Project Report
Malcolm Yeh
101109829
COMP 4107

# Style transfer using CycleGAN to synthesize CT scans from MRI images

# 1 Introduction

## 1.1 Background

Computed Tomography (CT) scan is a medical imaging technique that uses X-rays in order to create cross-sectional images of the body. CT scans introduce a risk as it involves subjecting the patient to exposure of ionizing radiation, increasing the risk of cancer over time. Magnetic resonance imaging (MRI) utilizes a magnetic field and radio waves to produce detailed images inside of the body. If CT scans can be synthesized from a less invasive MRI, more information could be used for diagnostic medicine while increasing the overall safety. On the other hand, MRI can be significantly more expensive than CT scans depending on several factors and if more information can be extracted from a CT scan through style transfer, it could cut costs to the diagnosis.

## 1.2 Prior work

Generative Adversarial Network (GAN) is a type of artificial neural network architecture specifically designed for generating new data based on training data first introduced by Ian Goodfellow and colleagues in 2014.[1] A GAN consists of two neural networks trained simultaneously in adversarial training where a generator takes random noise as input and generates fake data while the discriminator distinguishes between real and generated data. The two networks compete against each other where the generator attempts to generate data that can fool the discriminator while the discriminator tries to identify whether the data is fake or real. The gradients from the discriminator are used to update the generator's parameters through backpropagation in order to produce more convincing outputs and thus harder for the discriminator to distinguish. In an ideal scenario, this two-player zero-sum game leads to the discriminator always outputting 0.5 as it cannot distinguish between real and fake data while the generator produces data that is appropriate based on some evaluation metric.

GANS can be used for image-to-image translation such as the conditional generative adversarial network Pix2Pix. Pix2Pix consists of two networks, a generator and discriminator. [2] The generator takes an input image and conditioning image and generates an output in the desired style while the discriminator takes pairs of images, one generated and one real and tries to distinguish them. The key

innovation of this architecture is the use of conditional GAN where the conditioning image fed to the generator provides additional information for the output, allowing it to better learn the desired mapping.

## 1.3 Project Significance and Objective

The objective of this project is to learn more about the medical imaging preprocessing, the CycleGAN [3] architecture and explore techniques to improve the result in the context of medical imaging through application. It is not expected that the results of this project would see any medical or scientific relevance due to the limitation in proposed datasets and validation methods.

# 2 Methodology

## 2.1 Dataset

The scans used in this project are unpaired - the CT scans were obtained from the CQ500 Dataset while the MRI scans were obtained from the IXI Dataset. Type 2 weighted (T2) MRI scans were chosen as they visualize fluid-filled structures which are present in CT scans. Each scan file is made up of slices, which refers to a thin cross-sectional image produced by the scanner and shows a specific level or depth within the head. [4] Due to limitations in hardware and training time, this project focuses only on a single slice. The goal was to acquire slices that captured the lateral ventricle, which are the two irregularly shaped cavities on both sides of the midline of the brain. Since they contain a lot of cerebrospinal fluid, they appear as intense structures on both CT and T2 weighted MRI scans. While the MRI dataset included scans with the same slice calibrations, the CT dataset contained files with varying numbers of slices. Therefore, slices 60-80 in the MRI scans were included and CT scans with a high percentage of black pixels were discarded as they would represent slices that are too high or low on the head.

For CT scans, a linear transformation was applied to each slice, mapping the pixel values to Hounsfield Units (HU). HU are units of measurement used to express the radiodensity of a particular tissue and are used to better differentiate between different materials in the CT scan. This was done by applying RescaleIntercept and RescaleSlope from the PyDicom library. Next, for both CT and MRI images, the noise was removed by performing image segmentation to identify the largest connected component corresponding to the brain region and applying binary operations to the segmentation mask to remove and fill small holes in the bones. This removed most external artifacts, including the large arc present in many CT scans. Using OpenCV findCounter() and fitEllipse(), a fitted ellipse is drawn around the brain slice and a rotation matrix was used to correct the tilt such that the image is upright. Lastly, the images were cropped and resized such that both datasets would be similar in shape.

