MRI Image Translation Using CycleGAN with U-Net Generator: A Detailed Report

1 Introduction

Magnetic Resonance Imaging (MRI) is an essential tool in medical diagnostics, offering multiple modalities such as **T1-weighted** and **T2-weighted** scans. Each modality highlights different tissue properties, making both crucial for diagnosis. However, acquiring multiple modalities can be expensive, time-consuming, and sometimes impractical for patients.

To address this, **image-to-image translation** techniques can synthesize one modality from another. In this project, we focus on translating **T1-weighted images to T2-weighted images** and vice versa using a deep learning approach based on **CycleGAN (Cycle-Consistent Generative Adversarial Network)**, with a **U-Net architecture** used for the generator.

2 Objective

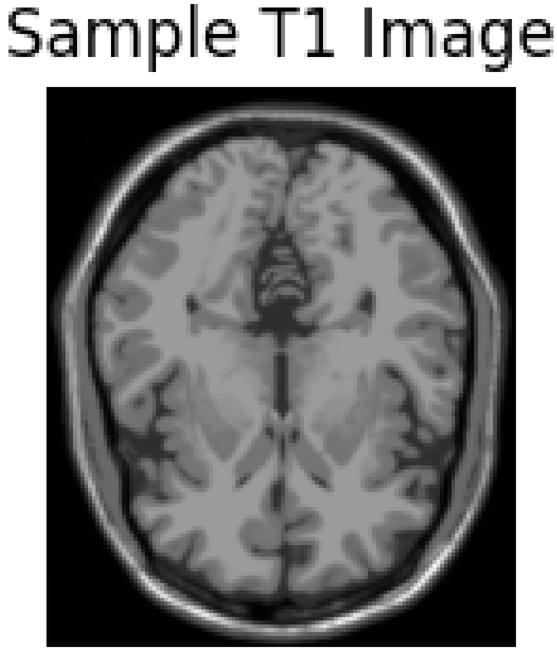
The primary goal of this project is to:

- Translate MRI images from T1 to T2 modality and back.
- Preserve structural information during translation.
- Maintain consistency between real and reconstructed images (cycle consistency).
- Demonstrate the capability of GANs to handle domain translation tasks in medical imaging.

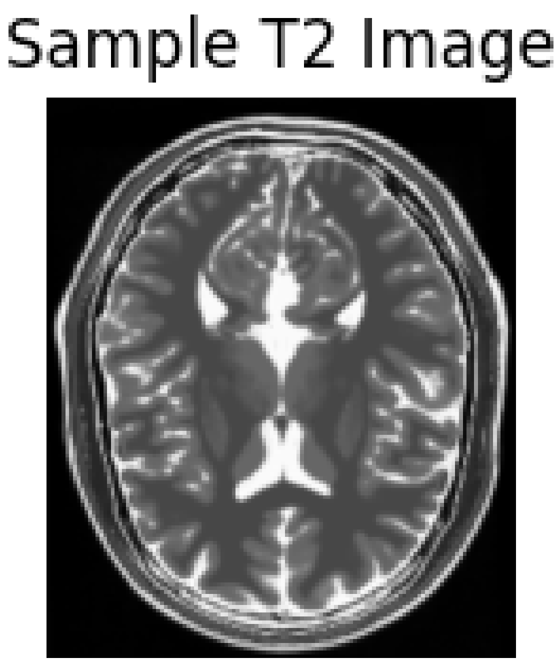
3 Dataset Overview

The dataset used in this project consists of two unpaired sets of MRI images:

- **T1-weighted images**: Show anatomical detail with bright fat and dark water.



- **T2-weighted images**: Highlight fluid (e.g., cerebrospinal fluid) while fat appears darker.



These images were loaded from a structured directory and preprocessed for deep learning training.

