optimized Neural Network". 20 Oct 2022, vol 27, pp. 1-6.
48. H. He, Y. Bai, E.A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), 2008, pp. 1322-1328.
49. S. Murugan, C. Venkatesan, M.G. Sumithra, B. Elakkiya, M. Akila, S. Manoharan "DEMNET: A Deep Learning Model for Diagnosis of Alzheimer Diseases and Dementia From MR Images" IEEE Access, vol. 1, pp. 90319-90329, 2021
50. N.V. Chawla et al., "SMOTE: synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002.
51. N.S. Altman, "An introduction to kernel and nearest-neighbour nonparametric regression," The American Statistician, vol. 46, no. 3, pp. 175-185, 1992.
52. M.W. Oktavian, N. Yudistira, A. Ridok. "Classification of Alzheimer's Disease Using the Convolutional Neural Network with Transfer Learning and Weighted Loss.
53. W. Liu et al., "Ssd: Single shot multibox detector," in European conference on computer vision, pp. 21–37, 2016
54. H. Helaly, M. Badawy, A. Haikal, "Deep Learning Approach for Early Detection of Alzheimer's Disease", pp. 1711-1727, 2021
55. S. Afzal et al., "A Data Augmentation-Based Framework to Handle Class Imbalance Problem for Alzheimer's Stage Detection," in IEEE Access, vol. 7, pp. 115528-115539, 2019, doi: 10.1109/ACCESS.2019.2932786.
56. I. Goodfellow et al., "Generative adversarial nets," in Advances in Neural Information Processing Systems, 2014, pp. 2672-2680.
57. C. Qu, Y. Zou, Q. Dai, Y. Ma, J. He, Q. Liu, W. Kuang, Z. jia, T. Chen and Q. Gong. "Review paper: Advancing diagnostic performance and clinical applicability of deep learning-driven generative adversarial networks for Alzheimer's disease". Psychoradiology, vol. 4, pp. 225-248.
58. A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv preprint arXiv:1511.06434, 2015.
59. Mukherjee, T., Sharma, S. and Suganthi, K. "Alzheimer detection using deep convolutional GAN". In 2021 IEEE Madras Section Conference (MASCON) (pp. 1-6). IEEE.
60. S. Hu, Wu Yu, Z. Chen, and S. Wang. "Medical image reconstruction using generative adversarial network for Alzheimer disease assessment with class-imbalance problem". In 2020 IEEE 6th international conference on computer and communications (ICCC) (pp. 1323-1327). IEEE.
61. A. Borji, "Pros and cons of GAN evaluation measures," Computer Vision and Image Understanding, vol. 179, pp. 41-65, 2019.
62. J. Islam and Y. Zhang. GAN-based synthetic brain PET image generation. Brain informatics, 2020, vol. 7, pp.1-12.
63. I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. Courville. Improved Training of Wasserstein GANs. Machine Learning, vol 3, 2017. pp. 1-20
64. F. Chollet & others, 2015. Keras. Available at: https://github.com/fchollet/keras.
65. M. Abadi et al., 2016. Tensorflow: A system for large-scale machine learning. In: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, 2016, pp. 265-283.
66. F. Mohr and M. Wever, 2022. Naive automated machine learning. Machine Learning, pp. 1-40.
67. J. Tan, J. Yang, S. Wu, G. Chen, and J. Zhao, 2021. A critical look at the current train/test split in machine learning. arXiv preprint arXiv:2106.04525.
68. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, 2016. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, June 2016, pp. 2818-2826.
69. F. Chollet, 2017. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, July 2017, pp. 1251-1258.

70. Tan, M. and Le QV, E., 1905. rethinking model scaling for convolutional neural networks. arXiv. 2019 doi: 10.48550. arXiv.
71. D.P. Kingma and J. Ba, 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, December 2014, pp. 1-15.
72. L. Prechelt, 2012. Early stopping—but when?. In: Neural networks: tricks of the trade: second edition, Springer, Berlin, Germany, April 2012, pp. 53-67.
73. K.H. Brodersen, C.S. Ong, K.E. Stephan, and J.M. Buhmann, August 2010. The balanced accuracy and its posterior distribution. In: 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23-26 August 2010, pp. 3121-3124.
74. D. Chicco and G. Jurman, 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics, 21, pp.1-13.
75. N. Bjorck, C.P. Gomes, B. Selman, and K.Q. Weinberger, 2018. Understanding batch normalization. In: Advances in neural information processing systems, vol. 31, Montreal, Canada, December 2018, pp. 758-768.
76. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S., 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems, 30.
77. M. Waskom et al., 2017. mwaskom/seaborn: v0.8.1 (September 2017), Zenodo. Available at: https://doi.org/10.5281/zenodo.883859. (accessed Mar. 17, 2023).
78. J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2223-2232.
79. K. O'Shea and R. Nash, "An introduction to convolutional neural networks," arXiv preprint arXiv:1511.08458, 2015.
80. C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang and C. Liu, "A survey on deep transfer learning," in Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III, pp. 270-279, Springer International Publishing, 2018.
81. O. Sagi and L. Rokach, "Ensemble learning: A survey," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 8, no. 4, pp. e1249, 2018.
82. M. Fatourechi, R. K. Ward, S. G. Mason, J. Huggins, A. Schlögl, and G. E. Birch, "Comparison of evaluation metrics in classification applications with imbalanced datasets," in 2008 Seventh International Conference on Machine Learning and Applications, San Diego, CA, USA, Dec. 2008, pp. 777-782.
83. A. Roy, M. Rahman, S. Ahmed, “Federated GAN Based Biomedical Image Augmentation and Classification for Alzheimer’s Disease”. Department of Computer Science and Engineering, BRAC University, Sept 2022, pp. 1-23
84. V. Jain, O. Nankar, DJ. Jerrish, S. Gite, S. Patil, K. Kotecha, “A Novel AI-Based System for Detection and Severity Prediction of Dementia Using MRI”, November 2021, IEEE Access, Vol 2, pp. 154324-154346.
85. J. Datta, B. Durdana, S. Rafi, “Deep Convolutional GAN-based Data Augmentation for Medical Image Classification”. Department of Computer Science and Engineering, BRAC University, Jan 2022, pp. 1-44