MRI Data Augmentation Utilizing L$_\alpha$-GAN

MTHE 493 Group 06

Deven Shidfar
Jillian Menikefs
Kaleb Huneau
David Laughton
Sudeepta Talukdar
Owen Borthwick
Oscar Brown

April 11, 2024

Contents

1 Abstract

To successfully study a human medical condition, researchers and doctors rely heavily on data collected from clinical trials. This data provides insight into the disease or condition, which leads to formulation of potential cures. However, the process of acquiring medical data comes with numerous challenges.

One significant challenge lies in acquiring Magnetic Resonance Imaging (MRI) scan data, crucial for understanding conditions like brain tumours. This is complicated by issues such as data privacy concerns, the high cost associated with conducting scans, and the potential risks associated with injecting contrast agents for clear images, which can deter individuals from participating in such studies. In fact, according to the National Library of Medicine, only 0.014% people in USA develop brain tumours each year, further worsening the scarcity of available data. The scarcity of data is an obstacle to advancing public health initiatives in terms of preventative measures.

In this project, the focus will be on exploring and implementing Generative Adversarial Networks (GANs) in the context of generation and augmentation. GANs represent a distinctive neural network architecture within the realm of machine learning. Since their introduction in 2014 by Goodfellow et al., GANs have demonstrated versatility across various applications, notably in image generation [1]. Specifically, this study aims to investigate the utilization of GANs for synthesizing MRI scans, thereby enhancing the diversity and volume of available datasets and improve classification algorithms.

2 Introduction and Background

2.1 Neural Networks

The building blocks of a GAN is neural networks. A neural network is a form of machine learning where computers learn or perform tasks by analyzing training examples. A neural network is made of layers of nodes or neurons made to mimic the way the human brain is structured. Neural networks begin with an input layer followed by several layers of nodes, also known as hidden layers, followed by the output layer. The structure of a neural network can be seen below in Figure 1.

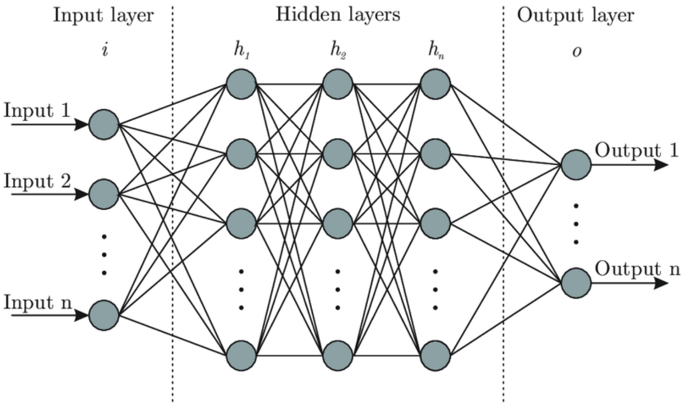


Figure 1: Architecture of a Neural Network [8]

Each node of a given layer is connected to every node in the previous layers, from which it receives data, and it is connected to every node after it, to which it sends data.

Each of the incoming connections are assigned weights. Additionally, there exists a bias term which is analogous to the role of a constant in a linear function such as $y = mx + c$; the weight is analogous to the slope. The weights and bias terms are used to create a linear combination of node values from the previous layer. Note that the bias is used for shifting the activation function left or right. This value passes through an activation function to add non-linearity to the model. Non linearity is important as it allows the network to learn complex patterns as seen in Figures 4a and 4b [9]. Common activation functions include the Sigmoid and ReLU activations. Details of these activation functions can be seen below in Figures 2 and 3.

Sigmoid function:

$$f(x) = \frac{1}{1+e^{-x}} \tag{1}$$

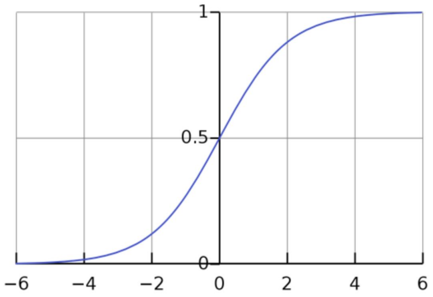


Figure 2: Sigmoid Activation Function

ReLU function:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

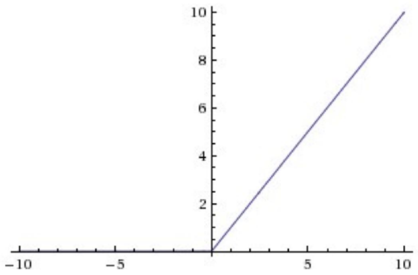
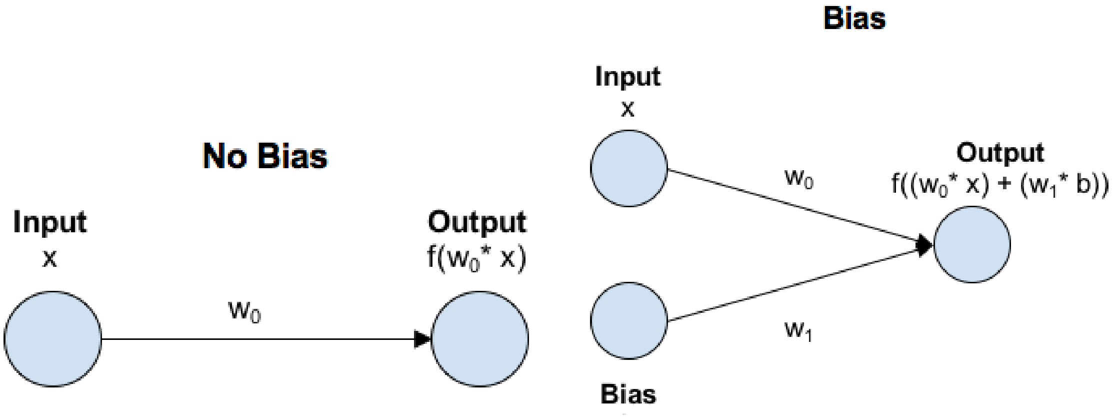


Figure 3: ReLU Activation Function



(a) Activation Function with No Bias

(b) Activation Function with Bias

Figure 4: Comparison of Activation Functions with and without Bias [10]

In order to learn, neural networks are given an objective function to minimize the gap between its predictions and the real data. To do this, the network is trained using labeled examples. It predicts labels for each example, then calculates how far off it was from the actual labels. This difference is used to adjust the network's internal settings, like weights and biases, through techniques such as back-propagation and gradient descent. Common tasks include classification through cross-entropy loss and regression tasks through quadratic mean squared error loss. With repeated exposure to examples and adjustments, the network aims to get better at its task, ideally achieving high accuracy and speed [11].

2.2 Convolutional Neural Networks

The Convolutional Neural Network (CNN) is a popular type of neural network layer that is used for analyzing images. It allows the network to analyze groups of pixels by sliding a window across the image and taking the convolution of the window with a filter known as the kernel. Each time the convolution of the window and kernel is taken, the window moves by a number of pixels, called the stride, until the window reaches the end of the image. A simple example from IBM is presented in Figure 5, showcasing the operations of a convolution layer [12].

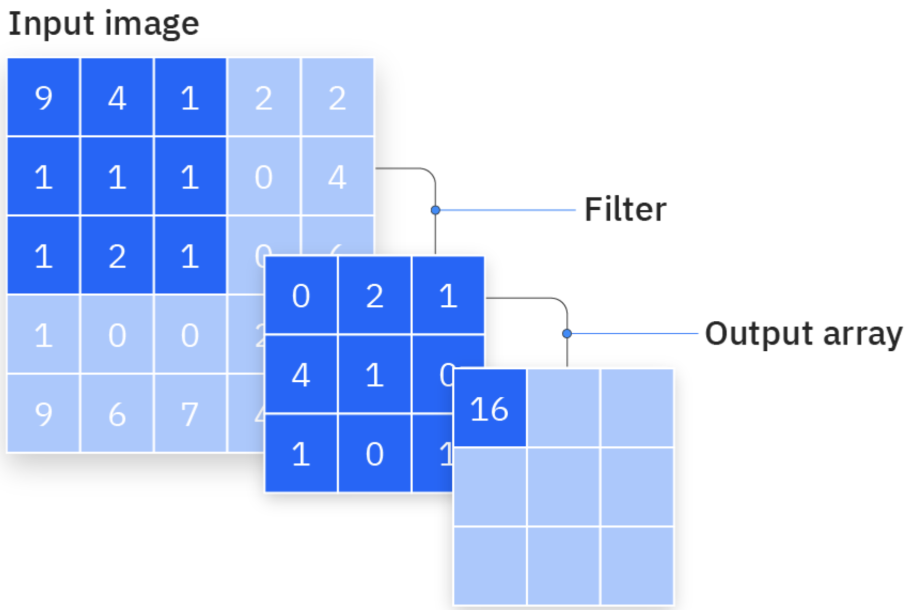


Figure 5: Convolution Layer where the Output[0][0] is given by:
(9*0) + (4*2) + (1*1) + (1*4) + (1*1) + (1*0) + (1*1) + (2*0) + (1*1)
= 0 + 8 + 1 + 4 + 1 + + 0 + 1 + 0 + 1
= 16

Within a GAN, the convolution layers play an important role in the definition of the discriminator. Using the CNN in the discriminator allows it to pick up on features and decide if the image is real or generated. The CNN appears again in the generator, but a different version called the transposed convolution layer is used. The transposed convolution has an output space which is larger than the input space. This makes them useful for any task where up-sampling is required and lends well to the problem of image generation.

2.3 Generative Adversarial Networks (GANs)

GAN represents a class of machine learning algorithms introduced by Ian Goodfellow et al. in 2014 [1]. GANs have gained prominence for their approach to generate synthetic data with close resemblance to real data.

GANs consist of two neural networks: a generator and a discriminator. These networks are made to engage in a competitive game where the generative network generates synthetic data from random noise, aiming to mimic the target dataset. The discriminator's primary function is to assess and differentiate the generative dataset from the real dataset. It assigns a score between 0 and 1 to the generated data, depending on how closely it resembles the actual data. In their interaction, the two players engage in a minimax game – a strategic interaction in which one neural network aims to minimize its maximum possible loss while the other neural network aims to maximize it. In other words, the minimax game focuses on reducing the measure of dissimilarity between the generative and real data distribution, which is quantified by the Jensen-Shannon divergence. As a result, the optimal generative network, under unlimited resource constraints, converges towards modeling the real data distribution accurately [1].

Since 2014, GANs have had numerous extensions that enhance their performance. These extensions include different approaches including Least Square GAN (LSGAN) [2] and Least k^{th} Order GAN (LkGAN) [3] .

Other models such as RenyiGAN [3] and L_α-GAN [4] introduce more versatile information-theoretic loss functions. Several of these GAN extensions generalize the equilibrium point theorem of the original GAN introduced by Goodfellow et al., resulting in potential enhanced stability and performance in various applications [3].

GANs are versatile and widely used in various fields. They help create realistic images, improve photos, and generate synthetic faces. In medicine, GANs enhance medical images and aid in disease detection. They also power text generation, translation, and speech synthesis, and have applications in anomaly detection and data de-noising [5].

2.3.1 Mathematics of GANs

As previously noted, GANs consist of two neural networks: a generator (G), and a discriminator (D) that compete in a minimax zero-sum game. The GAN architecture can be seen in Figure 6 [6].

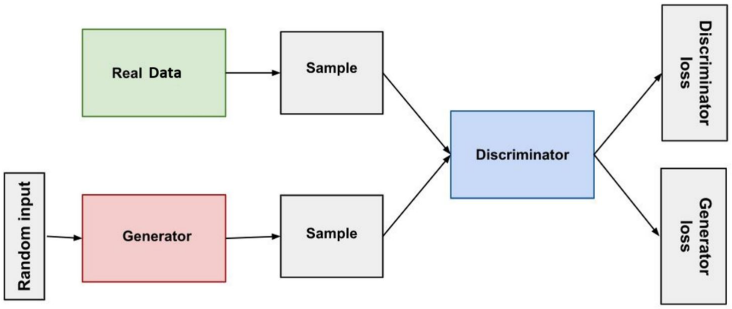


Figure 6: Architecture of a GAN system [6]

To quantitatively describe the minimax game, let us define the following terms:
X := random variable which models real data with probability density function (pdf) p_X.
G := random variable which models generated data with pdf p_G.
Z := random variable which models noise with pdf p_Z.
$\mathbb{X}$:= data sample space.

Quantitatively, it can be said that the generator takes p_Z as its input and gives p_G as its output, with an objective to have p_G mimic p_X as closely as possible. The discriminator then receives data from either p_X or p_G as input and produces an output within the interval $[0,1]$. The score that the discriminator assigns signifies the likelihood that the provided data belongs to the real distribution, where a score of 1 indicates that the discriminator is certain that the provided data is real. Therefore, the minimax game can be seen as a scenario where the discriminator seeks to minimize its classification error, while the generator strives to maximize this classification error.

Given the definitions above, the minimax game can be expressed through the loss function $V(G,D)$ of the original GAN, also known as VanillaGAN, as seen in (3). In this process, D is trained to maximize its ability to correctly label both training examples and samples from G. At the same time, G is trained to minimize the $\log(1-D(G(z)))$ term as seen below:

$$\min_G \max_D V(G,D) = \min_G \max_D \left(E_{A\sim p_X}[\log(D(A))] + E_{B\sim p_Z}[\log(1-D(G(B)))]\right). \tag{3}$$

The optimization problem in (3) reduces to minimizing the Jensen-Shannon divergence as follows:

$$\min_{G}\max_{D} V(G,D) = \min_{p_G} 2 \cdot JSD(p_X||p_G) - 2\log 2, \tag{4}$$

where $JSD(p_G||p_X)$ denotes the Jensen Shannon Divergence defined by

$$JSD(P||Q) = \frac{1}{2}D(p||M) + \frac{1}{2}D(q||M)\ , \tag{5}$$

where $M = \frac{1}{2}(p+q)$ is the mixture distribution of p and q, and $D(p||q)$ is the Kullback-Leibler Divergence between pdfs p and q with common support S:

$$D(p||q) = \int_S p(x)\log\left(\frac{p(x)}{q(x)}\right)dx\ . \tag{6}$$

Many variations of VanillaGAN have emerged since 2014. For example, α-GAN [7] was introduced in 2021 with its primary goal of creating a variety of loss functions parameterized by $\alpha > 0$. α-GAN has a tune-able loss function called the α-loss. These loss functions are expressed as a class probability estimation (CPE) loss between a real label and a predicted label. Being able to adjust α allows to one to apply a singular system to multiple datasets. Note that different datasets may perform optimally under different α values [7]. α-GAN has practical benefits as it addresses both the issues of vanishing gradients and mode collapse. It is worth noting that the α-GAN model has the capability to recover several other GAN variants, including VanillaGAN when α = 1.

α-GAN's concept was further extended in [7] to the concept of a dual objective (α_D, α_G)-GAN, where distinct α parameters can be used for the generator and discriminator loss functions. When α_D = α_G, optimizing α-GAN simplifies to minimizing an Arimoto divergence $D_{f_\alpha}(p_x||p_g)$, where

$$D_{f_\alpha}(p\,||\,q) = \frac{\alpha}{\alpha-1}\left(\int_S (p(x)^\alpha + q(x)^\alpha)^{\frac{1}{\alpha}}\ dx - 2^{\frac{1}{\alpha}}\right). \tag{7}$$

These alternative divergences can influence training stability, diversity in generated samples, and their effectiveness in dealing with mode collapse, where the model gets fixated on generating a limited set of patterns from the data, making them valuable tools for tailoring GANs to different applications and data characteristics. The selection of a divergence is often based on empirical experimentation, aligning with the problem's requirements and objectives.