Unfortunately, this method of data collection and preprocessing is crude in this context as there are large variations in the scans and several obviously incorrect or failed transformed images had to be manually removed from the dataset.

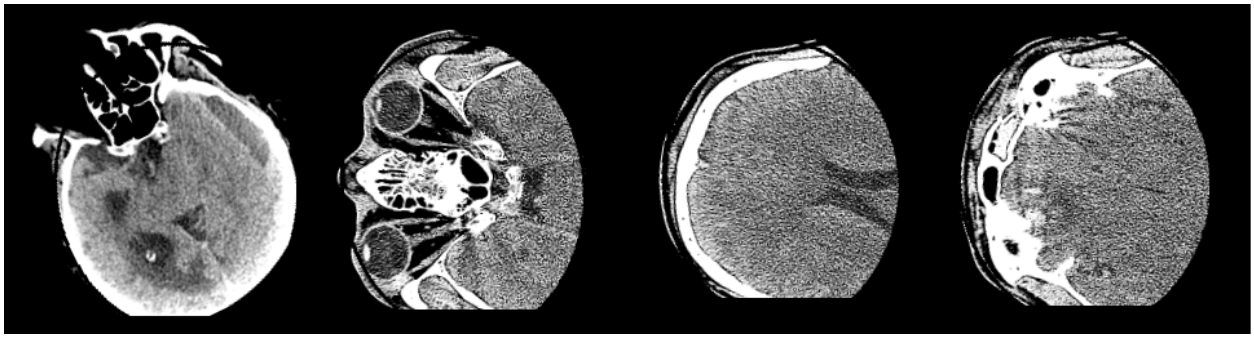

Figure 1. Examples of failed pre-processing and slice selection.

In the first and second images of Figure 1, the percentages of black pixels were within the threshold but it returned slices much lower on the human skull compared to the rest. In each case, the ellipse was incorrectly fitted and thus the rotation angle was not valid. The subsequent transformations rendered the resulting completely distorted.

Finally, the pixel values were limited and the images reshaped to 256 by 256 pixels.

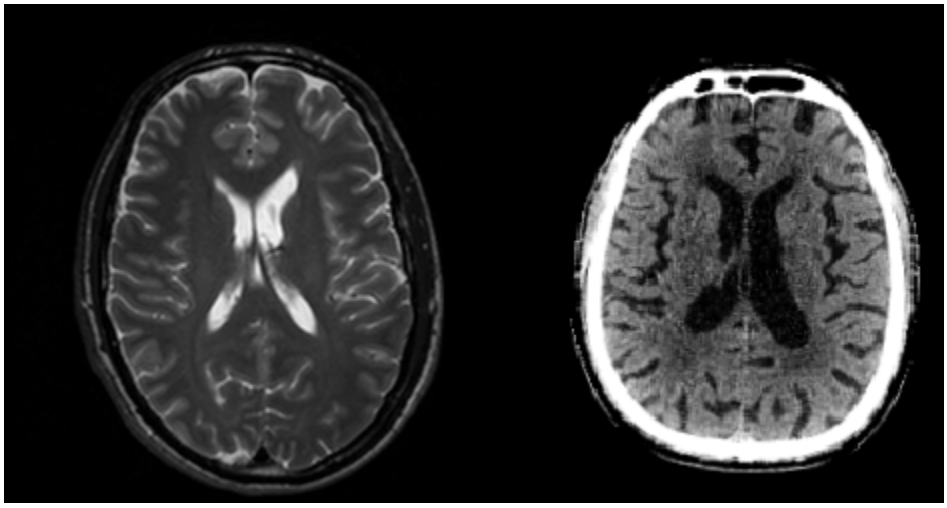

Figure 2. MRI Scan and CT Scan after preprocessing

## 2.2 Model

CycleGAN is a type of GAN that has two sets of generators and discriminators where one generator trains to translate images from one domain to the other and vice versa while the discriminators differentiate between real and fake images [3].

