Knee Magnetic Resonance Image Synthesis with 3D CycleGAN

Dominique A Barnes, William C Wang
CycleScan (https://github.com/dbarnes16/CycleScan2)

1. Introduction

Magnetic resonance imaging (MRI) of the anterior cruciate ligament (ACL) is a noninvasive way to track the integrity of the tissue. There is a diagnostic value to MRI as high signal intensity can be a sign of inflammation or injury to the area. Tracking changes in signal intensity overtime can help monitor injuries and improve rehabilitation strategies for patients that undergo ACL surgery. However, different hospitals may use different MRI modalities which will change the signal intensity of the images. This is particularly challenging in clinical trials where multi-center studies are conducted using various MRI modalities to measure the progress of patients after surgeries. The goal of this project was to utilize a deep learning based synthesis of MRI modalities for two clinical trials. After an extensive literature search, our group found that cycle Generative Adversarial Networks (cycleGAN) would be a suitable deep learning model to implement. The original cycleGAN model was introduced in this paper by Zhu et. al [1]. However, we will be re-implementing it by updating the libraries to be able to process three-dimensional grayscale images as opposed to the original work, which was done with two-dimensional color images using the Tensorflow library.

2. Methodology

2.1 Data

The MR images are from patients that had gone through ACL surgery and were included in ACL clinical trials. There were 96 patients that underwent CISS imaging 2-years post-operation and 68 patients returned for their 6-year follow-up and underwent DESS imaging [2,3]. The non-surgicial limb was included in the dataset in order to ensure no metallic artifacts were in any of the training images. This ensured that the model would not include any metal artifact hallucinations in the generated images.

2.2 Data Preprocess

The data was preprocessed by being cropped from the original size of 512x512x100 to 256 x 256x20 slices of normalized voxel values ranging from [-1,1]. The model training was conducted using a Google Cloud AI Notebook with an NVIDIA T4 GPU (16 vCPUs, 104 GB RAM) with an average of 6 hours to run the model through 100 epochs.

2.3 Architecture

The cycleGAN model was introduced by Zhu et al. utilizing a Generative Adversarial Network with two generators and two discrimnators that learn to capture specific characteristics of one collection of images and map the characteristics to be translated into another collection of images in the absence of paired training examples. Instance normalization was used. Used an Adam optimizer with a learning rate of 0.0002 and a batch size of 1. The generators use modules of the form convolution-InstanceNorm-ReLu

with 4 filters (Figure 1) and the discriminators use the module of the form convolution-InstanceNorm-Dropout-ReLu with a dropout rate of 50% (Figure 2).