3 Problem Description

3.1 Idea Generation

After conducting preliminary research into the mathematical and computational fundamentals of GANs, the team explored 3 distinct applications that could benefit from GAN-generated data. These applications are: fraud detection, synthetic financial data, and augmenting medical data.

3.1.1 Fraud Detection

Credit card fraud results in billions of dollars in losses per year and continues to grow as e-commerce and online payments become more prevalent [13]. Much of this cost leads to consumers paying higher fees from credit card companies and therefore higher prices for customers.

The effectiveness of fraud detection and prevention methods depends on the models that are trained with consumer information. Prediction models suffer from a lack of training data due to restrictions from privacy regulations, and companies limiting the sharing of their internally collected information.

Even given sufficient availability of data, fraud prevention models suffer from an imbalance in the training data since the class of fraudulent transactions is much smaller than its counterpart. A method called oversampling has been proposed to alleviate this bias [14]. Experiments have also shown that generating synthetic examples of this minority class using a GAN can improve classification of the dataset [14].

Another important consideration in binary classification problems is the asymmetric cost of misclassification. For example, labeling a transaction as potentially fraudulent when it is not will prove less costly than allowing definitely fraudulent transactions to go undetected. The Neyman-Pearson paradigm is a framework that is used to minimize the conditional probability of misclassifying an element of the minority class as one that is a part of the larger class [15]. It minimizes this error given an upper bound on the allowable error of the reverse misclassification. This is an important consideration in tuning models and finding the ideal trade-off for the specific application.

The team would utilize the Deep Regret Analytic GAN (DRAGAN), proposed in [16]. DRAGAN provides faster and more stable training, and is less likely to mode collapse. Experimental results presented in [17] show that this GAN variant is resistant to attacks and thus is a reasonable choice for this application.

3.1.2 Synthetic Financial Data

The accuracy and precision of financial machine learning models are dependent on the quality and quantity of data available to train the model. This contingency on large data sets quickly becomes a barrier to entry with financial data being heavily regulated. Proprietors of such resources severely limit access to the public and competitors due to its intrinsic value. Over the past several years financial institutions have steadily increased the amount of resources allocated to quantitative finance ventures to develop supervised, unsupervised, and reinforcement learning models, demonstrating the importance of quality data. Employing artificial intelligence to advise or automate market transactions serves to more efficiently analyze data, more accurately identify patterns, and reduce emotional bias, all while continuously learning.

Generating statistically consistent time series data allows developers to safely circumvent many of the regulations imposed by institutions while offering solutions to problems encountered during training. Traditional information is limited to scenario's that have occurred in the past, whereas synthetic estimates offer [18]:

- Tailor made solutions through adjusting the architecture and carefully selecting the training data for the GAN to replicate estimates in conditions where real data may be lacking or unavailable.
- High volume artificial test data reducing the impact of insufficient training.
- Improved risk management without depending on unreliable anonymization or encryption which impairs use cases.
- Enhanced collaboration without the distribution regulations.
- Greater degree of innovation and experimentation.

This application would invite the use of FIN-GAN, proposed in [19]. FIN-GAN was constructed to learn properties and structures of financial time-series data, and hence is a sensible choice of GAN for this application.

3.1.3 Augmenting Medical Data

Machine learning in biomedical segmentation tasks has the ability to vastly improve time and accuracy of diagnosis by employing automatic classification. A large issue with using machine learning in medical imaging is the difficulty of attaining large and supervised data sets. Limited training data inhibits the performance of supervised neural networks. The supervision, also referred to as labeling, is highly expensive in medical imaging as it involves expert observers [20]. GANs have the ability to create synthetic data sets with the appearance of real images in order to augment a dataset. Some studies that have been conducted in recent years include the generation of CT scans for the classification of brain lesions [20], and the generation of chest X-rays for pneumonia classification [21].

In order to ensure novelty, the team would explore augmenting medical data with the L_α-GAN, developed in [4]. This GAN has yet to be utilized in the medical imaging field, and thus it is an interesting endeavour to explore its capabilities in this sense.

3.2 Selection Process

When deciding on the application, the constraints of the project, choice of GAN, and team's own interests were considered. In order to make a fully informed decision, the team utilized a weighted evaluation matrix shown in Table 1 below. ”Interest” refers to how well the application matches with the team member's interests; ”Complexity” refers to the expected compute needed to train the GAN, which is affected by what type of data is used; ”GAN Model” indicates how well the team understands and how keen they are to use the type of GAN that the application requires; and ”Research” refers to how much relevant research has been completed that can enhance the project. The three applications were given a score 1, 2, or 3 for each criteria. 3 indicates the application which performs best in that criteria.

Criteria	Weight	Fraud Detection	Financial Data	Medical Data
Interest	25	1	2	3
Complexity	30	3	2	1
GAN Model	25	1	2	3
Research	20	2	1	3
Weighted Scores	1	180	180	240

Table 1: Weighted Evaluation Matrix

Evidently Augmenting Medical Data is the application of choice. The team expressed the most interest in this research avenue, and in the L_α-GAN for the choice of GAN variant. There is ample research to support the team's experiments, which are discussed in length below. The reason for the low ranking in complexity is due to the use of images for the data, whereas the other considered applications would make use of tabular and time-series data. Images require more compute power and thus will affect the needed compute budget and training time. The team determined that this added obstacle is manageable and thus not a cause for concern. The rest of this report will assume the application of Augmenting Medical Data, which will be furthered refined.

3.3 Problem Definition

Collection of real medical data is hindered by several factors. For one, the healthcare industry is subject to strict standards, especially pertaining to the collection, storage, and reporting of patient data [22]. Moreover, the safety of and costs felt by patients must also be considered.

MRI is a non-invasive imaging technique used by medical professionals in hospitals and clinics to obtain comprehensive soft tissue anatomical images [23]. A single MRI machine costs more than $1 million [23]. In Canada, the average cost of an MRI scan is $786 CAD, which is more than double the average cost of a computed tomography (CT) scan at $347 CAD [24].

Brain MRI scans are even more expensive than the average scan, at $786 [24]. Moreover, research and classification of brain scans have limitations due to the inherent class imbalance of these datasets. This imbalance comes from the majority of brains being healthy, and in the presence of a tumour, one type of brain tumour could be significantly more present in the dataset than others. This property restricts the effectiveness of deep-learning models for brain tumour classification [25].

With the above considered, there is a clear need for more balanced, and more readily accessible brain MRI scan datasets, that can be used to supplement medical research and improve the classification of brain anomalies, such as tumours. As such, GANs offer a favourable solution due to the ability to augment a dataset with synthetic data.

To address the problem, the project will employ L_α-GAN to create synthetic brain MRI scans to improve classification models. These synthetic scans adhere to the industry's commitment to patient privacy as the use of real scans will be limited; improve the issues of data storage as they can be generated as needed; and can allow for more balanced and more readily available datasets. L_α-GAN was chosen due to the unifying nature of the generator loss function, proven in [4], and since this particular GAN has not been applied to medical data to date, thus allowing for the novelty of results.

3.4 Related Works

3.4.1 Classical Data Augmentation

Many studies to date have used classical data augmentation to both increase the size and to reduce over fitting of a model. Shorten et al. survey the most common forms of classical data augmentation [26]. These include techniques such as geometric transformations, color space augmentation, kernel filters, mixing images, and random erasing. Geometric transformations refer to the act of rotating, scaling, or flipping the original images in the dataset. Color space augmentation refers to the act of changing the color scheme of an image to generalize the "lighting conditions" of the training dataset. This allows the neural network to become lighting invariant. Kernel filters refer to sliding an nxn matrix along the original image with either a Gaussian blur or a horizontal or vertical edge filter. Mixing images refers to averaging the respective pixel values of each image along their color channels and combining them into one image. Shorten et al. note how this is a counter-intuitive approach, however, it can be developed into an effective augmentation strategy. None of these techniques exploit deep learning, and thus have many limitations [26].

3.4.2 Augmentation using Neural Networks

In recent years, many researchers have been looking at moving past using classical data augmentation techniques and instead using neural networks [26]. Some notable techniques involve using GANs, neural style transfer, and meta-learning [26]. GANs have been shown to reduce overfitting in many image classification tasks by augmenting the dataset with new, generated unseen images. Hung et al. find success using a GAN to augment a small dataset [27]. The paper applies a slight modification to the original GAN by applying two transformation matrices during training which balance the quality and diversity requirements of the generated data. They determined that using the vanilla GAN did not improve the performance of an image classification task. This motivated the study of augmentation with the novel L_α-GAN. Bowles et al. where they use the Progressive Growing of GANs (PGGAN) architecture to augment the training dataset for CT scans [28]. They compare the accuracy results of a classification network (U-Net and UResNet) by applying either no augmentation, classical augmentation, PGGAN augmentation, or both. They conclude that the best results come from using both classical augmentation with PGGAN augmentation. This was a motivating factor to studying both classical and L_α-GAN augmentation combined in the subsequent analysis.

3.4.3 Image Classification Networks

Hung et al., and Motamed et al. used pre-existing classification models to test their augmentation methods in [27] and [29]. Respectively Hung et al. used AlexNet, GoogLeNet, ResNet, and VGGNet and compared accuracy results. Motamed et al. used a basic 4-layer convolutional network with a 2-layer fully connected backend. Since this paper had a dataset and method most similar to this project, we decided to use a similar simple classification network further discussed in Section 5.9.1.

3.4.4 Performance Analysis

Common performance analysis metrics used among these papers include F1, F2, ROC, AUPRC, DSC, and FID scores. It was concluded that F1, F2, and FID scores were most relevant, as well as generating a confusion matrix to better visualize false negatives and false positives. Additionally, Hung et al. and Bowles et al. used a student's t-test to measure the statistical significance of results, which is explained further in Section 6.2.3).

3.5 Constraints

Below is a breakdown of constraints involved in the execution of the project, including applicable standards, economic, environmental, cultural and societal considerations.

3.5.1 Applicable Standards

While there are no specific regulations dedicated solely to medical image synthesis using GAN models due to its recent conceptualization, it's necessary to adhere to broader regulatory frameworks concerning medical imaging and data privacy. Some key considerations include:

- Medical Imaging Regulations: The proposed solution falls under the classification of Class II Software as a Medical Device (SaMD) under the Food and Drugs Act (FDA) [30]. Therefore, the GAN model needs to be compliant with the 21st Century Cures Act of 2016 to adhere to FDA's principles [31], such as:

  1. Clear expectations on quality systems and good machine learning practices.
  2. Premarket assessment submission for SaMD products requiring it.
  3. Routine monitoring to determine when algorithm changes necessitate FDA review.
  4. Commitment to transparency in ongoing performance monitoring.

  Additionally, the FDA's guidance on imaging standards in clinical trials, released in 2018 provides nonbinding recommendations [32], including:

  1. Creating a charter specifying minimum requirements for vendor-specific upgrades within the trial period
  2. Establishing a process for acquisition quality control monitoring and data storage/transmission expectations
  3. Implementing robust archiving practices for images and interpretations to ensure data integrity.

- Data Privacy Laws: While there is currently no specific legislation addressing synthetic medical data, it's imperative to consider existing data protection regulations such as the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA), despite their limitations in fully addressing synthetic data risks [33].

3.5.2 Economic and Environmental Constraints, and Societal Considerations

- Cultural and Societal Considerations: While the project aims to enhance privacy protection through synthetic data, concerns regarding potential misuse still exist. These concerns are addressed in detail in Section 8.1.
- Environmental Constraints: The computational resources required for training and managing data involves energy consumption and was calculated to be 100 GPU-hours, which equates to 0.2 kW-hours of energy. A thorough discussion on the environmental impact, along with the benefits and drawbacks, can be found in Section 8.2.
- Economic Constraints: The project was allotted a budget of $500 and access to a supercomputer. A detailed breakdown of costs, including cost related to code adaptation, dataset processing, and power consumption, compared to costs related to traditional MRI scans can be found in Section 8.3.

4 Proposed Solution

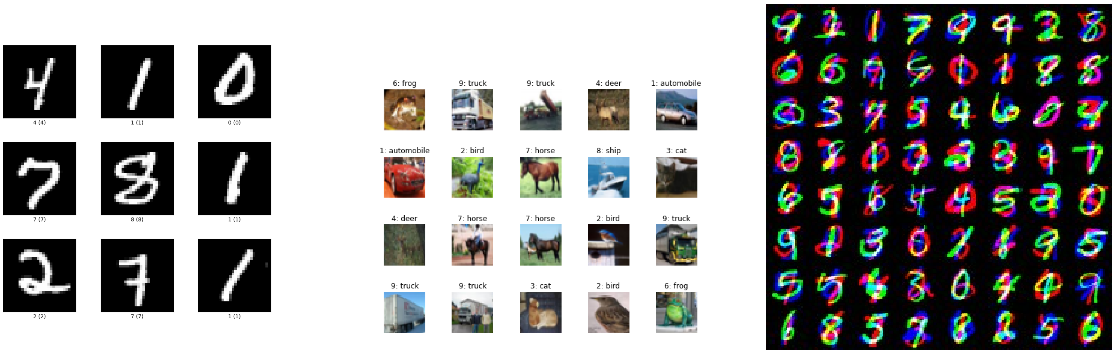
4.1 L_α-GAN Problem analysis and context

GANs are useful for many applications including generating image data and image resolution improvement. The objective of the project is to adopt and design GAN algorithms in [4] to increase the size of a data set to better train existing models.

We will take advantage of L_α-GAN, a model developed in [4], due to its ease of tuning hyper-parameters and generality. It is a higher complexity GAN compared to LkGAN. The performance complexity trade-off is acceptable due to the simplicity of the data being augmented.

The L_α-GAN, a dual-objective GAN that was built off the construction of the α-GAN [7], which combines a classical discriminator loss with a symmetric class probability estimation function. The generator's optimization in L_α-GAN minimizes a Jensen-f_α-divergence, a generalization of the Jensen-Shannon divergence, under an optimal discriminator [7]. This approach encompasses various GAN variants, including Vanilla-GAN, LSGAN, LkGAN, and (α_D,α_G)-GAN with D = 1 [40].

It was experimentally demonstrated on MNIST, CIFAR-10, and Stacked MNIST data sets in [4], as shown in Figure 7. The teams goal is to translate and adapt the code to alternative datasets.



(a) MNIST (b) CIFAR-10 (c) Stacked MNIST

Figure 7: Samples of Datasets Experimented on in [4]

The programming language utilized to train to L_α-GAN was Python. The existing L_α GAN code published to GITHUB in [4] was the base program and modifications were made to fit the specific application.

4.2 Engineering Tools

The primary engineering tool used for our model and simulations was Python. The team evaluated the Python tool and decided to move forward with it as the primary language as there existed previous L_α-GAN Python architecture put forth by Veiner in [4]. Other advantages considered by the team when considering the Python language were the positive of the extensive network of libraries build on Python and the high readability of the language and its ease of debugging. Another large advantage of Python is the compatibility of CUDA and Pytorch (Python compatible packages) and efficient NVIDIA GPU computation. NVIDIA built out the CUDA (Compute Unified Device Architecture) platform to expand the capabilities of CUDA compatible GPUs. The most computationally expensive operations: FID matrix multiplication and GAN discriminator and generator model creation/ use are greatly sped up by the use of this architecture.