This project follows a similar network architecture to that presented in the CycleGAN paper, in which the generative networks are adopted from Johnson et al. Namely, the network has 3 convolutions for both the encoder and decoder, but only 4 residual blocks for the transformer. Residual blocks allow the network to learn residual functions through adding shortcut connections that bypass layers of the network and try to mitigate the problem of vanishing gradients. The discriminator consists of PatchGANs, which

combine input convolutional layers, ReLU activation and strided convolutions instead of max pooling layers to downsample the feature maps. PathGAN discriminator evaluates image patches rather than the whole image allowing it to capture fine-grained entails and textures of the input.

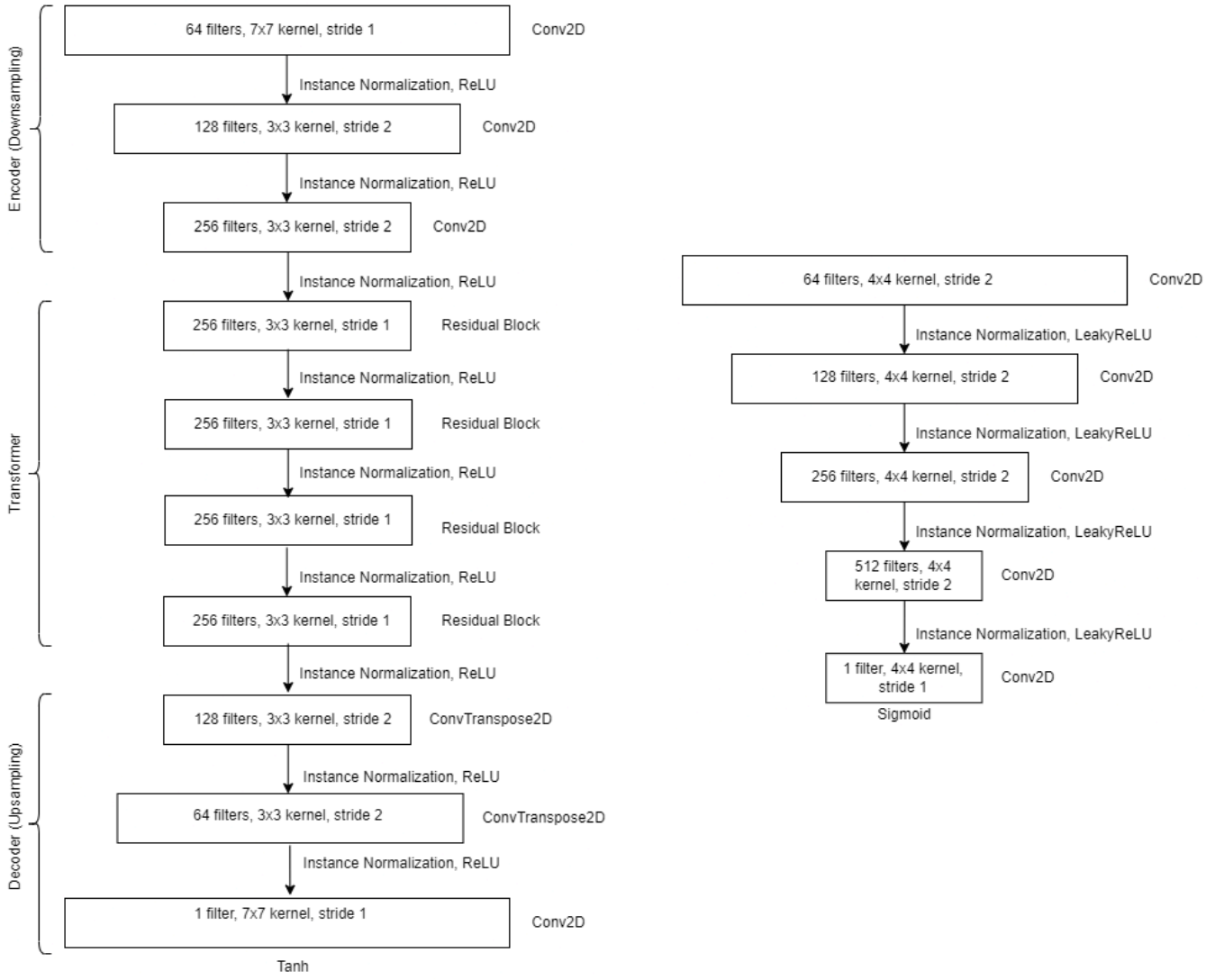


Figure 3. Generator architecture (right) and PatchGAN discriminator architecture (left)

## 2.3 Losses

Cycle consistency is the idea that the generated image from generator A used as an input to generator B should match the original input image to the first generator and vice versa. The CycleGAN architecture enforces cycle consistency using the loss between the original image and the reconverted image as part of the total generator loss and by minimizing this loss, ensures that generated images from each generator are high quality and consistent with the original images. [4]

Identity mapping is the mapping of an image from one domain to itself . Identity loss is calculated by measuring the loss between an image in domain A and the output of generator A that is fed the original image. [4] By minimizing this loss, it ensures that the mapping between domains is consistent with the respective identity mappings, which helps preserve the tint in the images.