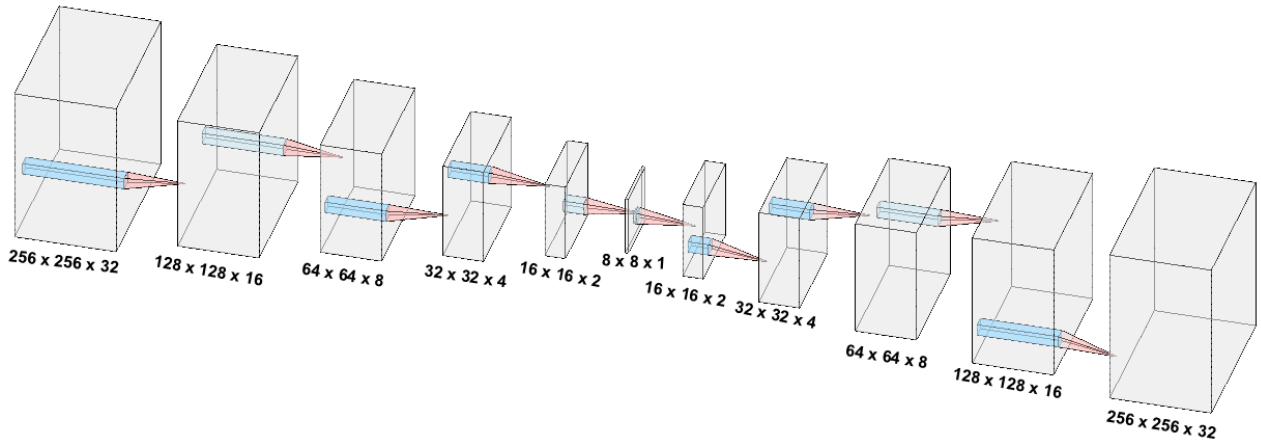


Figure 1. U-Net architecture for the generator

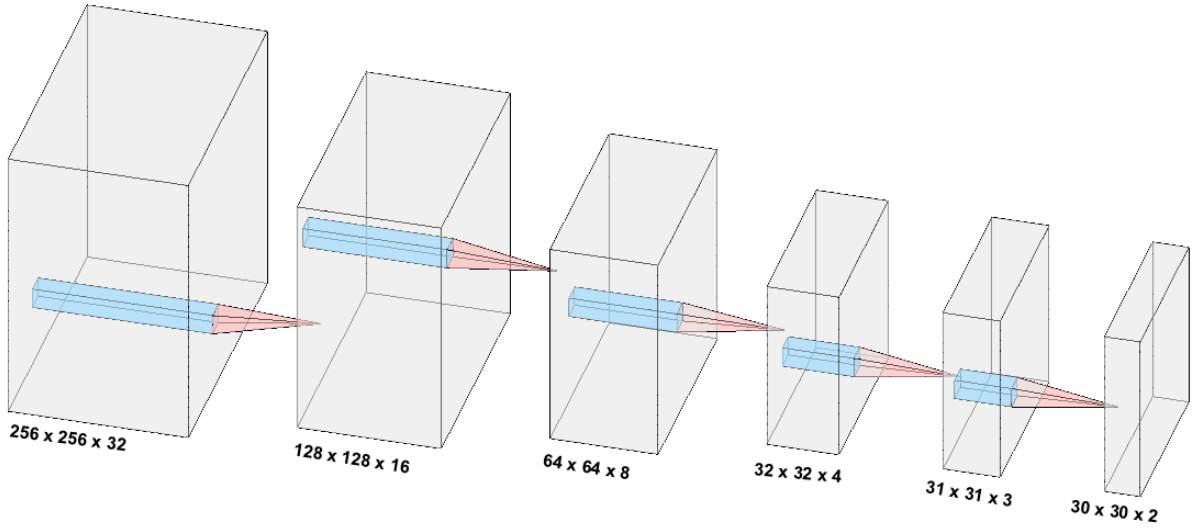


Figure 2. Discriminator architecture

The system overview highlights how the generators and discriminators are used in the model to create realistic generated images (Figure 3). The process begins with the source domain that is used as the input for the generator. The generator then creates a generated image which is used as an input to the discriminator. The goal of the discriminator is to distinguish whether or not the input image is real or fake. The loss functions then are used to guide the generator to improve the generated image. This cycle continues until convergence is achieved where the discriminator can no longer correctly distinguish between the generated image and the real image, or until the final amount of epochs is reached.