The engineering tools the team utilized in our project along with thier respective versions are listed below.

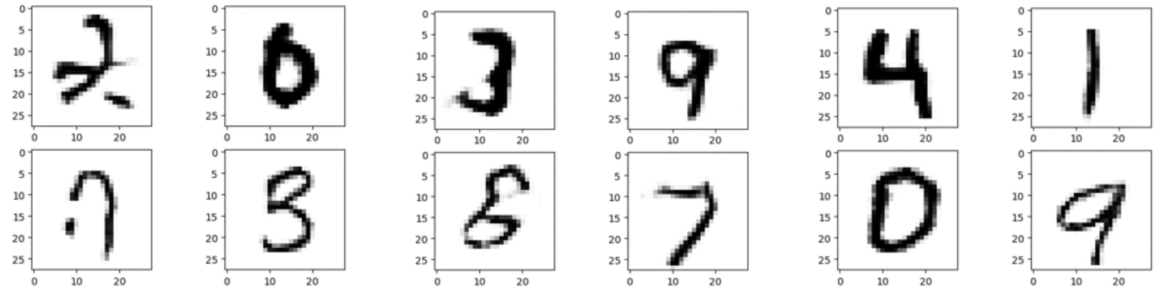
- Python v3.10.12
- Visual Studio code, v1.85.1
- Visual Studio code Remote - SSH, v0.107.1
- VPN, FortiClient
- GPU - 2 NVIDIA RTX A5500 chips
- CPU - AMD Ryzen Threadripper PRO 5975WX 32-Cores
- GitHub, the groups repository:`https://github.com/Kaleb-Huneau/GAN-Group6/tree/main`

The team's primary IDE was Visual Studio Code (VS Code), particularly leveraging its Remote - SSH extension, for collaborative coding on the local Math and Engineering computer machine. The Math and Engineering Linux machine has an x86_64 architecture with a 64-core AMD Ryzen Threadripper PRO 5975WX CPU. This CPU is capable of supporting both 32-bit and 64-bit op-modes. The CPU has 503GB of RAM and 3.5TB of storage. It operated on Ubuntu 22.04.4 LTS with a kernel version of 6.5.0-26-generic. Its architecture and specifications are well-aligned for programming demanding engineering analyses. The engineering trade-off of using VS Code vs other IDEs (PyCharm, Spyder, Xcode, etc.) was how lightweight the application is with the ability to add extensive plugins, like the .npy file viewer and the VS Code SSH extension which were utilized routinely. Additionally, the team utilized the FortiClient VPN, a requirement from Queen's, to remotely access their network and thus the supercomputer.

5 Design Implementation

5.1 α-GAN applied to MNIST and CIFAR-10

Implementation began by downloading the α-GAN code from [4] to the team's local environment. Various package dependencies and updates needed to be performed in order for the code to run. To ensure these resolutions maintained the integrity of the GAN, it was tested on the MNIST dataset consisting of grey-scale handwritten digits found in [41]. Figure 8 depicts the results, with $\alpha = 3$ and with increasing numbers of epochs:



(a) 10 Epochs, $\alpha = 3$ (b) 100 Epochs, $\alpha = 3$ (c) 250 Epochs, $\alpha = 3$

Figure 8: α-GAN-Generated Handwritten Digits

As expected, while increasing the number of epochs, the digits become cleaner and more recognizable. With 250 epochs, the digits can be easily determined. This preliminary experiment confirms that the L_α-GAN code was accurately localized and updated, and can now be used for the application at hand.

5.2 Dataset and Preprocessing

5.2.1 Dataset

The dataset being used to train the GAN is called ”Brain tumour MRI Dataset” found in [42]. This dataset contains 7023 images of human brain MRI images, which are classified into 4 classes: glioma, meningioma, pituitary, and no tumour. There are 1500 images of tumour-free brain images. The size of the images are not congruent in size, and hence preprocessing measures must be imposed.

5.2.2 Preprocessing

The images from the ”Brain tumour MRI Dataset” consisted of tumour MRI JPEGs [42] which varied in size and ratios. A Python script was developed to scale these images to the necessary dimensions. The ’crop image’ function was designed to specifically retain relevant regions of interest from the original images by converting them to grayscale, applying Gaussian blur for noise reduction and identifying the largest contour in the image. This preprocessing function ensured that the resulting images focus on the areas relevant to the analysis. With these cropped jpeg images, the ‘create dataset from directory’ function create of TensorFlow datasets from cropped image directories. These datasets could then be passed to the model to be trained on.

5.3 L$_\alpha$-GAN Variants

Since the L$_\alpha$-GAN can recover a continuum of loss functions, the cross validation procedure was extended to testing a range of hyperparameter values across several GAN variants for their respective tuning parameters. The variants considered were the Lk-GAN, LSGAN, Vanilla-GAN, α-GAN, Hellinger-GAN, and (α_D, α_G)-GAN. Each training iteration consisted of 250 epochs with a batch size of 64 and a gradient penalty of 5, using the no tumour subset with a gray scale image resolution of (64×64). Synthetic MRI scans were then produced by the generator model after 250 epochs, where the images shown here are amongst some of the more promising scans. It is important to note that all hyperparameters values of the neural networks themselves were held constant over all trials and assumed to be appropriate selections for the application.

For the following GAN discussions, each variant recovered from the L$_\alpha$-GAN framework will solve a type of f-divergence [4]. Consider a continuous convex function $f : [0,\infty) \to \mathbb{R}$ such that $f(1) = 0$. The f-divergence between two probability densities p and q, with common support $\mathscr{R} \subseteq \mathbb{R}^d$, on the Lebesgue measurable space $\{\mathscr{R}, \mathscr{B}(\mathscr{R}), \mu\}$, is denoted by $D_f(p||q)$ [4], defined as

$$D_f(p||q) = \mathbb{E}_{A\sim q}[f(\frac{p(A)}{q(A)})] = \int_{\mathscr{R}} qf(\frac{p}{q})d\mu. \tag{8}$$

Examples of f-divergences under various choices of their generating functions f will be presented in the following analysis and will refer to the above definition.

The L$_\alpha$-GAN model capitalizes on the flexibility of the α-GAN model, to explained in more detail in this section, in conjunction with a conditional probability estimation loss function, $\mathscr{L}_\alpha$ [4].

The mathematical description of the L$_\alpha$-GAN applied to our data set requires defining the measure space $(\mathscr{X},\mathscr{B}(\mathscr{X}),\mu)$ of $n\times n\times 1$ resolution images. After pre-processing, this space contains all cropped $64\times 64\ \times 1$ gray-scale MRI brain scans. The discriminator neural network is defined as $D:\mathscr{X}\to[0,1]$, and the generator neural network is defined as $G:\mathscr{Z}\to\mathscr{X}$, where $\mathscr{Z}\subseteq\mathbb{R}^d$ [4]. We define another measure space $(\mathscr{Z},\mathscr{B}(\mathscr{Z}),\mu)$, which translates to the space over which the generators input noise will be sampled according to a multivariate Gaussian distribution $P_Z:\mathscr{Z}\to[0,1]$ [4]. The pixel distributions of the real MRI scans and generated MRI scans are defined as $P_X:\mathscr{X}\to[0,1]$, and $P_G:\mathscr{X}\to[0,1]$, respectively. Now, for a fixed $\alpha\in\mathbb{R}$, the symmetric function $\mathscr{L}_\alpha$ can be defined as $\mathscr{L}_\alpha:\{0,1\}\times[0,1]\to[0,\infty)$. The all encompassing $\mathscr{L}_\alpha$-GAN system is described by the $\mathscr{L}_\alpha$-GAN optimization,

$$\sup_D V_D(D,G), \tag{9}$$

$$\inf_G V_{\mathscr{L}_\alpha,G}(D,G), \tag{10}$$

where $V_D:\mathscr{X}\times\mathscr{Z}\to\mathbb{R}$ is the discriminator loss function, and $V_{\mathscr{L}_\alpha,G}:\mathscr{X}\times\mathscr{Z}\to\mathbb{R}$ is the generator loss function. The generator loss function can be express explicitly as [4]

$$V_{\mathscr{L}_\alpha,G}(D,G) = \mathbb{E}_{A\sim P_X}[-\mathscr{L}_\alpha(1,D(A))] + \mathbb{E}_{B\sim P_G}[-\mathscr{L}_\alpha(0,D(B))]. \tag{11}$$

However, more regularity conditions must be known to recover a general expression for the discriminator loss function.

5.3.1 LkGAN

The LkGAN's loss function is derived from the absolute error distortion measure of order $k\geq 1$, $k\in\mathbb{R}$, with the goal being to minimize the distortion between the data samples and a target value that the discriminator will assign the samples to [4]. This GAN model recovers the Pearson Vajda divergence of order k,

$$D_{|\mathscr{X}|^k}(p||q) = \int_{\mathscr{R}} \frac{|q-p|^k}{p^{k-1}}d\mu, \tag{12}$$

$$f(u) = u^{1-k}|1-u|^k, \tag{13}$$

and the LkGAN problem is the optimization of the pair,

$$\sup_D V_{LsGAN,D}(D,G), \tag{14}$$

$$\inf_G V_{k,G}(D,G). \tag{15}$$

In the joint optimization problem, for $\gamma,\beta,c\in[0,1]$, and $k\geq 1$, the LkGAN's loss functions, are defined as [4],

$$V_{LsGAN,D}(D,G) = -\frac{1}{2}\mathbb{E}_{A\sim P_X}[(D(A)-\beta)^2] - \frac{1}{2}\mathbb{E}_{B\sim P_G}[(D(B)-\gamma)^2], \tag{16}$$

$$V_{k,G}(D,G) = \mathbb{E}_{A\sim P_X}[|D(A)-c|^k] - \mathbb{E}_{B\sim P_G}[|D(B)-c|^k]. \tag{17}$$

Note the discriminator loss function in the LkGAN formulation is shared with the LSGAN, where instead of using the absolute error distortion measure of order $k \geq 1$, the squared error distortion is used.

The LkGAN was tested for all permutations of the set of k values $\{1,4,8\}$, the set of γ values $\{1,0.75\}$, and the set of β values $\{0,0.25\}$, and a set of c values $\{0,1\}$. The most promising scan belonged to the training iteration with set of hyperparameters $\{k=1,\beta=0,\gamma=0.75,c=1\}$, as seen in Figure 9.

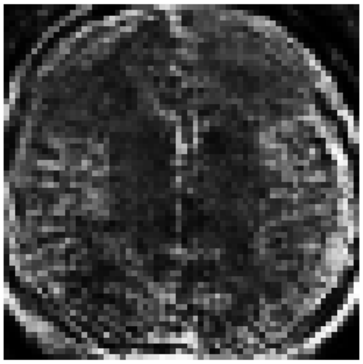


Figure 9: Highest fidelity to input MRI scans produced by the LkGAN $\{k=1,\beta=0,\gamma=0.75,c=1\}$

The decreased β value, and the typical γ value indicates that the LkGAN produced more realistic images when the discriminator was less aggressive when penalizing the the generators output, while the generator learned from the fully penalized score of the generator. This implies for optimal results for the LkGAN, the discriminator had to be inhibited from becoming too good at detection, while the generator had to learn from the fully penalized score, rather than the score of the inhibited model. The value of $k=1$ suggests that the best model relied on a pixel-wise loss with the L_1 norm.

5.3.2 LSGAN

As previously mentioned, the LSGAN is recovered from the LkGAN model for a subset of hyperparameter values where $k=2$, hence it shares many similarities with the general case LkGAN, however, lacks the additional degree of freedom in the generator loss function [4]. The LSGAN shares the same optimization problem as described for the LkGAN (14)(15), with almost identical sets of loss functions for hyperparameters $\gamma,\beta,c[0,1]$,

$$V_{LsGAN,D}(D,G) = -\frac{1}{2}\mathbb{E}_{A\sim P_X}[(D(A)-\beta)^2] - \frac{1}{2}\mathbb{E}_{B\sim P_G}[(D(B)-\gamma)^2], \tag{18}$$

$$V_{k,D}(D,G) = \mathbb{E}_{A\sim P_X}[|D(A)-c|^2] - \mathbb{E}_{B\sim P_G}[|D(B)-c|^2]. \tag{19}$$

Again, the LSGAN recovers the Pearson Vajda divergence, this time of order $k=2$

$$D_{\mathscr{X}^2}(p||q) = \int_{\mathscr{R}} \frac{|q-p|^2}{p} d\mu, \tag{20}$$

$$f(u) = (\sqrt{x} - \frac{1}{\sqrt{x}})^2. \tag{21}$$

The LSGAN was tested for all permutations of the set of γ values $\{1,0.75\}$, the set of β values $\{0,0.25\}$, and a set of c values $\{0,1\}$. The best scan produced by the LSGAN model belonged to the training iteration with set of hyperparameters $\{\beta = 0, \gamma = 0.75, c = 1\}$, as seen in Figure 10, which corroborates the findings for the general case LkGAN model. The rationale supporting these results follows a similar argument to that in the previous section. This may also suggest that the order selection for the Pearson Vajda divergence for orders 1 and 2 is insignificant.

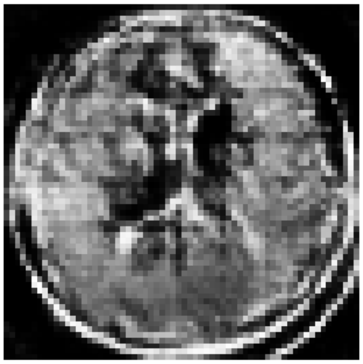


Figure 10: Highest fidelity to input MRI scans produced by the LSGAN $\{\beta = 0, \gamma = 0.75, c = 1\}$

5.3.3 α-GAN

The α-GAN introduces a parameter α, to control the balance between the generator's adversarial training and reconstruction loss, proportional to the balance between image quality and diversity. This model solves the Arimoto divergence for $\alpha > 0, \alpha \neq 1$ [4]

$$D_{\mathscr{A}_\alpha}(p||q) = \frac{\alpha}{\alpha-1}\left(\int_{\mathscr{R}} (p^\alpha + q^\alpha)^{\frac{1}{\alpha}} d\mu - 2^{\frac{1}{\alpha}}\right), \tag{22}$$

$$f(u) = \frac{\alpha}{\alpha-1}\left((1+u)^{\frac{1}{\alpha}} - (1+u) - 2^{\frac{1}{\alpha}} + 2\right). \tag{23}$$

The flexibility of the α-GAN conveniently unifies several existing GANs using a parameterized loss function. For $y \in \{0,1\}$ binary label, $\hat{y} \in [0,1]$, fixed $a > 0$, the α-loss between y and $\hat{y}$ is the map $l_\alpha : \{0,1\} \times [0,1] \rightarrow [0,\infty)$ given by [4]

$$l_\alpha(y,\hat{y}) = \begin{cases} \frac{\alpha}{\alpha-1}(1 - y\hat{y}^{\frac{\alpha-1}{\alpha}} + (1-y)(1-\hat{y})^{\frac{\alpha-1}{\alpha}}), \text{ for } \alpha \in (0,1) \cup (1,\infty) \\ -y\log(\hat{y}) - (1-y)\log(1-\hat{y}), \text{ for } \alpha = 1 \end{cases}, \tag{24}$$

where the loss function for the optimization of the α-GAN problem [4],

$$\inf_G \sup_D V_\alpha(D,G), \tag{25}$$

can be expressed explicitly as [4]

$$V_\alpha(D,G) = \mathbb{E}_{A\sim P_X}[-l_\alpha(1,D(A))] + \mathbb{E}_{B\sim P_G}[-l_\alpha(0,D(B))]. \tag{26}$$

Hence why it is the foundation of the L$_\alpha$-GAN framework. Although some of the GANs previously mentioned are described as being recovered from models other than the α-GAN, it is possible to reformat the derivation to show recovery from the α-GAN model, with some slight modifications.