First introduced in 2004, Structural Similarity Index measure (SSIM) is a method for measuring the similarity between two images. [6] While MSE quantifies absolute error through the difference in the values of each of the corresponding pixels, SSIM works by comparing the structural information of the images, which takes into account luminance, contrast and structure using the pixel sample means, which are important perceptual phenomena. Luminance is the brightness or intensity of the image, measured by averaging over all the pixel values. Contrast is the difference in luminance across different parts of the image, measured using the standard deviation of the pixels. Structural information is composed of high-frequency components like edges and textures. As a result, SSIM provides a more comprehensive measure of image similarity than metrics such as MSE or peak signal-to-noise ratio.

$$L_{SSIM}(x,\hat{x}) = contrast(x,\hat{x}) \ * \ luminance(x,\hat{x}) \ * \ structure(x,\hat{x})$$

Each component is weighted but the variables are not shown for simplicity. The components are defined as

$$contrast(x,\hat{x}) = \frac{2\sigma_x\sigma_{\hat{x}} + C_2}{\sigma_x^{\,2} + \sigma_{\hat{x}}^{\,2} + C_2}$$

$$luminance(x,\hat{x}) = \frac{2\mu_x\mu_{\hat{x}} + C_1}{\mu_x^{\,2} + \mu_{\hat{x}}^{\,2} + C_1}$$

$$structure(x,\hat{x}) = \frac{\sigma_{x\hat{x}} + C_3}{\sigma_x\sigma_{\hat{x}} + C_3}$$

where $C_1$, $C_2$ are constants $K_1$, $K_2$ multiplied by the dynamic range of pixel values then squared and $C_3 = \frac{C_2}{2}$

VGG perceptual loss is a loss function that has been used in deep learning image processing tasks like image restoration and super resolution based on the VGG-19 network that is pre-trained on a large dataset of images. [7] The high level features extracted from this network are used to measure perceptual similarity between two images and the loss function is the mean squared error between the two feature maps computed at a given layer of the VGG-19 network. Since high-level features capture both the global and local features of images, minimizing the loss function leads to more visually appealing results in terms of textures and patterns.

The project implementation uses Binary Cross Entropy loss function combined with VGG perceptual loss for adversarial loss and SSIM for the cycle-consistency loss and identity loss.