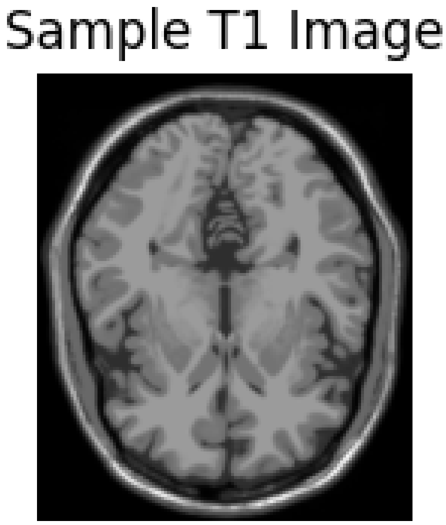
4 Preprocessing Pipeline

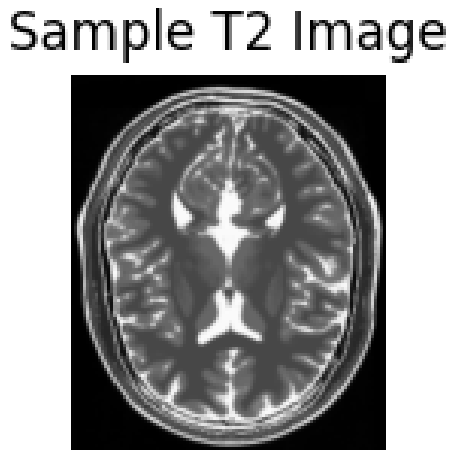
4.1 Image Resizing

All MRI scans were resized to a uniform resolution (64×64 pixels) to ensure computational efficiency and compatibility with the network structure.

4.2 Normalization

Images were normalized to a pixel value range of [**-1, 1**] to stabilize GAN training.





4.3 Reshaping

The dataset was reshaped into four-dimensional tensors required by convolutional layers (batch size, height, width, channels).

4.4 Batching and Shuffling

Efficient data pipelines were constructed using TensorFlow's `Dataset` API to feed data into the model in shuffled batches, improving training dynamics and convergence.

5 Architecture Description

5.1 Generator: U-Net Based

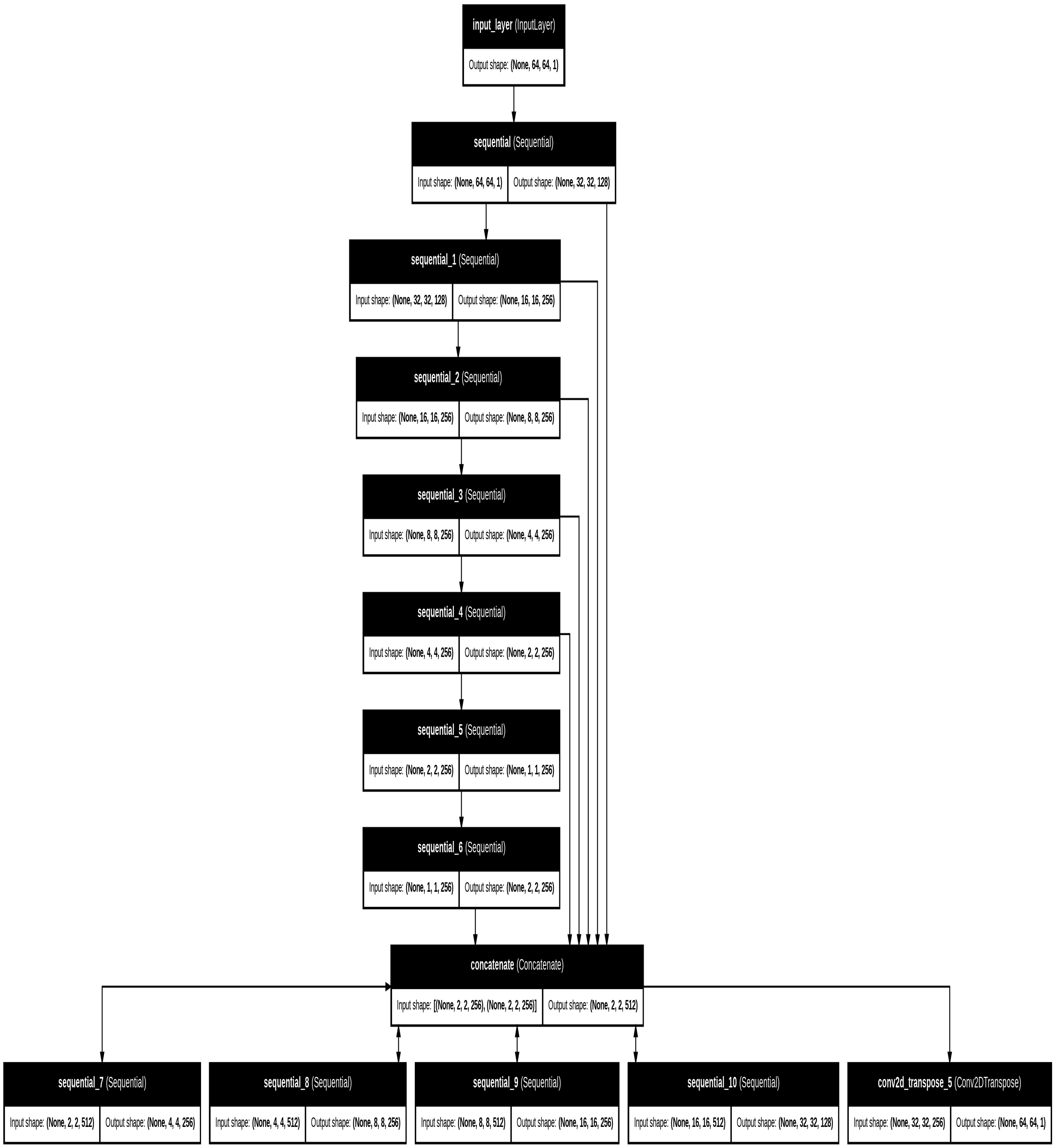
Each generator in the CycleGAN follows a **U-Net architecture**, which is an encoder-decoder structure with skip connections. This allows:

- Downsampling to extract hierarchical features.

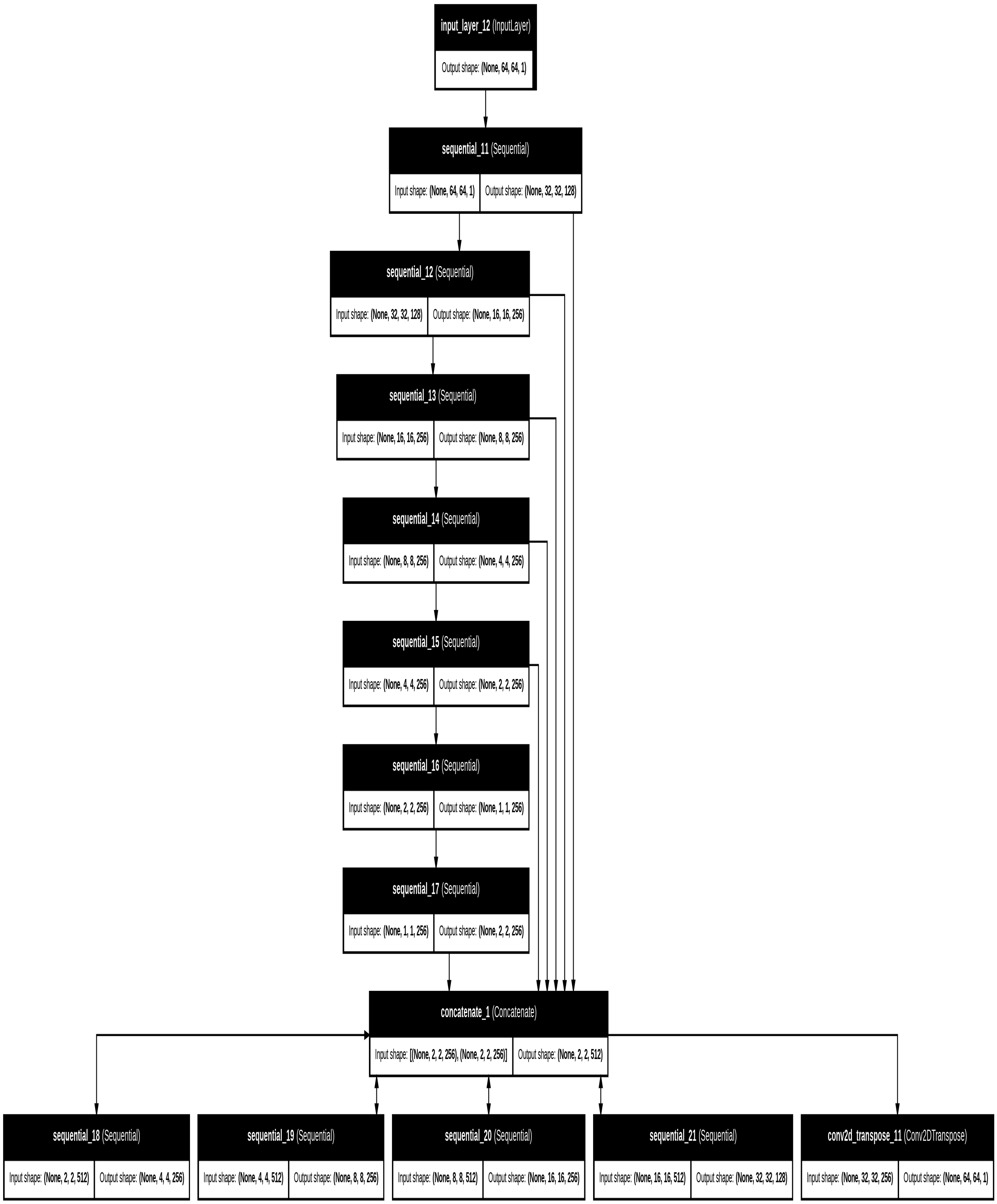
- Upsampling to reconstruct the image while preserving local details through skip connections.