The α-GAN was tested over the range of values $\{0.25, 0.75, 1.25, 1.75, 2, 4, 16, 64\}$. The most promising scan generated by the α-GAN model belonged to the training iteration with the set of hyperparameter values$\{\alpha = 0.75\}$, as seen in Figure 11. This suggests that the α-GAN model produced the most realistic images with a relatively small α value which is a result of the generator emphasizing the reconstruction loss and increasing the fidelity to the original images. A high α in this model would instead emphasize the reconstruction loss, increasing the degree of diversity amongst synthetic samples.

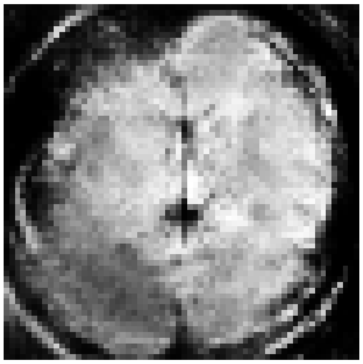


Figure 11: Highest fidelity to input MRI scans produced by the α-GAN $\{\alpha = 0.75\}$

5.3.4 Vanilla-GAN

The original GAN, also referred to as Vanilla-GAN, can be recovered from the α-GAN model for a subset of parameters $\{\alpha = 1\}$. The Vanilla-GAN uses the Jensen Shannon divergence [4]

$$D_{JS}(p||q) = \frac{1}{2}D_{KL}(p||\frac{p+q}{2}) + \frac{1}{2}D_{KL}(q||\frac{p+q}{2}), \tag{27}$$

$$f(u) = \frac{1}{2}(u\log(u) - (u+1)\log(\frac{u+1}{2})). \tag{28}$$

where D_{KL} denotes the Kullback-Leiber divergence [4], defined as

$$D_{KL}(p||q) = \int_{\mathscr{R}} p\log(\frac{p}{q})d\mu, \tag{29}$$

$$f(u) = u\log(u). \tag{30}$$

The Vanilla-GAN solves the optimization problem

$$\inf_G \sup_D V_{VG}(D,G). \tag{31}$$

The typical discount of 1 is imposed on the discriminator component, and 0 on the generator component, of the discriminators loss function. The typical discount of 1 is imposed on the generators loss function. Outside of the hyperparamaters common to all neural networks and consequently GANs, there exist no other controllable parameters resulting in the traditional binary cross-entropy loss used for both the generator and discriminator [4].

$$V_{VG}(D,G) = \mathbb{E}_{A\sim P_X}[-\log(D(A))] + \mathbb{E}_{B\sim P_G}[-log(1-D(B))]. \tag{32}$$

As a result, only one trial was performed, and the image generated can be seen in Figure 12.

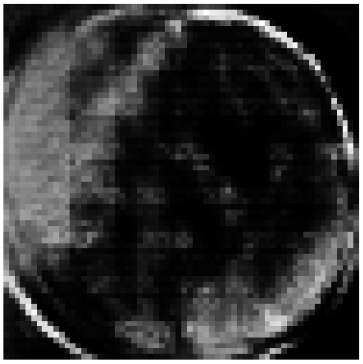


Figure 12: Highest fidelity to input MRI scans produced by the Vanilla-GAN

5.3.5 Hellinger-GAN

The Hellinger-GAN is a special case of the α-GAN for a subset of parameters $\{\alpha = \frac{1}{2}\}$. Similarly, this model solves the Arimoto divergence, for constant $\alpha = \frac{1}{2}$ [4]

$$D_{\mathscr{A}_{\alpha=\frac{1}{2}}}(p||q) = \frac{1}{\alpha-1}(\int_{\mathscr{R}} p^{\alpha} q^{1-\alpha} d\mu - 1), \tag{33}$$

$$f(u) = \frac{u^{\alpha}-1}{\alpha-1}. \tag{34}$$

The Hellinger-GAN solves the optimization problem

$$\inf_{G} \sup_{D} V_{\alpha}(D,G). \tag{35}$$

Similar to the Vanilla-GAN, outside of the hyperparamaters common to all neural networks and consequently GANs, there exists no other controllable parameters. The resulting loss function uses the mapping l_{α} : $\{0,1\} \times [0,1] \rightarrow [0,\infty)$ described in (24) [4], and can be expressed as,

$$V_{\alpha=\frac{1}{2}}(D,G) = \mathbb{E}_{A\sim P_X}[-l_{\frac{1}{2}}(1,D(A))] + \mathbb{E}_{B\sim P_G}[-l_{\frac{1}{2}}(0,D(B))]. \tag{36}$$

Similar to the Vanilla-GAN, the Hellinger-GAN model was only tested for one trial, and the image generated can be seen in Figure 13.

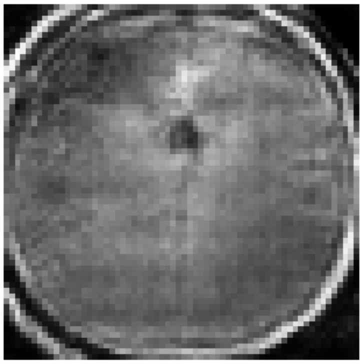


Figure 13: Highest fidelity to input MRI scans produced by the Hellinger-GAN

5.3.6 (α_D, α_G)-GAN

The (α_D, α_G)-GAN is an extension of the α-GAN, where another alpha parameter is introduced to control the discriminators balance between adversarial training and reconstructive loss, denoted α_D, to accompany the parameter found the the α-GAN model, denoted α_G, which controls the generators balance between adversarial training and reconstructive loss [4]. Intuitively, the α_D parameter controls the balance between degree of strictness, versus the magnitude of the penalty, and α_G controls the balance between the degree of diversity among synthetic samples, versus the fidelity to the original images. This model solves the Arimoto divergence, as defined in (22), twice for α_D and α_G [4], with the generating function defined for $\alpha_D, \alpha_G > 0$ being

$$f_{\alpha_D,\alpha_G}(u) = \frac{\alpha_G}{\alpha_G - 1}\left(\frac{u^{\alpha_D(1-\frac{1}{\alpha_G})+1} + 1}{(u^{\alpha_D} + 1)^{1-\frac{1}{\alpha_G}}}\right). \tag{37}$$

The (α_D, α_G)-GAN's optimization is described by [4]

$$\sup_D V_{\alpha_D}(D, G), \tag{38}$$

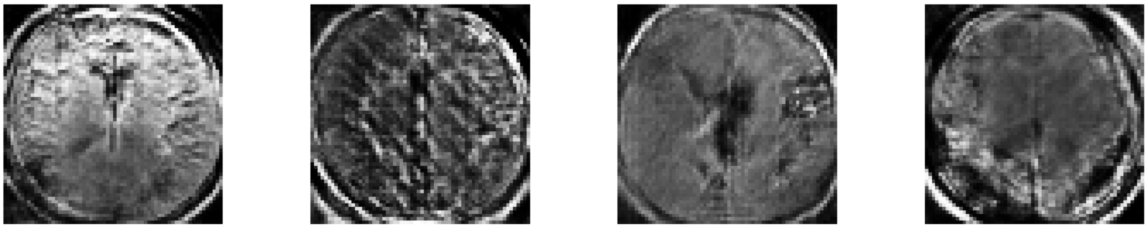
$$\inf_G V_{\alpha_G}(D, G), \tag{39}$$

where $V_{\alpha_D}(D, G)$ and $V_{\alpha_G}(D, G)$ are defined as [4]

$$V_{\alpha_D}(D, G) = \mathbb{E}_{A \sim P_X}[-l_{\alpha_D}(1, D(A))] + \mathbb{E}_{B \sim P_G}[-l_{\alpha_D}(0, D(B))], \tag{40}$$

$$V_{\alpha_G}(D, G) = \mathbb{E}_{A \sim P_X}[-l_{\alpha_G}(1, D(A))] + \mathbb{E}_{B \sim P_G}[-l_{\alpha_G}(0, D(B))]. \tag{41}$$

The (α_D, α_G)-GAN model was tested for all permutations over sets of values of for α_D, and α_G, equivalent to $\{0.25, 0.5, 0.75, 1, 2, 4, 16, 32, 64\}$. The most promising scan belonged to the training iteration with set of hyper parameters, $\{\alpha_D = 0.75, \alpha_G = 0.75\}$. In general, it was observed that results were optimal for values of $\alpha_D \in (0, 2]$, and $\alpha_G \in (0, 64)$. Results were slightly more accurate when restricting the range of values for α_G to $(0, 2]$. The images produced for the ranges of α_D and α_G can be seen below in Figure 14



(a) $\alpha_D \in (0, 2], \alpha_G \in (0, 2]$

(b) $\alpha_D \in (0, 2], \alpha_G \in (2, 64]$

(c) $\alpha_D \in (2, 64], \alpha_G \in (0, 2]$

(d) $\alpha_D \in (2, 64], \alpha_G \in (2, 64]$

Figure 14: Highest fidelity to input MRI scans produced by the (α_D, α_G)-GAN

5.3.7 (64×64) Analysis

To determine which GAN variant produced the most promising scans, an iterative selection process was followed. For each variant, several images were generated by all saved generator models for each permutation of hyperparameter values, after 250 epochs. The best MRI scan from each subset of generated images for a particular variant were subsequently compared and the best was chosen by a visual examination. Although a more scientific method of cross validation parameter selection would be to systematically assess the scores of each generated image via some performance metric, due to time and computational limitations, the team opted to use visual inspection. This was justified since in most cases it was apparent which models produced the higher quality MRI scans. The criteria upheld by what were considered the more promising images was twofold: generator model capable of retaining the most features from the original scans, and absence of excessive noise (essentially just high fidelity to original images). Here, diversity amongst samples was omitted as a criterion as this would be a more challenging thing to assess without a more refined selection procedure. Once the optimal set of hyperparameters had been selected for each variant, the best images generated by the best models with the optimal set of hyperparameters for each GAN variant were then subject to another eye test to determine the optimal variant for our application. The (α_D, α_G)-GAN was selected as the best variant since the best images generated by this model appeared to have the highest fidelity to the original MRI scans relative to all other models.

With these findings concluding the first round of broad cross validation, the (α_D, α_G)-GAN would undergo another round of cross validation for higher resolutions and finer increments between values in the sets of hyperparameters to verify whether the hypothesis at a lower resolution would still hold.

5.4 Finer Tuning for (α_D, α_G)-GAN

After scaling the generator network and discriminator network, and adjusting the pre-processing reshaping, finer cross validation for hyperparameter selection began for the (α_D, α_G)-GAN. An identical process was followed in the second round of cross validation as was followed in the first round, with the only differences being the necessary adjustments to accommodate input and generated images of a larger resolution. Parameter values for α_D were selected from $\{0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4\}$and α_G were selected from the set $\{0.25, 0.5, 0.75, 1, 2, 3, 4, 5, 10\}$. These values were chosen in light of the findings from the previous round where it had been discovered that the optimal α_D value belonged to the range $[0, 2]$, and the optimal α_G value belonged to the range $[0, 10]$. Again, one model was fitted for each permutation of α_D, α_G values with each training iteration consisting of 250 epochs with a batch size of 64 and a gradient penalty of 5, using the no tumour subset with a gray scale image resolution of 128×128. Synthetic MRI scans were then produced by the generator model after 250 epochs. It is important to note that all hyperparameters values of the neural networks themselves were held constant over all trials and assumed to be appropriate selections for the application. Following this set of trials, four of the most promising generator models were selected according to the same selection process and criteria used in the first round. The set of pairs of hyperparameters belonging to the most promising (α_D, α_G)-GAN models was $\{(1, 0.5), (1, 5), (1, 10)\}$. These results suggest a definitive value for the optimal α_D parameter of 1, whereas a range of values for α_G were found to be near optimal. These four models were then subjected to another round of cross validation, this time holding the hyperparameters constant in each model but changing the neural network parameters. The learning rate for the networks was selected from the set of values $\{0.01, 0.001, 0.005, 0.0001\}$, and the batch size was chosen from the set $\{64, 128, 256\}$. Each of the four models were fitted for each permutation of network parameters. The results of this final round of cross validation will be discussed in section 5.8 concerning the x of these models as each generator model was then subjected to more rigorous evaluation metrics.

5.5 Generator Model

The generator used a basic Convolutional layer frontend combined with a fully connected backend. Figure 12.2 depicts a detailed schematic of the generator model. The model was based off the model by Veiner et al. [4]. The only changes made were to accommodate the 128x128 image inputs required for the MRI images, as compared to the smaller image size for the MNIST, CIFAR10, and stacked MNIST datasets used in [4].

5.6 Summary of Design Process

To document the design process followed in prior sections, an iterative approach was followed to gradually refine our solution. This process was segmented into stages corresponding to the work that was done to develop the solution at each of four MRI scan resolutions.

At an image resolution of (32×32) seen in Figure 15, the necessary modifications were made to Veiner's L_α-GAN framework to accommodate our MRI scan dataset. This included configuring the GAN and resolving dependencies, developing the new pre-processing method, and other initial modifications. The previous work of [4] had been successfully replicated on the datasets which they had previously tested. Then using that same architecture the MRI dataset was pre-processed and the new simulations began on the Google Colab Platform, where the team successfully generated (28×28), and (32×32)resolution images. This engineering decision to leverage the Google Colab Platform was driven by the high accessibility and ease of collaboration.

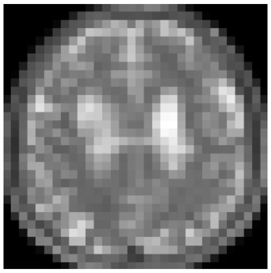


Figure 15: Sample gray-scale (32×32) MRI scan

At an image resolution of (64×64) seen in Figure 16, an additional convolution layer was required by the generator and the input layer of the discriminator network required reshaping to accommodate the larger MRI scans. This increase in resolution was accompanied by memory allocation issues with the limited space available on Google Colab, which catalyzed the migration of the working repository to the Math and Engineering machine to continue with training. This is justified given the details in 4.2 . At this resolution the first round of cross validation for GAN variants and their respective hyperparameters was conducted due to the jump in computational complexity for any subsequent increase in resolution significantly reducing the completion speed of training iterations. Shell scripts were developed to recursively train each variant model for a range of hyperparameter values, and to generate synthetic MRI scans from each variant model to assess performance.

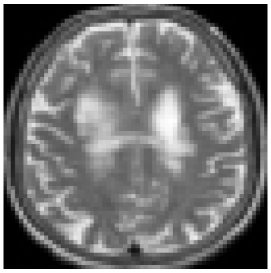


Figure 16: Sample gray-scale (64×64) MRI scan

At an image resolution of (128×128) seen in Figure 17, another additional convolution layer was required by the generator and the input layer of the discriminator network required reshaping to accommodate the larger MRI scans. At this resolution, finer tuning of the decided best model, the (α_D, α_G)-GAN, was conducted.