## 2.4 Evaluation strategy

Frechet inception distance (FID) measures the distance between two distributions of images and is used as a metric to assess the quality of images created by generative models such as GANS. It considers both distributions of images from generators and the reference "ground truth" images from the dataset and measures the distance in terms of feature embeddings produced by the Inception network. In particular it compares the mean and standard deviation of the deepest layer of the network, which capture more abstract and higher-level features of the image rather than the pixel values, leading to more meaningful and interpretable evaluations mimicking human perceptions of image similarity. FID is calculated by computing the Frechet distance between two Gaussians distributions fitted to the feature representation. The implementation used in this project uses the Inception v3 model trained on the ImageNet database.

# 3 Results

In the main training function, the progress was recorded by selecting two samples of both MRI and CT scans and displaying the initial image, the generated image in the other domain and the reconverted image. Furthermore, the generator loss, discriminator loss, cycle consistency loss and identity loss were plotted against the data batch. At the end of training, the Inception model was initialized and the FID was calculated on the testing set of both datasets.

There were many hyperparameter choices as the network architecture contains multiple networks and multiple losses, particularly combined losses for the generators. The output and performance of the CycleGAN was extremely sensitive to small changes in hyperparameters. During training, the first major issue was that the generator and discriminator implementation trained extremely slowly and always ended in convergence failure after trying many different hyperparameter choices and different loss functions.

Figure 4. Example generated MRI using network built and trained from scratch after 200 epochs

As shown in the above figure, the generated image is extremely noisy and blurred, with patterns filling in the negative space and losing almost all of the details of the brain from the input image. Taking the recommendation in the project proposal feedback, the model for both the generator and discriminator was changed to an existing model with pretrained weights. A U-NET network pretrained on the imagenet dataset was used for both the generators and discriminators. Because training time needed to be optimized in order to test multiple hyperparameters, the backbone with the least amount of parameters was chosen, thus resnet18 was used, which has 11M parameters.

In the 2014 GAN paper, algorithm 1 describes the number of steps to apply to the discriminator k, which indicates that for every training iteration the discriminator is trained k times while the generator is trained once. During this project's experiments, it was found that training the discriminator more than the generator or even at a one-to-one ratio caused the discriminator to improve too rapidly and cause the generator to output black or completely distorted and noisy images. Ultimately, updating the discriminator for every 6 generator updates seemed to produce the best results visually, holding the other hyperparameters constant.

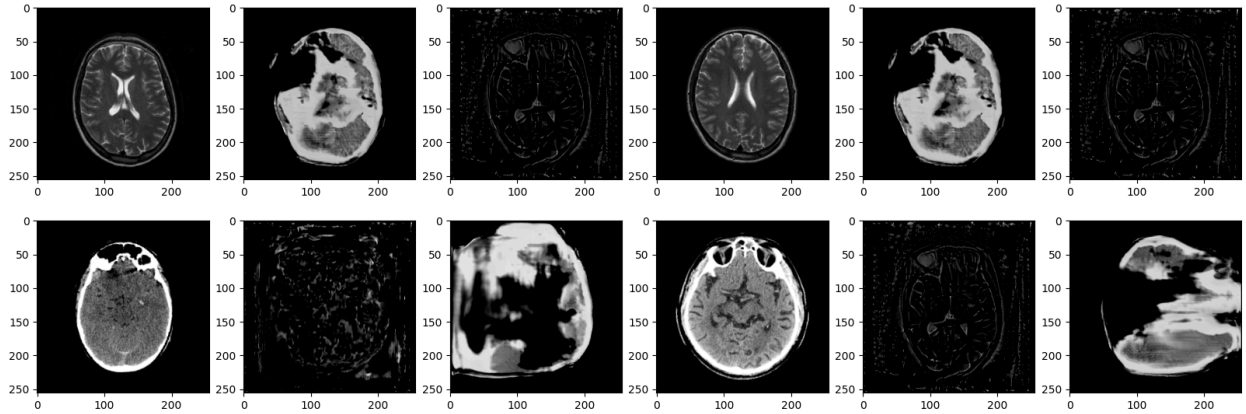

Figure 5. Using 1 to 1 discriminator and generator learning ratio after 200 epochs. Each group of three images are the original image, generated image and reconverted image