Two generators were used:

- **Generator G**: Translates from T1 to T2.



- **Generator F**: Translates from T2 to T1.



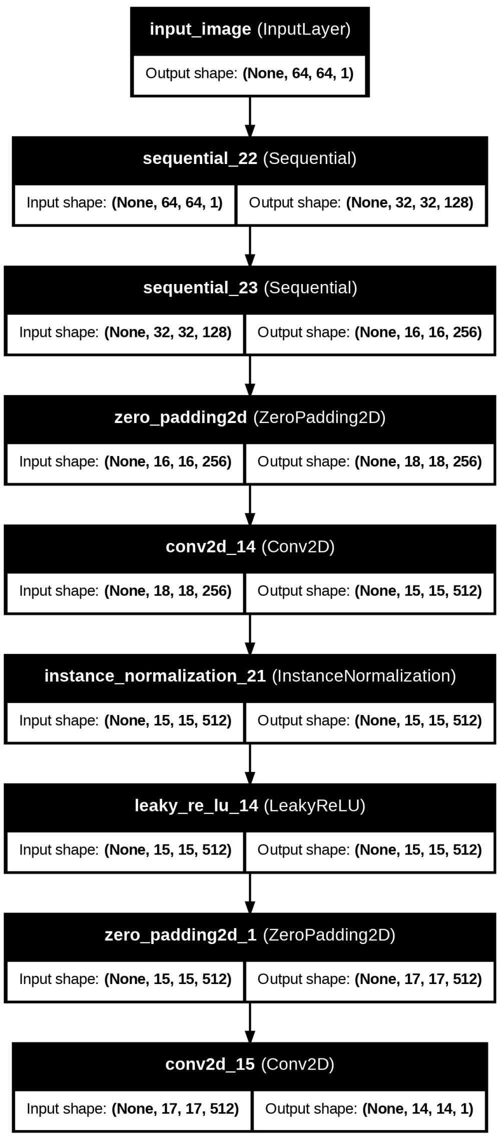
5.2 Discriminator: PatchGAN

Rather than classifying entire images as real or fake, we use a **PatchGAN** discriminator that classifies overlapping image patches. This:

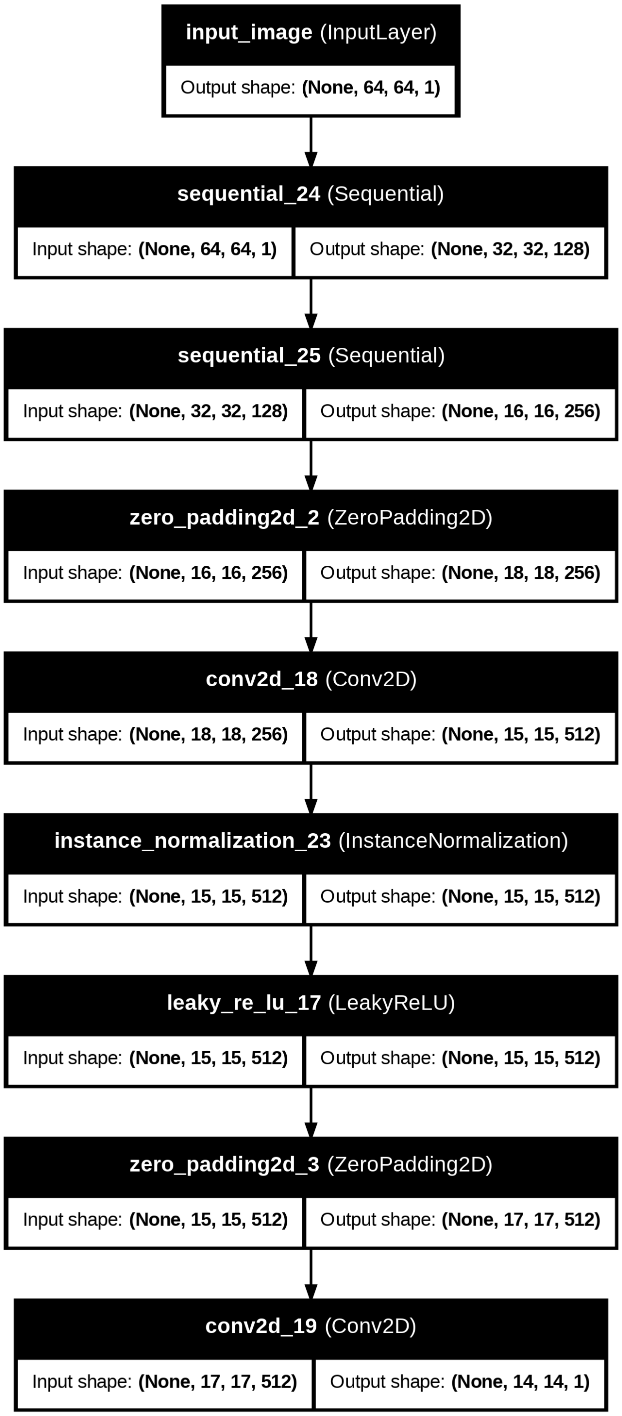
- Focuses on high-frequency structures.
- Provides local feedback that helps the generator preserve texture and fine details.

Two discriminators were used:

- **Discriminator X**: Evaluates the realism of T1 images.



- **Discriminator Y**: Evaluates the realism of T2 images.



6 Cycle Consistency Loss

One of the core innovations of CycleGAN is **cycle consistency**:

- If an image is translated from domain A to B, and then back to A, the result should resemble the original.
- This ensures that the transformation retains meaningful content, even in the absence of paired data.