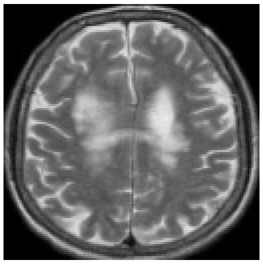


Figure 17: Sample gray-scale (128 × 128) MRI scan

At an image resolution of (256 × 256) seen in Figure 18, memory allocation issues were encountered again, this time with the Math and Engineering Machine, indicating the team had likely reached the upper bound of feasible image resolutions. It was decided that the final model would use input MRI scans of (128 × 128) resolution. Theoretically, with another additional convolution layer in the generator network and reshaping the input layer to the discriminator network to accommodate the larger MRI scans, images of (256 × 256) resolution could be used with greater computing capacity.

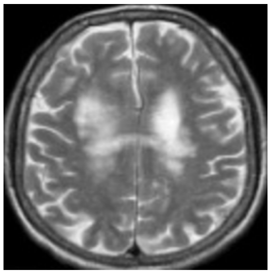


Figure 18: Sample gray-scale (256 × 256) MRI scan

5.7 Assumptions

Listed below are the major assumptions made in the design process. Less crucial assumptions are listed throughout the design process.

- MRI dataset is labeled correctly.
- Duplicates in the dataset have marginal effect on model performance.
- Three GAN model runs is sufficient to evaluate success (low variation in FID).
- Five classification model runs is sufficient to evaluate success (low variation in classification accuracy and F2 score).

5.8 Augmenting the Dataset

Since a conditional GAN was not used, it was required to separately train a model on positive-tumour MRI images and negative-tumour MRI images. Synthetic images were then labeled based on what images the GAN was trained on. To augment the dataset, 10,000 synthetic images were generated of either positive-tumour or negative-tumour images. Table 2 details the datasets used. After creating the synthetic datasets for each respective α_D and α_G value, the models were trained, discussed in 5.9. Note that in the subsequent analysis, the discussion on the augmented negative-tumour and augmented positive-tumour datasets is omitted since the results were for experimentation purposes and not relevant to the final discussion. Only the base dataset and the augmented both datasets were used for subsequent analysis.

Dataset	Real Negative-tumour	Real Positive-tumour	Synthetic Negative-tumour	Synthetic Positive-tumour	Total Images
Base Dataset	1,595	4,117	0	0	5,712
Augmented Negative-tumour	1,595	4,117	10,000	0	15,712
Augmented Positive-tumour	1,595	4,117	0	10,000	15,712
Augmented Both	1,595	4,117	10,000	10,000	25,712

Table 2: Dataset Details

5.9 Classification Model

5.9.1 Architecture

Image classification models have greatly improved in recent years thanks to the Image Large Scale Vision Recognition Challenge (ILSVRC). The ILSVRC is a yearly computer vision conference which has produced state-of-the-art image classification networks such as AlexNet, Res-Net, and VGG being introduced there [37]. As discussed in Section 2.2, convolutional neural networks are commonly used in computer vision tasks, and are invariant to scale and translation features [35]. The application did not require the use of any of the complex the state-of-the-art models from the ILSVRC, however, many of the techniques learned from the conference were used. It is worth noting that the original analysis was done using Res-Net and VGG but the model would overfit to the MRI dataset, since the number of images were less than recommended for such complex models.

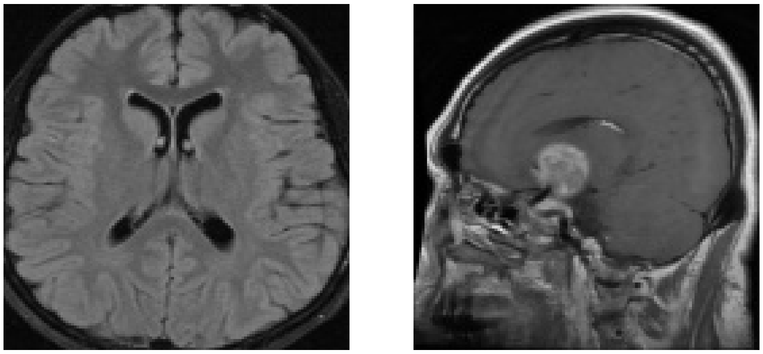
The final model was called MonkeyNet and consisted of 3 convolutional layers and 2 fully connected layers. A ReLU activation function was used at each convolutional layer and the first fully connected layer, and then a final activation of Sigmoid was used. In preliminary tests, the model seemed to overfit even with such a small network due to the small size of the original dataset. To mitigate this, dropout was used with a probability of 0.5 at each fully connected layer. Dropout is a technique in which each node has a certain probability (in this model 0.5) of being inactive during training. This has been shown to reduce overfitting [36]. Batch normalization was also used since it has been shown to increase stability in training and allow for higher learning rates, leading to faster convergence [38]. Batch normalization refers to normalizing the inputs at each image by the mean and variance of the batch. It is calculated as,

$$\hat{x}_{i,k} = \frac{x_{i,k} - \mu_k}{\sqrt{\sigma_k^2 + \varepsilon}},$$

where $\hat{x}_{i,k}$ is the normalized value of the k-th feature for the i-th input, μ_k is the mean of the batch, σ_k^2 is the variance of the batch, and ε is a small constant added for numerical stability. When training MonkeyNet, it was found that batch normalization accelerated convergence. A ReLU activation was used at each hidden layer, as is common in the literature for an image classification task [39]. ReLU introduces a non-linearity to the model thereby increasing its ability to *fit* to the dataset. Sigmoid was used in the last layer of the network. This is a commonly used activation function in the last layer of a binary classification network [39]. See Section 2.1 for a definition of the ReLU and Sigmoid functions. See Section 12.3 for a detailed outline of the model's architecture.

5.9.2 Classification Task

The task was a binary classification task with an image labeled either ”tumour” or ”no-tumour”. Real images were pre-labeled from the dataset directly. Images from the no-tumour folder were given label 0 and images from the tumour folder were given label 1. A sample of two (128 × 128) MRI images, one with tumour and one without can be seen in Figure 19 with their corresponding labels.



Label = 0

Label = 1

Figure 19: Real MRI images, one no-tumour, one tumour

5.9.3 Creating Simulations and Tuning Hyper-parameters

Before beginning the main task, the optimal hyper-parameters were found for the particular application. The learning rates and batch sizes used for testing are shown in Table 3 below.

Learning Rates	Batch Sizes
0.01, 0.001, 0.005, 0.0001	64, 128, 256

Table 3: Learning rates and batch sizes used for testing

All combinations of the above learning rates and batch sizes were trained to find the optimal combination. There does not exist a analytic method to determine the optimal learning rate or batch size, and therefore it was done heuristically. It was determined that the optimal combination was having a learning rate of 0.005 with a batch size of 64. Figure 20 shows the accuracy and loss of the optimal hyper-parameter configuration over 100 epochs. The blue line is for training data, and the orange line is for testing data. The *testing* loss and accuracy are the important metrics to track since the testing data is new unseen data, whereas the training data is what the neural network has used as inputs for optimization.

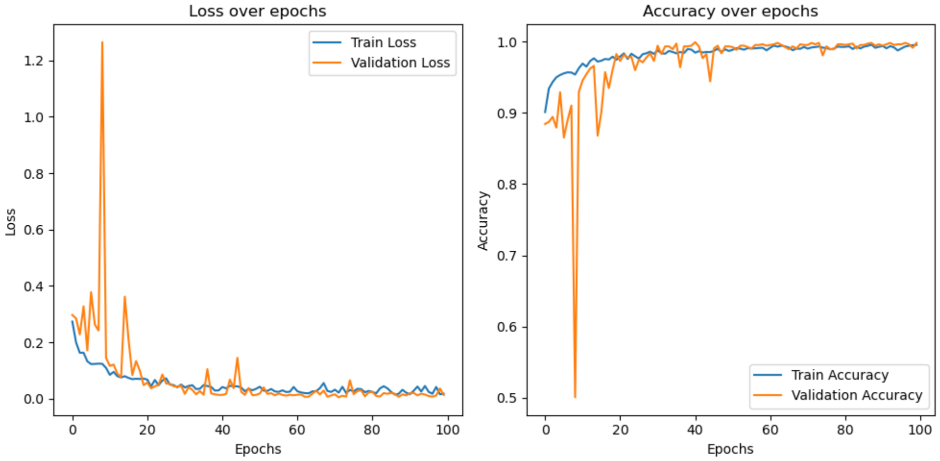


Figure 20: Loss and Accuracy over Epochs for lr $= 0.005$, batch size $= 64$

Figure 20 shows that the test accuracy and loss closely follow the training accuracy and loss, which is an indication of a perfect fit. With a higher learning rate, there was more variation and trouble with convergence in the loss and accuracy plots. With a lower learning rate, there was slow convergence and a large gap between the training and testing accuracy and loss. Among batch sizes, there was a negligible difference in results between 64, 128, and 256, and thus a batch size of 64 was chosen based on GPU memory constraints. In the analysis of results in 6.2, the model used was MonkeyNet, with a learning rate of 0.005, and a batch size of 64.

6 Performance Evaluation

A major limitation in applying typically effective classification models to medical application is their imbalanced nature yet high cost to generate real images. Therefore, the evaluation of the design will be conducted by comparing the results using the dataset balanced using synthetic, GAN generated images to the original unbalanced dataset. In particular, the classification accuracy, F1 scores and F2 scores will be compared using the classification model outlined in Section 5.9.

Additionally, for the design to be successful, the GAN generated images must be of high enough quality to be visually intelligible to humans. This criteria will increase the widespread applicability of the results by allowing for the GAN generated image labels to be validated by medical professionals for applications such as training of doctors–without typical medical privacy concerns–in addition to their use in data augmentation.

6.1 Performance of (α_D, α_G)-GAN

The metric utilized to assess the efficacy of the (α_D, α_G)-GAN model was the Fréchet Inception Distance (FID) score, which is calculated as,

$$\text{FID} = \|\mu_{\text{real}} - \mu_{\text{fake}}\|_2^2 + \text{Tr}(\Sigma_{\text{real}} + \Sigma_{\text{fake}} - 2\sqrt{\Sigma_{\text{fake}} \cdot \Sigma_{\text{real}}}). \tag{42}$$

The FID score provides a quantitative measure the results from (α_D, α_G)-GAN models. This quantifies the performance by measuring the similarity between the real and model generated images. As seen in Equation 42 the FID calculation incorporates information regarding the mean and covariance for the image distributions. Despite its utility, the FID score neglects the positives of diversity within the generated image set. Additionally, due to the large scale of image set, (128x128), the covariance calculation consisted of a matrix multiplication of two square 16384 matrices. This is an extremely computationally expensive operation to run and could not be integrated into the model for computation every epoch as was done in [4]. Instead the team took an approach of saving the previous discriminator and generator models every 10 epochs and implemented a separate Python script to evaluate the FID scores of the models.

Building off the experiments put forward by Veiner et al [4] inspired to the choice of (α_D, α_G) values. Trials began on lower dimensional MRI images of (28x28). As the dimension and complexity of the image increased the associated FID score scaled as well. A successful trial was defined as one where the model could generate images which resembled an MRI scan. Models were created using both the no-tumour dataset and tumour dataset (Meningioma, Pituitary and Glioma) together.

Table 4: (α_D, α_G)-GAN results for 128x128 tumour MRI scans at 300 epochs.

(α_D, α_G)-GAN	Best FID	Number of Successful Trials $(/3)$
$(0.5, 0.5)$-GAN	363	2
$(1, 0.5)$-GAN	231	3
$(1, 1)$-GAN	466	0
$(1, 5)$-GAN	**105**	3
$(1, 10)$-GAN	146	3

The results presented in Table 4 show the lowest FID score was achieved with the (1,5)-GAN. There was frequent convergence with help from the use of gradient penalty in the model. The engineering trade-off of higher computational complexity was worth it for the increase in the convergence rate. The (1,1)-GAN did not converge over our multiple trials. It produced noisy images from suspected mode collapse as this was the case from epoch 60 onwards.

Table 5: (α_D, α_G)-GAN results for 128x128 no-tumour MRI scans at 300 epochs.

α_D, α_G-GAN	Best FID	Number of Successful Trials $(/3)$
$(1, 1)$-GAN	503	0
$(1, 5)$-GAN	**112**	3
$(1, 10)$-GAN	181	3

Similar FID results to those in Table 4 came from the models generated from no-tumour datasets.
Figure 21 shows the FID results for the (1.0, 5.0)-GAN over 10 intervals on 300 epochs. There is constant improvement on the calculated intervals.

6.1.1 FID over epochs figure

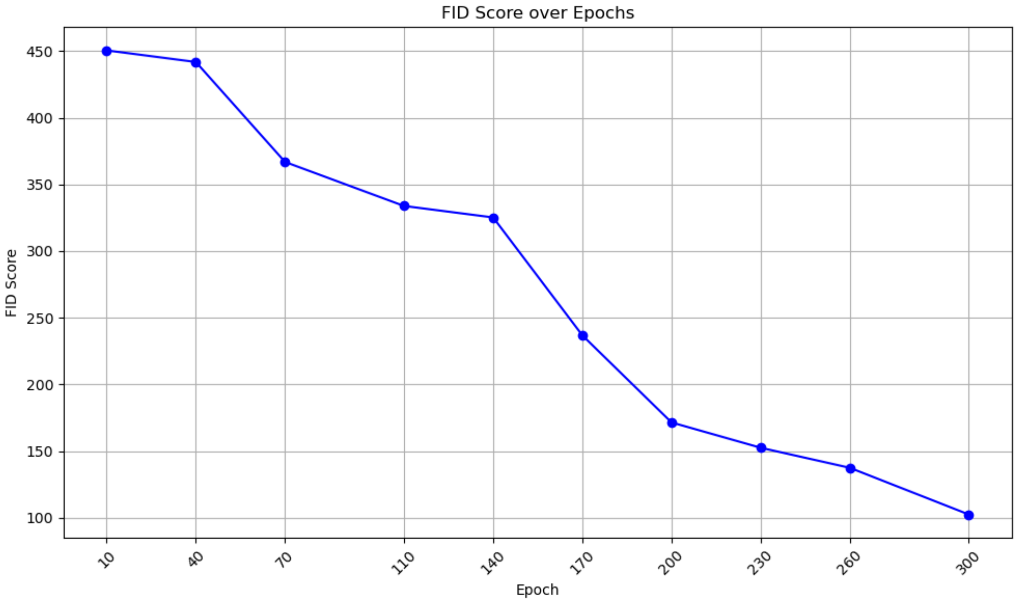
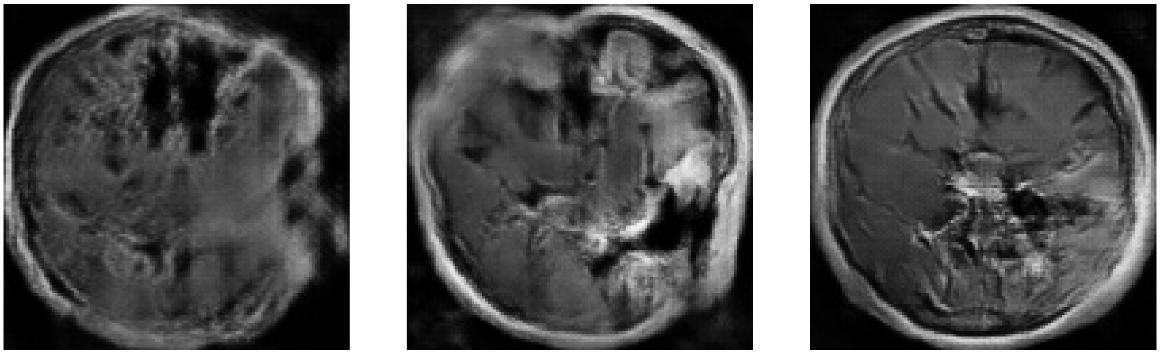


Figure 21: (1-5)-GAN, FID score over epoch.