Using only Binary Cross Entropy loss as discriminator loss, a lot of the images had artificial looking textures. Furthermore, the training seemed to result in convergence failure around 100 epochs where the generated images would default to several styles. This is indicative of the generator failing to learn a rich enough feature representation since it associates many different input images to the same output. However, after introducing the perceptual loss using VGG when calculating the cycle consistency, the quality of images was improved as they had more clear features of the output domain and the training did not result in mode collapse as often.

| Method | CT to MRI FID | MRI to CT FID |
|---|---|---|
| BCE | 240.24 | 213.93 |
| BCE + VGG Perceptual | 178.15 | 180.16 |

Figure 6. FID score on test dataset after training for 200 epochs

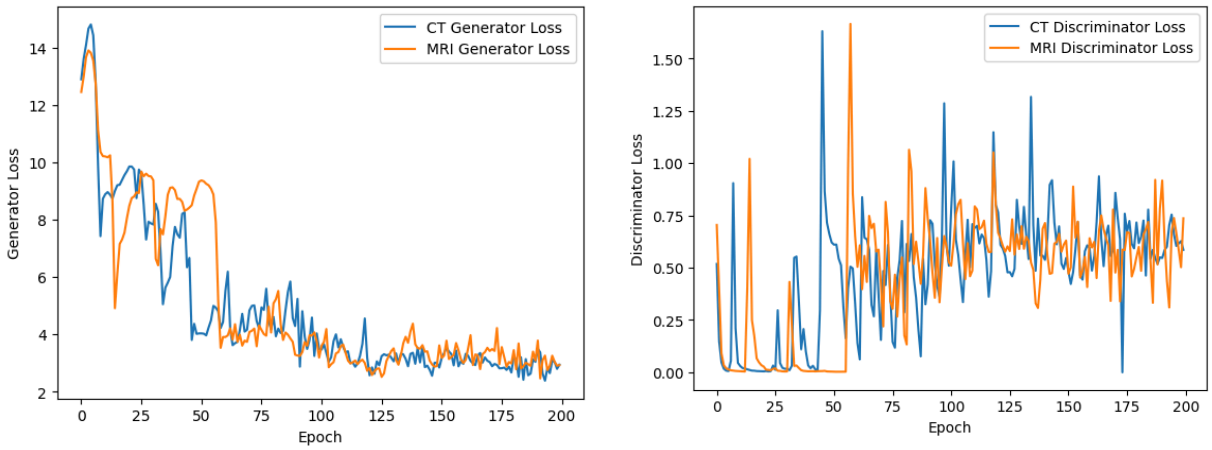


Figure 7. Generator (left) and discriminator (right) loss over 200 epochs

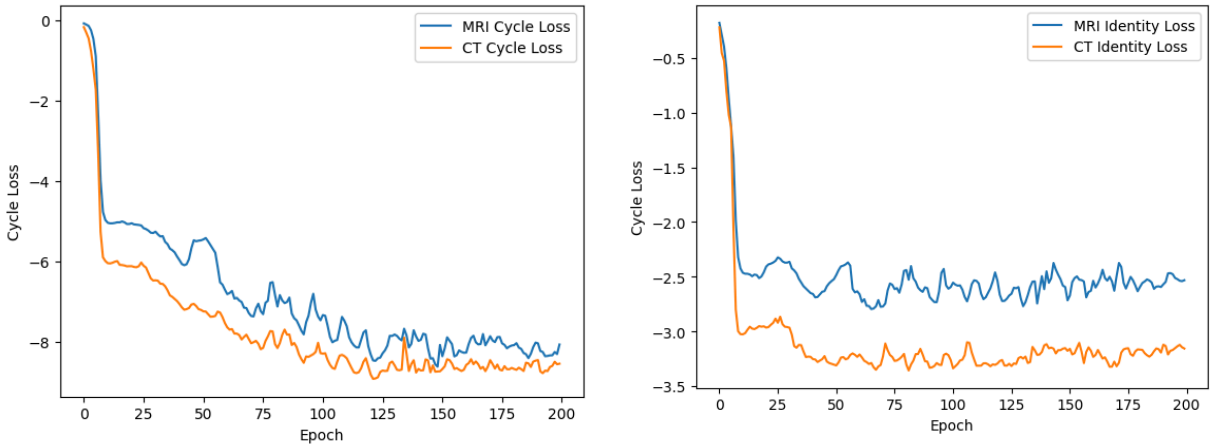


Figure 8. Cycle (left) and identity (right) loss over 200 epochs

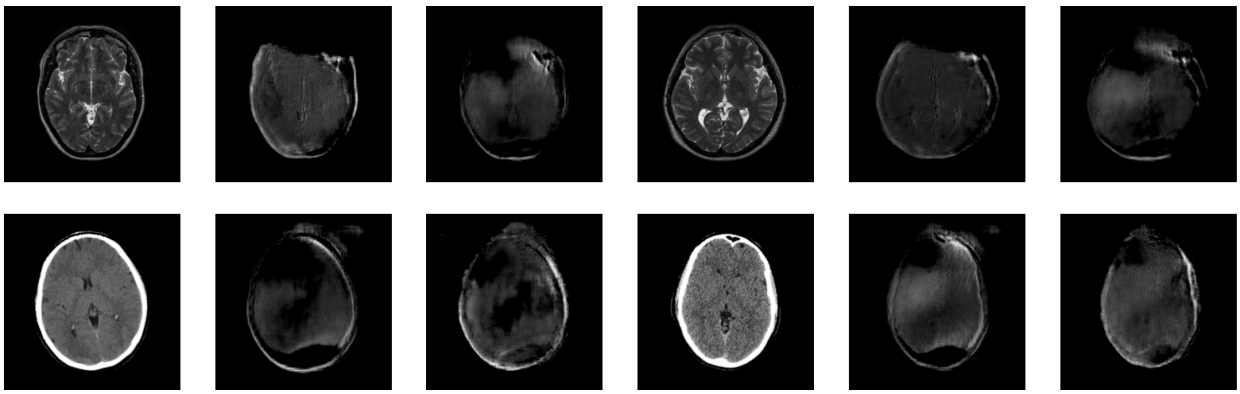

Figure X. Using 1 to 6 discriminator and generator learning ratio after 30 epochs.