Cycle Example:

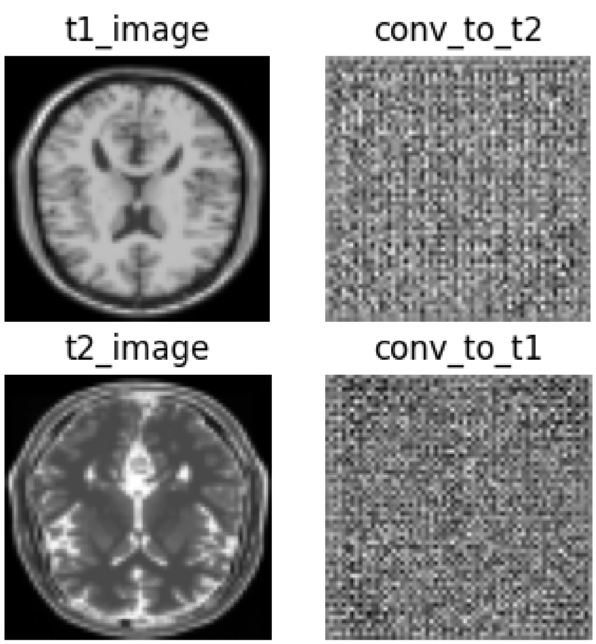
1. T1 → T2 (Fake T2)
2. Fake T2 → T1 (Reconstructed T1)
3. Compare Reconstructed T1 with Original T1 (cycle loss)

This bidirectional training setup allows the model to generalize better across unpaired domains.

7 Training Process

Though the model in this version is not trained yet, typically:

- Both generators and discriminators are trained alternately.
- Losses used include adversarial loss, cycle consistency loss, and identity loss.
- Training continues until generated images from T1 closely resemble real T2 images, and vice versa.



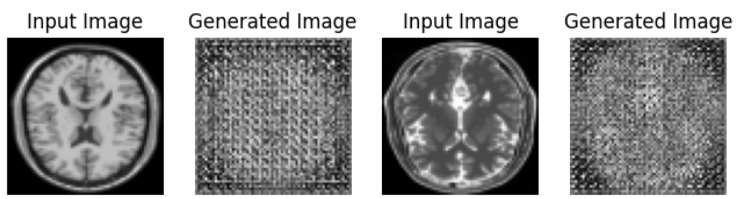
In this setup, we perform a **forward pass only** to visualize what the untrained generator outputs look like.

8 Visualization

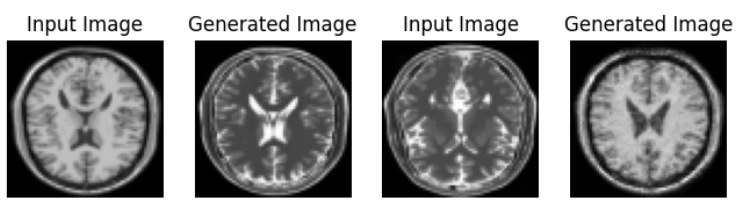
After loading and preprocessing the images:

- A random **T1 image** is passed through the **T1 → T2 generator** to obtain a synthetic T2 image.
- A random **T2 image** is passed through the **T2 → T1 generator** to obtain a synthetic T1 image.
- These generated outputs are visualized alongside the original inputs.

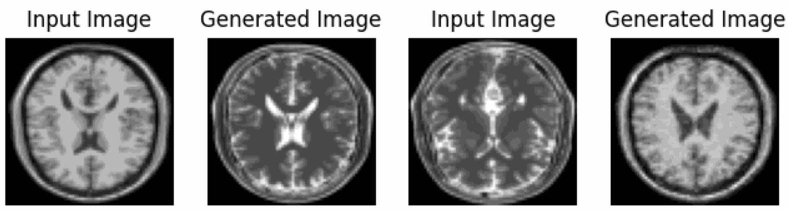
Epoch 1



Epoch 300



9 Result Example



10 Applications in Medical Imaging

This approach has practical potential in the medical field:

- **Modality Synthesis**: Generate missing modalities when only one is available.
- **Data Augmentation**: Create synthetic training data for other models.
- **Cross-Modality Registration**: Improve alignment of multimodal images.
- **Assistive Diagnosis**: Help radiologists by presenting alternative visualizations.

11 Conclusion

This project demonstrates a foundational implementation of **CycleGAN with U-Net architecture** for MRI modality translation. While the current model is untrained, it provides the structural framework necessary to learn meaningful mappings between T1 and T2 images. With proper training and tuning, this system could enhance diagnostic workflows, reduce imaging costs, and open new possibilities in medical AI.