(a) Epoch 100 FID score 340

(b) Epoch 200 FID score 170

(c) Epoch 300 FID score 105

Figure 22: $(1,5)$-GAN results for 128x128 no-tumour MRI scans at intervals over 300 epochs. Images corresponding to epochs 100, 200, and 300.

The results in Figure 21 are an illustration of the improvement of the $(1,5)$-GAN model over 300 epochs.The continual decrease in FID indicates the progressive generator models are producing more similar images to the underlying dataset. FID was only able to be calculated for 10 generator models due to the large computational requirements of a 16000×16000 matrix multiplication. The corresponding images in Figure 22 come from the associated models which were used to calculate FID in Figure 21. The images generated in Figures 22a, 22b,22c are representative of the average quality of MRI image which the generator model was capable of generating at that epoch.

6.1.2 Diversity in Generated Images

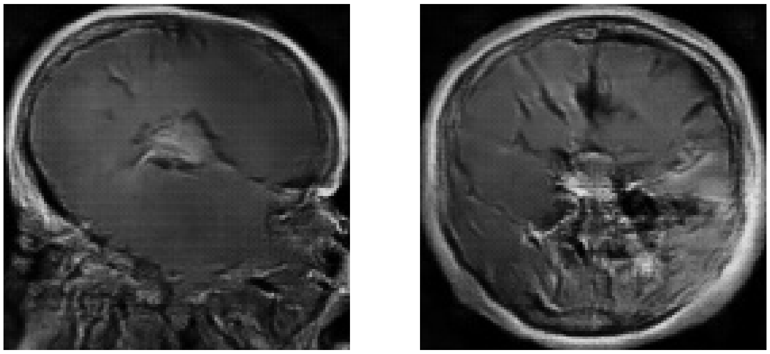


Figure 23: Generated tumour images from the $(1,5)$-GAN with the 300 epoch Generator.

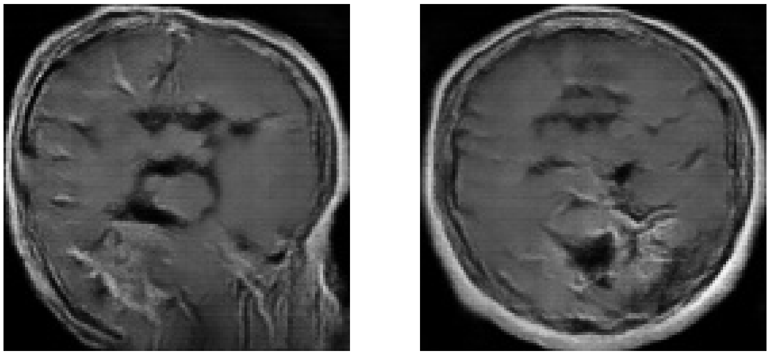


Figure 24: Generated no tumour images from the $(1,5)$-GAN with the 300 epoch Generator.

This section outlines the (α_D, α_G) = (1, 5) model capabilities. The GAN network is able to produce a diverse set of images. The results in Figure 23 are from the epoch 300 Generator trained on the tumour dataset. The Figure on the left closely resembles a side view MRI scan while the Figure on the right is a top view. Similar results were achieved in Figure 24 with less success is the selected side view scan. This could be attributed to the worse quality the no tumour model or just slight differences in single image selection. On the aggregate the FID for this no tumour model was 7 units greater than the tumour model, as seen from Table 4 and 5.

6.2 Performance of Classification Model

6.2.1 Metrics Used

Given the application, classification accuracy was the first metric to consider. This is defined as the percentage of predictions that were correct. For example, if the classification accuracy was 90%, this would mean the model predicted the correct label 9 out of 10 times. The number of false negatives is not considered in the classification accuracy metric. This brings us to the F1 and F2 scores. The F1 score is the harmonic mean of precision and recall, and is calculated as,

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{43}$$

This is a preferable metric to only considering accuracy because it takes the size of the classes into account through the trade-off between recall and precision. This is especially important for applications to medical datasets due to their imbalanced nature.

The F2 score is comparable to the commonly used F1 score but more heavily punishes false negatives, and is calculated as,

$$F2 = \frac{5 \times \text{Precision} \times \text{Recall}}{4 \times \text{Precision} + \text{Recall}}. \tag{44}$$

F2 score is an appropriate metric for this application because of the higher cost of misdiagnosing a positive patient. A false negative is more damaging as it could lead a patient to believe that they no longer need to seek care.

Other metrics were considered such as Area Under Precision Recal Curve (AUPRC), and Dice Similarity coefficient (DSC), however, they were deemed unnecessary and not appropriate for the analysis.

6.2.2 Identifying the Best Model

In Table 6, and Table 7, the different classification accuracies and the F2 scores are shown, respectively, with the best model for each dataset having bold numbers. For the rest of this section, **GC** dataset will refer to the dataset augmented with 10,000 images using the (α_D, α_G)-GAN and with classical data augmentation. Classical data augmentation involved scaling the training images by a random factor between 1 and 2 (normally distributed around 1.5, with variance 0.5), and convolving the image with a Gaussian blur kernel with size 3x3 and variance 1.

Classification Accuracy	Base Dataset	Only Classical Augmentation	Only GAN Augmentation	GC
$(1,0.5)$-GAN	98.58%	98.65%	94.08%	94.13%
$(1,1)$-GAN	98.58%	98.65%	82.7%	83.1%
$(1,5)$-GAN	98.58%	98.65%	**96.14%**	**96.18%**
$(1,10)$-GAN	98.58%	98.65%	94.74%	94.42%

Table 6: Classification Accuracies for different GAN models and Datasets

F2 Scores	Base Dataset	Only Classical Augmentation	Only GAN Augmentation	GC
$(1,0.5)$-GAN	0.971	0.975	0.939	0.940
$(1,1)$-GAN	0.971	0.975	0.841	0.844
$(1,5)$-GAN	0.971	0.975	**0.956**	**0.963**
$(1,10)$-GAN	0.971	0.975	0.947	0.949

Table 7: F2 Scores for different GAN models and Datasets

For all models, the best metrics were when trained on the GC dataset. The best model was the $(1,5)$-GAN by classification accuracy, F1 score (table omitted), and F2 score. Figure 27 shows the confusion matrix of the classification network trained on the base model, and Figure 28 shows the confusion matrix of the classification trained on the $(1,5)$-GC dataset. The confusion matrix for only classical or only GAN augmentation is omitted.

Table 6 shows that the accuracy for the original MRI dataset using the classification model was 98.58%, and the accuracy for the best GC dataset was 96.18%. This is a 2.4% decrease in accuracy. The accuracy and loss plots of both the base dataset and (α_1, α_5)-GC dataset are shown below in Figures 25, and 26.

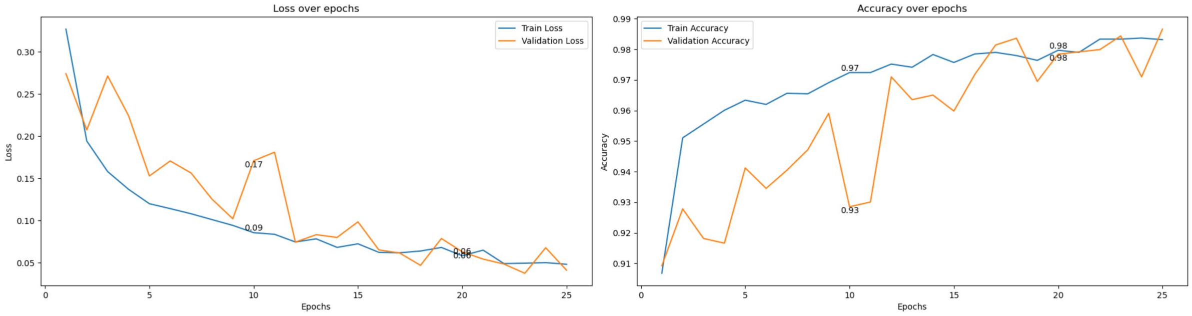


Figure 25: Accuracy and Loss for Base Dataset

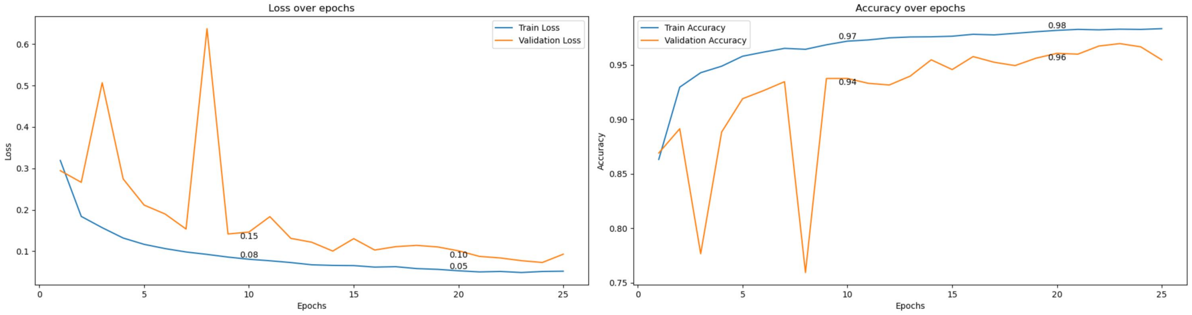


Figure 26: Accuracy and Loss for (α_1, α_5)-GC Dataset

Figures 27 and 28 display confusion matrices that depict the true positive, true negative, false positive, and false negative predictions for the respective datasets. Note that the off-diagonals are the entries of concern, with the bottom left entry, the number of false negatives being the entry that's most important to minimize.

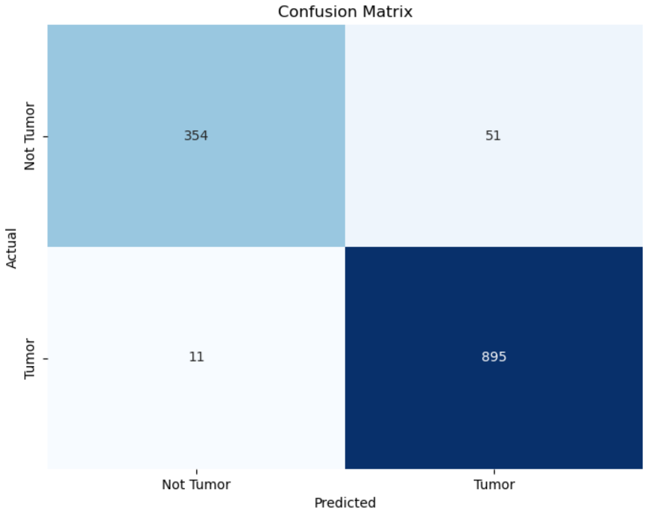


Figure 27: Confusion Matrix for Base Dataset

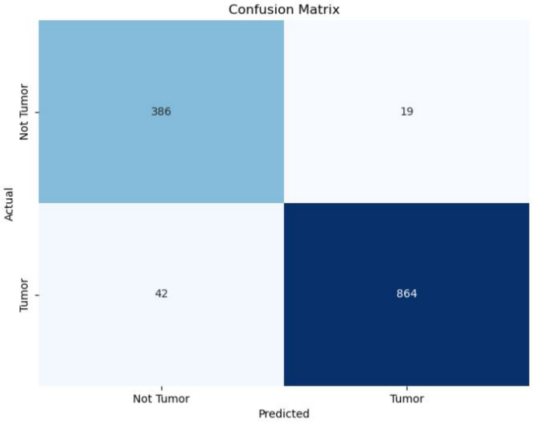


Figure 28: Confusion Matrix for GC Dataset

The number of false negatives for the base dataset was 11, and the number of false negatives for the GC dataset was 42. The number of false positives decreased with the GC dataset compared with the base dataset with the numbers being 19, and 51, respectively.

6.2.3 Statistical Significance

Following the work of [27], [29], a student's two-sample t-test was conducted to compare the accuracy and F2 results between the two groups and determine if there is a statistical significance. The two groups were Group 1: Base Dataset, compared with Group 2: Classical + GAN Augmented Dataset. Five trials of each group were compared to determine a statistically significant change between the groups, i.e. identical models were trained five times on either the base dataset or the augmented dataset. The null hypothesis was that training a classification network on the GC dataset yields the same accuracy and F2 results as training a classification network on the base dataset. Since we determined that the GC dataset on average does not increase classification accuracy or F2 score, the hypothesis being tested was that training a classification network on the GC dataset yields worse accuracy and F2 results than training a classification network on the base dataset. Table 8 displays the p-values for each metric as a result of this test. The p-value in this context indicates the likelihood that the observed differences between groups are attributable to random variation alone.

Metric	GAN Augmentation and Classical Augmentation vs. Base Dataset
Accuracy	0.0003
F2	0.0031

Table 8: p-values

Since both metrics yield a p-value less than 0.005, it was concluded that the null hypothesis may be rejected and thus have statistically significant differences among groups. A benchmark of 0.005 is a common number used in the literature [44] for such a task. Thus it is concluded that training a classification network on the GC dataset yields worse accuracy and F2 results than training a classification network on the base dataset. See Section 7.2 for a discussion on the results.

6.3 Limitations in Analysis

For a comprehensive analysis to be completed and appropriate conclusions to be drawn, it is necessary to identify limitations that may have impacted the results.

The main limitation encountered in these experiments was that there is no analytical method to determine the optimal α parameters for the GAN. This implies that there exists a possible solution which would yield better results. The approach taken to determine the choices for α was to refer to related experiments, such as those conducted by Veiner et al. [4]. Ideally, more trials could be conducted with different parameters to find a better solution, or to reinforce these conclusions. Another limitation in terms of analysis is due to the unstable nature of GANs that can cause them to have varying outputs, even for fixed model parameters and inputs. This means that exact reconstruction of the results may not be possible and the performance may fluctuate over small samples of outputs. With time to run more trials, more certain results could be acquired by averaging and discarding outliers.

Although the FID score is the ideal metric to analyze the performance of GANs, it does have some drawbacks. Firstly, the evaluation of the results may also be impacted by an overestimate of the FID score. Due to computing power constraints, the chosen dataset consisted of only 7023 images. However, the FID metric is most effective when considering more than 10,000 samples [43]. Additionally, FID poorly accounts for the advantage of diversity in the generator by categorizing it with quality [43]. Computing the FID score at every epoch also proved to be too computationally demanding which required for results to be interpolated based on the general trends.

7 Discussion of Results

7.1 Summary of GAN

The (1, 5)-GAN attained the lowest FID of all GANs trained on (128x128) image size. The FID score for the (1, 5)-GAN was calculated for 10 intervals on 300 epochs. Over these intervals there was a constant decrease in FID, indicated the underlying distribution of the generated image set was becoming closer to the underlying distribution of the real image set. This generator model at 300 epochs was capable of producing MRI images which looked similar to real ones to the human eye. The best generated images were then used to train the classification model and there was a relatively low impact on classification accuracy. This low impact on classification accuracy in combination with the context specificaly low FID score is a positive result for the GAN models.