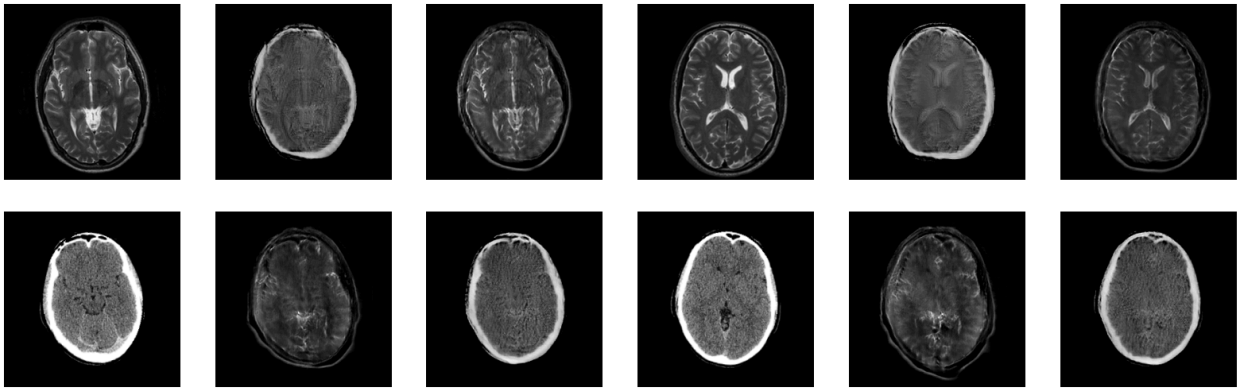

Figure 9. Using 1 to 6 discriminator and generator learning ratio after 200 epochs.

# 4 Discussion

## 4.1 Limitations and Biases

The preprocessing of the medical image slices was done using simple affine transformations based on contour lines and fitted ellipses, which means that information may have been distorted to a point where it is not appropriate in terms of information extraction (evident in the examples of failed preprocessing in Figure X). Due to running training on a single GPU, it was not feasible to try out each permutation of hyperparameters with the entire dataset as training took over 16 hours even when using the optimized pretrained ResNet generators and discriminators. Thus, the optimal set of hyperparameters has not been discovered for this particular dataset. Although FID is a standard in measuring similarity of images for generative networks, it is ultimately not sufficient as a sole measure of effectiveness in medical image synthesis. The consistency of the generators should be tested against real images in a paired dataset.

## 4.2 Analysis of results/outcome with respect to objectives

The biggest takeaway from this project implementation is that training GANs is very difficult as they are extremely sensitive to small hyperparameter changes and finding the proper balance between the discriminator and generator proved to be much more work than expected. Overall, I was able to learn more about and implement simple preprocessing methods for medical imaging, the CycleGAN network and perceptual losses using VGG and the resulting generated images were visually reasonable from a non medical perspective.

## 4.3 Future Work

As discussed in the limitations, the preprocessing portion relies on rudimentary image processing techniques and can be improved through image registration. Feature-based registration should be explored

as a next step for a better dataset consistent with the original goals of the project. Registration allows for multiple transformation types such as deformable (non-rigid) transformations which combine affine transformations with vector fields. ITK and the Python implementation SimpleITK provide an optimized registration framework that can perform such registration.

Regarding the network itself, different losses could be explored such as dual contrastive loss [8], which tries to learn a feature embedding space to group similar images and push apart dissimilar images. As such the discriminator learns more generalized and distinguishable representations which better trains the generator. To further increase the scope of the project, slices beyond the middle range of the brain should be considered. If an updated synthesis model can work with the entirety of brain scans, the Dicom objects could be reconstructed in 3D.

# 5 References


[1] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros: “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”, 2017; arXiv:1703.10593.

[2] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros: “Image-to-Image Translation with Conditional Adversarial Networks”, 2016; arXiv:1611.07004.

[3] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros: “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”, 2017; arXiv:1703.10593.

[4] Justin Johnson, Alexandre Alahi, Li Fei-Fei: “Perceptual Losses for Real-Time Style Transfer and Super-Resolution”, 2016; arXiv:1603.08155.

[5] Vaskovi, J. (2023, April 12). Normal Brain MRI. Kenhub. Retrieved April 13, 2023, from https://www.kenhub.com/en/library/anatomy/normal-brain-mri

[6] Heusel, Martin; Ramsauer, Hubert; Unterthiner, Thomas; Nessler, Bernhard; Hochreiter, Sepp (2017). "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium". Advances in Neural Information Processing Systems. 30. arXiv:1706.08500.

[7] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi: “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network”, 2016; arXiv:1609.04802. perceptual loss https://arxiv.org/abs/1603.08155

[8] Ning Yu, Guilin Liu, Aysegul Dundar, Andrew Tao, Bryan Catanzaro, Larry Davis, Mario Fritz: “Dual Contrastive Loss and Attention for GANs”, 2021;