7.2 Summary of Classification model

The classification network results show that there is a need for further experimentation in augmenting MRI datasets. A proof of concept has been demonstrated using the (α_D, α_G)-GAN. Related studies have seen success with similar paradigms [29][27][20][28]. These studies had more positive outcomes given their access to better computational resources and longer time available for careful model optimization. The attained results this project demonstrates the need for extensive computational resources, precise hyperparameter tuning, and time required for model iteration. Since there is no method to analytically determine the optimal set of hyper-parameters for both the GAN and the classification network, an exhaustive search was required, one that was not feasible with the time and computational resource constraints. There is an opportunity for future research to build upon these findings and utilize the potential of the (α_D, α_G)-GAN in augmenting MRI datasets. Additionally, the slight decrease in accuracy observed when using the augmented dataset can be thought of as the cost of maintaining privacy. The generated images are synthetic and thus do not violate any medical privacy standards and may be used more freely and distributed to many groups interested in performing the same task.

8 Impact on Triple Bottom Line

As the cornerstone of supervised machine learning, large data sets have helped solve problems such as facial recognition and medical image analysis. It has been shown that large models can face issues such as representation bias [45] and expensive data collection. Data augmentation through GANs can provide benefits by reducing representation bias, reducing energy spent collecting data, and reducing the cost of collecting and storing data.

8.1 Social Impact

Using GANs for MRI scan data augmentation has advantageous societal impacts. Medical imaging datasets, including MRI data, often suffer from imbalance and limited diversity. Currently, countries such as the United States, China, and Japan account for more than 50% of all cancer related research publications, often conducted with data collected within the respective counties. However, more than 60% of cancer fatalities occur in low- and middle-income countries, which have less access to cancer treatment services. Only one in five such countries have the necessary data to drive cancer policy and treatment, including treatment for brain tumours [48]. GANs can reduce this bias by generating synthetic scans, increasing inclusivity and diversity in datasets, and facilitating collaborative research across countries.

Furthermore, since GANs enable the creation of synthetic MRI scans for research purposes, it eliminates the need for patient consent procedures and preserving privacy. This saves time and resources for researchers while maintaining data integrity and confidentiality [49].

Moreover, GANs offer a solution to the ethical concerns surrounding the use of Gadolinium based contrast agents (GBCA) used in MRI examinations particularly in the realms of research [50]. Synthetic scans avoid exposing individuals to potential side effects of Gadolinium, such as nausea, headaches, and even poisoning.

However, using GANs for data synthesis and augmentation also presents drawbacks. There maybe potential misuse of the synthetic data if datasets were to be publicly accessible, as this project intends. A good data synthesizer may capture outliers in the original data and project them onto the synthetic data [51]. At present, there does not exist any clear legislation surrounding the use of synthetic data. This disables regulatory frameworks like General Data Protection Regulation (GDPR) and Health Insurance Portability and Accountability Act (HIPAA) from addressing all potential risks related to medical synthetic data. Additionally, the substantial computational resources required to train GAN models pose accessibility challenges for smaller organizations [52].

8.2 Environmental Impact

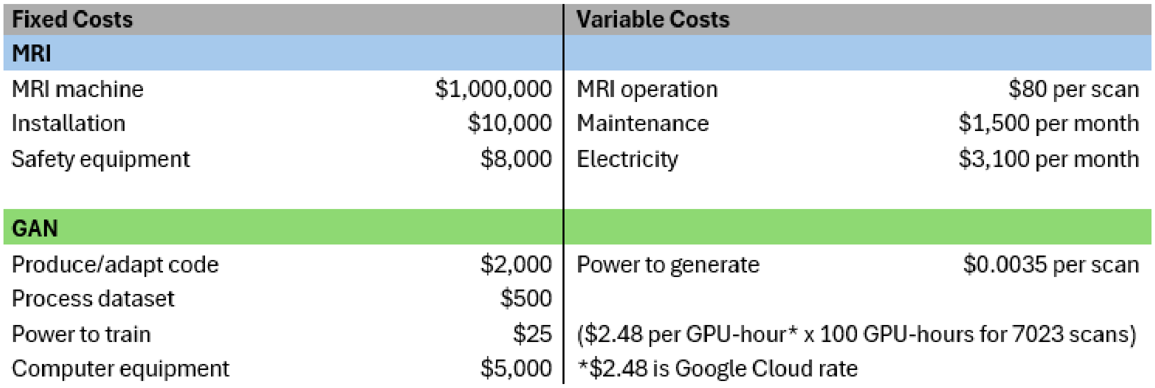
Using GANs for data augmentation of MRI scans has several environmental advantages and a few drawbacks. MRI machines need 25-80 kW of power to conduct an examination depending on the complexity of the examination. MRI machines also have built in cooling systems which requires additional power. Due to the nature of the mechanism within MRI machines, they need a constant source of power even when not in use [53]. Additionally, the use of Gadolinium based contrast agents (GBCA), which is injected into patients to produce clear images, leads to water contamination, and poses risks to several aquatic species [54]. Therefore, using GANs to generate MRI scans eliminates the consumption of substantial amounts of electricity that they run on, and also mitigates the environmental impacts.

However, there are drawbacks to training and running a GAN. The computational resources required to operate GANs are considerable and given that this project required running a few different iterations of the L_α-GAN, more computational resources and power was consumed [55]. Additionally, the process of creating, storing, and managing the augmented data demands significant amounts of energy [56]. Therefore, while GAN-based data augmentation presents several environmental benefits, its energy-intensive nature comes with challenges that need to be addressed.

8.3 Economic Considerations

Following the fact that data collection takes time and resources, it also comes at a financial cost. Companies that wish to implement a machine learning model must first collect or purchase data. Augmenting data sets through GANs means that less data needs to be gathered which saves money. Furthermore, the data set will not take up as much storage on a device until after the data set has been generated. This leads to a reduction in storage space which can potentially lower costs if the data set is very large.

For a new technology to be implemented, it must be economically feasible relative to the alternative and current solutions. In Table 9 below, the initial capital costs and operating costs of the GAN-based solution are compared to the costs associated with traditional MRI scanning.



Fixed Costs		Variable Costs	
MRI			
MRI machine	$1,000,000	MRI operation	$80 per scan
Installation	$10,000	Maintenance	$1,500 per month
Safety equipment	$8,000	Electricity	$3,100 per month
GAN			
Produce/adapt code	$2,000	Power to generate	$0.0035 per scan
Process dataset	$500		
Power to train	$25	($2.48 per GPU-hour* x 100 GPU-hours for 7023 scans)	
Computer equipment	$5,000	*$2.48 is Google Cloud rate	

Table 9: Cost breakdown between the generation of MRI scans using GAN versus using physical equipment [23] [57].

Evidently, the GAN based solution of generating MRI scan images is far superior, assuming the existence of datasets to be trained on.

Another consideration that favours the utilizing GANs is its potential in reducing the cost of health impacts. The 5 year survival rate of glioblastoma, which is the most prevalent form of brain tumour, is 4.7% and requires many intensive procedures that are costly [58]. Any increased rate of early detection due to automation of detection would result in a higher survival rate and less burden on the healthcare system. Additionally, the use of GANs to generate MRI scans can have a positive economic impact in the training of doctors. Since the estimated incidence rate for brain tumours is only 5 per the lifetime of 100,000 people, and due to patient privacy regulations, it can be difficult for medical schools to have a diverse set of positive tests for students be be familiarized with [58]. Therefore, the ability of GANs to balance datasets and generate new instances of positive scans is valuable.

For these reasons, from an economic perspective. GANs should be a complementary tool that every medical provider should consider to operate alongside their existing MRI machine.

9 Ethics and Equity

In addition to the societal considerations already discussed, various ethical and equity factors also must be considered. For instance:

Informed consent: Informed consent is critical within the context of datasets employed in training the GAN model. While it is important to obtain explicit consent from patients who have contributed their scans to the creation of the datasets being used, there is a notable gap in the understanding of whether the patients are aware about the potential utilization of their medical data for creating synthetic data. The dataset used to train the GAN model is called the ‘Brain Tumour MRI Dataset’ found in [42]. The contents of the dataset were collected by one of the authors, and there exists no explicit details on how they were collected. However, the datasets are published under the MIT License and can be used freely [59].

Bias and Fairness: GANs inherently learn from the data they are trained on, which could emphasize biases present in the training data. If the training dataset lacks diversity, synthetic scans may not accurately reflect the complete range of clinical scenarios [60]. The training dataset consists of four distinct classes of tumours: glioma, meningioma, pituitary, and no tumour. While the four classification contributes to mitigating bias and ensuring a more comprehensive representation of clinical diversity, no other details about the locality and diversity of the dataset are available.

Quality and Reliability: The accuracy and reliability of synthetic MRI scans generated by GANs may not be on par with real scans. If clinicians or researchers rely on these synthetic scans for diagnostic or research purposes, there could be risks of misinterpretation or incorrect conclusions, potentially compromising patient care or scientific validity.

10 Future Work

10.1 Implementing Conditional GAN

The conditional generative adversarial network (CGAN) was originally proposed as an extension of the GAN where both the generator and discriminator have access to additional information **y** [47]. In the case of MRI scans, the additional information will be the class labels. The minimax game from the GAN given the additional information is formulated as

$$\min_G \max_D V(G,D) = \min_G \max_D \left(E_{A\sim p_X}[\log(D(A \mid y))] + E_{B\sim p_Z}[\log(1 - D(G(B \mid y)))]\right). \tag{45}$$

The result of implementing the CGAN is that forward passes of the generator network will produce images corresponding to the given label. In cases where datasets have several labels, one would have to build and train a GAN for each class label. The benefit of the CGAN is that only one is required for any number of classes.

Considering the case of brain tumours, a classification model can be built on different granularities of labels. The most simple of them being ”tumour or no tumour”. In addition, tumours can be located in different regions of the brain and can have many features that increase the number of labels required to determine the type of tumour. The CGAN is therefore recommended for future work where there are more than 2 labels.

10.2 Systematically Analyzing Generated Images

The analysis process did not include a method of removing images that were not of high quality. Given the nature of neural networks, particularly a generator, some percentage of generated images even for a high-quality model are prone to being low quality. Some methods of analysis are applicable such as FID comparisons or simply mean and variance comparisons with the original dataset. This is something that would greatly increase the quality of the final augmented dataset. Note that manually removing images that look of low quality is both unreliable and unfeasible given the number of images used for data augmentation.

10.3 Image Resolution of (256×256)

To generate images of higher quality and greater detail it is recommended to train (α_D, α_G)-GAN on an image size of (256x256). This image size would be closer to the original dimensions of the MRI dataset at 500x500. There is a possibility that this upscaling would improve the results of the classification model. A limitation is that this is only possible to train with access to GPUs which have memory greater than 24 GB.

10.4 Systematic Cross Validation

With the L_α-GAN framework capable of recovering many types of GANs each with their own set of unique hyperparameters, a more systematic approach to cross validation would have perhaps have uncovered a more optimal set of hyperpamaeters for the (α_D, α_G)-GAN, or even suggested a different GAN type as the most optimal GAN for our application. This could be accomplished with access to greater computing resources rendering the use of python methods, such as the .GridSearchCV method from the SciKit-Learn library, feasible. This method or any similar methods would evaluate each permutation of test hyperparameter values according to a specified metric and return the optimal combination.

10.5 Other Applications

The need for extensive pre-processing of the MRI images prior to data augmentation using the GAN has increased the versatility of this design process in that it is able to handle a wide variety of images and image sizes. Therefore, the results could be applied to other types of medical images that are different dimensions, as well as to other applications where augmentation of imbalanced image datasets would be valuable.

One application that would be promising to explore would be using this design process on dental imagery. Dental imaging and scanning shares several characteristics, including high cost of generation and imbalanced nature of the datasets, which presented a need for GANs to be applied to MRI brain scans. Successful augmentation of dental image datasets has the potential to enable automated identification of cavities, gum disease, and the need for tooth re-alignment. Research has shown that GANs have the ability to generate images that are indistinguishable to dentists [61]. Consequently, synthetically generated dental images could be used in the training of dentists by giving them a more extensive database to learn from without needing to infringe on patient privacy.

11 Conclusion

The aim of the project was to explore the potential of GANs for the augmentation of a brain MRI scan dataset to help mitigate challenges associated with data scarcity and class imbalance. The L_α-GAN model was selected for its flexibility and ability to recover various GAN variants through its tunable hyperparameters. The best variant recovered was found to be the (α_D, α_G)-GAN.

An iterative approach was employed for the design implementation, evaluating all recoverable variants from the L_α-GAN model, increasing the resolution of the generated images and fine-tuning the GAN hyperparameters. The (α_D, α_G)-GAN variant proved to be the most promising, with the best results obtained for $\alpha_D = 1$ and $\alpha_G = 5$. The generator model architecture was adapted to accommodate the larger MRI scan resolutions, and the FID score was used to assess the quality of the generated images.

Performance evaluation of the developed GAN models focused on two key aspects: the quality of the generated MRI scans and the impact of data augmentation on a classification model. The $(1,5)$-GAN achieved the lowest FID score among the tested variants, indicating that the generated images closely resembled the real MRI scans. The classification model, MonkeyNet, was trained on both the original dataset and the augmented dataset, which included a combination of GAN-generated and classically augmented images. While the augmented dataset led to a slight decrease in classification accuracy (96.18% compared to 98.58% for the base dataset), it is important to note that this can be considered as the cost of maintaining patient privacy through the use of synthetic data.

The impact on the triple bottom line was also considered, discussing the social, environmental, and economic implications of using GANs for medical data augmentation. The use of synthetic data can help reduce bias, protect patient privacy, and decrease the environmental impact associated with acquiring real MRI scans. However, challenges such as potential misuse of synthetic data and the energy-intensive nature of training GANs need to be addressed.

Future work could explore the implementation of conditional GANs to handle multiple class labels, increasing the image resolution to (256×256) for improved quality, and applying the design process to other medical imaging domains, such as dental imagery.

In conclusion, this project reinforces the potential of GANs for augmenting brain MRI scan datasets, offering a promising solution for data scarcity and class imbalance issues. However, further research and optimization are necessary to fully leverage the capabilities of the (α_D, α_G)-GAN in medical imaging applications.

References

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, ”Generative adversarial nets,” *Communications of the ACM*, vol. 57, no. 10, pp. 122-131, 2014.

[2] X. Mao, Q. Li, H. Xie, R. Lau, W. Zhen, and S. Smolley, ”Least Squares Generative Adversarial Networks.” *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2813-2821, 2017. [Online]. Available: `https://doi.org/10.1109/ICCV.2017.304`

[3] H. Bhatia, ”Generalized Loss Functions for Generative Adversarial Networks,” Master’s thesis, Department of Mathematics and Statistics, Queen’s University, Kingston, Ontario, Canada, October 2020.

[4] J. Veiner, ”Generative Adversarial Networks Based on a General Parameterized Family of Generator Loss Functions,” Master’s thesis, Department of Mathematics and Statistics, Queen’s University, Kingston, Ontario, Canada, September 2023.

[5] H. Algahtani, M. Kavakli-Thorne, and G. Kumar. ”Applications of generative adversarial networks (GANs): an updated review.” *Archives of Computational Methods in Engineering*, vol. 28, no. 2, pp. 525-552, 2021. [Online] Available: `https://doi.org/10.1007/s11831-019-09388-y`

[6] Google. Overview of GAN structures, 2019. `https://developers.google.com/machine-learning/gan/gan_structure`

[7] G. R. Kurri, T. Sypherd, and L. Sankar, ”Realizing GANs via a tunable loss function,” *2021 IEEE information theory workshop (ITW)*, pp. 1-6, 2021. [Online]. Available: `https://api.semanticscholar.org/CorpusID:235376829`.

[8] L. Shukla, ”Designing your neural networks,” *Medium*, Towards Data Science, Sep. 2019. [Online]. Available: `https://towardsdatascience.com/designing-your-neural-networks-a5e4617027ed`

[9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016. [Online]. Available: `http://www.deeplearningbook.org`

[10] ”The Role of Bias in Neural Networks,”. [Online]. Available: `https://www.pico.net/kb/the-role-of-bias-in-neural-networks/`

[11] ”Supervised Learning,” *IBM*. [Online]. Available: `https://www.ibm.com/topics/supervised-learning`

[12] ”What are Convolutional Neural Networks?” *IBM*. [Online] Available: `https://www.ibm.com/topics/convolutional-neural-networks#:~:text=the%20intended%20object.-,Convolutional%20layer,matrix%20of%20pixels%20in%203D.`

[13] A. K. Nandi, K. K. Randhawa, H. S. Chua, M. Seera, and C. P. Lim, “Credit card fraud detection using a hierarchical behavior-knowledge space model,”*PloS ONE*, vol. 17, no. 1, pp. e0360679, 2022. [Online]. Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8775357/`.

[14] U. Fiore, A. De Santis, F. Perla, P. Zanetti, and F. Palmieri, “Using generative adversarial networks for improving classification effectiveness in credit card fraud detection,” *Information Sciences*, vol. 479, pp. 448-455, 2019. [Online]. Available: `https://www.sciencedirect.com/science/article/abs/pii/S0020025517311519`

[15] X. Tong et al. ”Neyman-Pearson classification algorithms and NP receiver operating characteristics,” *Sci. Adv.,* vol. 4, 2018. [Online]. Available: `https://doi.org/10.1126/sciadv.aao1659`

[16] N. Kodali, J. Abernethy, J. Hays, Z. Kira, ”On Convergence and Stability of GANs,” 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1705.07215

[17] A. ArjomandBigdeli, M. Amirmazlaghani and M. Khalooei, ”Defense against adversarial attacks using DRAGAN,” *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, pp. 1-5, 2020. [Online] Available: https://doi.org/10.1109/ICSPIS51611.2020.9349536

[18] A. Lamberti. ”Synthetic financial data: solution to the problem of limited financial data”. *Syntheticus*. 2023. [Online]. Available: https://syntheticus.ai/blog/synthetic-financial-data.

[19] S. Takahashi, Y. Chen, K. Tanaka-Ishii. ”Modeling financial time-series with generative adversarial networks”. *Physica A: Statistical Mechanics and its Applications*, vol. 527, pp. 1-14, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0378437119307277

[20] V. Sandfort, K. Yan, P. J. Pickhardt, et al., ”Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks”. *Sci Rep*, vol. 9, 2019. [Online]. Available: https://doi.org/10.1038/s41598-019-52737-x

[21] D. S. Kermany, K. Zhang, and M. H. Goldbaum, ”Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification,” 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:126183849

[22] Editorial Staff, ”Top 10 challenges of Big Data Analytics in Healthcare,” *HealthITAnalytics*, 2024. [Online]. Available: https://healthitanalytics.com/news/top-10-challenges-of-big-data-analytics-in-healthcare

[23] ”MRI costs,” *Imaging Technology News*, 2021. [Online]. Available: https://www.itnonline.com/content/mri-costs

[24] CADTH, ”Private Imaging Facilities in Canada: MRI and CT,” *Canadian Medical Imaging Inventory*, 2022. [Online]. Available: https://www.cadth.ca/sites/default/files/attachments/2022-06/CMII-MRI-CT-Final_3.pdf

[25] S. Deepak, P.M. Ameer. ”Brain tumour categorization from imbalanced MRI dataset using weighted loss and deep feature fusion,” *Neurocomputing*, vol. 520, pp. 94-102, 2023. [Online]. Available: https://doi.org/10.1016/j.neucom.2022.11.039

[26] C. Shorten, T.M. Khoshgoftaar, ”A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, p.60, 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0

[27] S.K. Hung, J.Q. Gan, ”Augmentation of Small Training Data Using GANs for Enhancing the Performance of Image Classification,” in *2020 25th International Conference on Pattern Recognition (ICPR),* pp.3350-3356, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:233877415

[28] C. Bowles et al., ”GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks,” 2018. [Online]. Available: https://arxiv.org/abs/1810.10863

[29] S.Motamed, P. Rogalla, F. Khalvati, ”Data augmentation using Generative Adversarial Networks (GANs) for GAN-based detection of Pnemumonia and COVID-19 in chest X-ray Images,” *Informatics in Medicine Unlocked,* vol. 27, 2021, Art. no. 100779, ISSN: 2352-9148. [Online]. Available: https://doi.org/10.1016/j.imu.2021.100779

[30] ”Guidance Document: Software as a Medical Device (SaMD): Definition and Classification,” *Health Canada*, Government of Canada, 2019. [Online]. Available: `https://www.canada.ca/en/health-canada/services/drugs-health-products/medical-devices/application-information/guidance-documents/software-medical-device-guidance-document.html`

[31] M. Gabay, ”21st Century Cures Act,” *Hosp Pharm*, vol. 52, no. 4, pp. 264–265, 2017. [Online]. Available: `https://doi.org/10.1310/hpj5204-264`

[32] U.S. Department of Health and Human Services, Food and Drug Administration, ”Clinical Trial Imaging Endpoint Process Standards Guidance for Industry,” 2018. [Online]. Available: `https://www.fda.gov/drugs/guidance-compliance-regulatory-information/guidances-drugs`

[33] Public Health Professionals Gateway, ”Health Insurance Portability and Accountability Act of 1996 (HIPAA),” 1996. [Online]. Available: `https://www.cdc.gov/phlp/publications/topic/hipaa.html`

[34] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, ”Unrolled Generative Adversarial Networks,” *International Conference on Learning Representations*, 2017. [Online]. Available: `https://openreview.net/forum?id=BydrOIcle`

[35] A. Krizhevsky, I. Sutskever, G. E. Hinton, ”ImageNet Classification with Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, 2012. [Online]. Available: `https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, ”Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014. [Online]. Available: `http://jmlr.org/papers/v15/srivastava14a.html`

[37] O. Russakovsky et al., ”ImageNet Large Scale Visual Recognition Challenge,” 2015. [Online]. Available: `https://doi.org/10.48550/arXiv.1409.0575`

[38] S. Ioffe, C. Szegedy, ”Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” 2015. [Online]. Available: `https://doi.org/10.48550/arXiv.1502.03167`

[39] B. Xu, N. Wang, T. Chen, M. Li, ”Empirical Evaluation of Rectified Activations in Convolution Network,” University of Alberta, Hong Kong University of Science and Technology, University of Washington, Carnegie Mellon University, November 27, 2015.

[40] J. Veiner, F. Alajaji, and B. Gharesifard, ”A unifying generator loss function for generative adversarial networks,” 2024. [Online]. Available: `https://arxiv.org/pdf/2308.07233.pdf`

[41] Y. LeCun, C. Cortes, C.J.C. Burges, ”The MNIST database of handwritten digits”, 1998. [Dataset]. Available: `http://yann.lecun.com/exdb/mnist/`

[42] M. Nickparvar. ”Brain tumour MRI Dataset”, 2022. [Dataset]. Available: 10.34740/kaggle/dsv/2645886

[43] A. Borji, ”Pros and cons of GAN evaluation measures: New developments,” *Computer Vision and Image Understanding,* vol. 215, p. 103329, 2022. [Online]. Available: `https://doi.org/10.1016/j.cviu.2021.103329`

[44] Herman Aguinis, Matt Vassar, and Cole Wayant. *On reporting and interpreting statistical significance and p values in medical research*. BMJ Evid Based Med, 26(2): 39–42, April 2021. Published online November 15, 2019. doi: 10.1136/bmjebm-2019-111264.

[45] H. F. Menezes, A. S. C. Ferreira, E. T. Pereira, and H. M. Gomes, ”Bias and Fairness in Face Detection,” *34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Gramado, Rio Grande do Sul, Brazil, 2021, pp. 247-254. Available: `https://ieeexplore.ieee.org/document/9643102`

[46] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, ”Deep Learning with Differential Privacy,” *2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308-318, 2016 doi: `http://dx.doi.org/10.1145/2976749.2978318`

[47] M. Mirza and S. Osindero, ”Conditional generative adversarial nets,” 2014. [Online]. Available: `https://doi.org/10.48550/arXiv.1411.1784`

[48] B.P. Cabral, F.B. Mota, ”The recent landscape of cancer research worldwide: a bibliometric and network analysis,” *Oncotarget*, vol. 9, no. 55, pp. 30474–30484, Jul. 17, 2018. [Online]. Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6078146/`

[49] P. Singleton, M. Wadsworth, ”Consent for the use of personal medical data in research,” *BMJ*, vol. 333, no. 7561, pp. 255–258, Jul. 29, 2006.[Online]. Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1523480/`

[50] N. Iyad, M. S Ahmad, S.G. Alkhatib, M. Hjouj, ”Gadolinium contrast agents- challenges and opportunities of a multidisciplinary approach: Literature review,” *Eur J Radiol Open*, vol. 11, p. 100503, Jul. 4, 2023. [Online] Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10344828/`

[51] M. Abadi, A. Chu, I. Goodfellow, H.B. McMahan, I. Mironov, K. Talwar, L. Zhang, ”Deep Learning with Differential Privacy,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (CCS ’16), New York, NY, USA, Association for Computing Machinery, 2016, pp. 308–318. [Online]. Available: `https://doi.org/10.1145/2976749.2978318`

[52] M. Giuffrè, D.L. Shung, ”Harnessing the power of synthetic data in healthcare: innovation, application, and privacy,” *NPJ Digit Med*, vol. 6, no. 1, p. 186, 2023. [Online]. Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10562365/`

[53] M. Chodorowski, J. Ognard, À. Rovira, J.C. Gentric, D. Bourhis, D. Ben Salem, ”Energy consumption in MRI: Determinants and management options,” *J Neuroradiol*, vol. 51, no. 2, pp. 182–189, 2024. [Online]. Available: `https://pubmed.ncbi.nlm.nih.gov/38065429/`

[54] R. Brünjes, T. Hofmann, ”Anthropogenic gadolinium in freshwater and drinking water systems,” *Water Res*, vol. 182, p. 115966, 2020. [Online]. Available: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7256513/`

[55] D. Saxena, J. Cao, ”Generative Adversarial Networks (GANs): Challenges, Solutions, and Future Directions,” *ACM Comput. Surv.*, vol. 54, no. 3, 2021, [Online]. Available: `https://doi.org/10.1145/3446374`.

[56] C.L. Chen, C. Zhang, ”Data-intensive applications, challenges, techniques and technologies: A survey on Big Data,” *Information Sciences*, vol. 275, pp. 314-347, 2014. [Online]. Available: `https://doi.org/10.1016/j.ins.2014.01.015`.

[57] A. Sahu, H. Vikas, and N. Sharma, ”Lift cycle costing of MRI machine at tertiary care teaching hosptial,” *The Indian Journal of Radiology & Imaging,* vol. 30, no. 2, pp. 190-194, 2020. [Online]. Available: `https://doi.org/10.4103/ijri.IJRI_54_19`

[58] A.I. Neugut, P. Sackstein, G.C., Hillyer, J.S.Jacobson, J.Bruce, A.B. Lassman, and P.A. Stieg, ”Magnetic Resonance Imaging-Based Screening for Asymptomatic Brain Tumors: A Review,” *The Oncologist,* vol. 24, no. 3, pp.375-384, 2019. [Online]. Available: `https://doi.org/10.1634/theoncologist.2018-0177`

[59] Open Source Initiative, ”The MIT License,” [Online]. Available: `https://opensource.org/license/mit`

[60] R. Challen, J. Denny, M. Pitt, L. Gompels, T. Edwards, K. Tsaneva-Atanasova, ”Artificial intelligence, bias and clinical safety,” *BMJ Quality & Safety*, vol. 28, no. 3, pp. 231–237, 2019. [Online]. Available: `https://qualitysafety.bmj.com/content/28/3/231`.

[61] K. Kokomoto, R. Okawa, K. Nakano, et al., ”Intraoral image generation by progressive growing of generative adversarial network and evaluation of generated image quality of dentists,” *Sci. Rep.,* vol. 11, no. 1, p.18517, 2021. [Online]. Available: `https://doi.org/10.1038/s41598-021-98043-3`

12 Appendix

12.1 Appendix I: Discriminator Architecture

Discriminator Summary

Layer (type)	Output Shape	Param #
Conv2D	(None, 128, 128, 64)	640
LeakyReLU	(None, 128, 128, 64)	0
Conv2D	(None, 64, 64, 128)	73,856
LeakyReLU	(None, 64, 64, 128)	0
Conv2D	(None, 32, 32, 128)	147,584
LeakyReLU	(None, 32, 32, 128)	0
Conv2D	(None, 16, 16, 256)	295,168
LeakyReLU	(None, 16, 16, 256)	0
Flatten	(None, 65536)	0
Dropout	(None, 65536)	0
Dense	(None, 1)	65,537

Total params: 582,785 (2.22 MB)
Trainable params: 582,785 (2.22 MB)
Non-trainable params: 0 (0.00 B)

12.2 Appendix II: Generator Architecture

Generator Summary

Layer (type)	Output Shape	Param #
Dense	(None, 16384)	268,435,456
Batch Normalization	(None, 16384)	65,536
LeakyReLU	(None, 16384)	0
Reshape	(None, 8, 8, 256)	0
Conv2D Transpose	(None, 8, 8, 128)	819,200
Batch Normalization	(None, 8, 8, 128)	512
LeakyReLU	(None, 8, 8, 128)	0
Conv2D Transpose	(None, 16, 16, 64)	204,800
Batch Normalization	(None, 16, 16, 64)	256
LeakyReLU	(None, 16, 16, 64)	0
Conv2D Transpose	(None, 32, 32, 64)	102,400
Conv2D Transpose	(None, 64, 64, 64)	102,400
Batch Normalization	(None, 64, 64, 64)	256
LeakyReLU	(None, 64, 64, 64)	0
Conv2D Transpose	(None, 128, 128, 1)	1,600

Total params: 269,732,416 (1.00 GB)
Trainable params: 269,699,136 (1.00 GB)
Non-trainable params: 33,280 (130.00 KB)

12.3 Appendix III: Classification Model Architecture

Table 10: Classification Model Architecture

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 32)	896
batch_normalization (BatchNormalization)	(None, 64, 64, 32)	128
max_pooling2d (MaxPooling2D)	(None, 32, 32, 32)	0
conv2d_1 (Conv2D)	(None, 32, 32, 64)	18,496
batch_normalization_1 (BatchNormalization)	(None, 32, 32, 64)	256
max_pooling2d_1 (MaxPooling2D)	(None, 16, 16, 64)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64)	36,928
max_pooling2d_2 (MaxPooling2D)	(None, 8, 8, 64)	0
global_average_pooling2d (GlobalAveragePooling2D)	(None, 64)	0
dropout (Dropout)	(None, 64)	0
dense (Dense)	(None, 64)	4,160
batch_normalization_2 (BatchNormalization)	(None, 64)	256
dropout_1 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 1)	65

12.4 Appendix IV: Lowest FID $(\alpha_\mathbf{d}, \alpha_\mathbf{g})$-GAN

Table 11: Parameter Summary

Parameter	Value
gan_type	alpha
alpha	3.0
seed	42
c_type	discrete
n_epochs	300
dataset	mri
loss_type	alpha
lambda_d	1.0
lambda_c	0.1
num_images	120
gp	True
gen_lr	0.0002
dis_lr	0.0002
q_lr	0.0002
gp_coef	5.0
alpha_d	1.0
alpha_g	5.0
k	2.0
shifted	False
l1	False