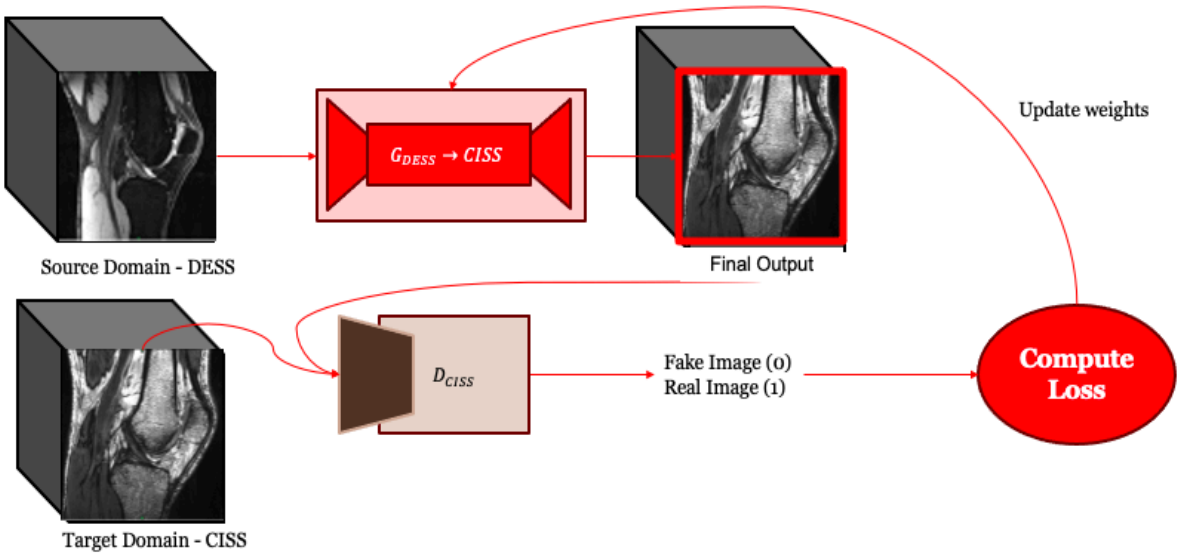


Figure 3. Example of the system overview for Generator and Discriminator D (DESS to CISS)

2.4 Loss Functions

The adversarial loss function used was binary cross entropy. Binary cross entropy provides a measure of dissimilarity between the generated image and the real image, tracking incorrect labeling of the data and penalizing the model if deviations occur in classifying the images. The generators aim to minimize the objective function, while the discriminators aim to maximize it (Equation 1) [1].

$$L_{GAN}(G, D_Y, X, Y) = E_{y \sim p_{data}(y)}[logD_Y(y)] + E_{x \sim p_{data}(x)}[log(1 - D_Y(G(x)))] \quad (1)$$

Cycle consistency (Equation 2) and identity (Equation 3) loss were measured by mean squared error. Cycle consistency loss was used to ensure that the mappings should be reversed of each other, which encforces forward-backward consistency. Identity loss encourages mapping to maintain contrast consistency. This is particularly important in this application as the signal intensities of both modalities are very different.

$$L_{cyc}(G, F) = E_{x \sim p_{data}(x)}[||F(G(x)) - x||_1] + E_{y \sim p_{data}(y)}[||F(G(y)) - y||_1] \quad (2)$$

$$L_{ID}(G, F) = E_{y \sim p_{data}(y)}[||G(y) - y||_1] + E_{x \sim p_{data}(x)}[||F(x) - x||_1] \quad (3)$$

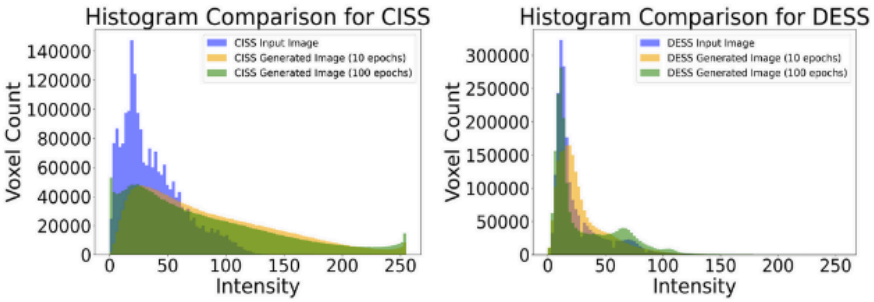
The full objective with a lambda of 10 to control the relative importance of the objectives:

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F) + \lambda L_{ID}(G, F)$$

For further quantativetive analysis we measured the Kullback-Leibler (KL) divergence between the two histograms of the input image (X) and the generated image (x') (Equation 4). KL divergence was useful to quantify the difference between two histogram plots, however, we did not include it in the overall loss functions included in updating the weights, it was strictly for quantitative analysis purposes.

$$D_{KL}(X||x') = \sum_{x \epsilon X} P(X) log(P(X)/P(x')) \quad (4)$$

3. Results



KL Divergence	CISS	DESS
Input || Generated Image (10 Epochs)	54.169	13.054
Input || Generated Image (100 Epochs)	39.117	8.952

The generator from CISS to DESS(generator C) generally performed much better than the generator from DESS to CISS(generator D). This can be seen in the histograms or by the KL divergence statistics. The suspected reason for this is the fact that CISS is more texture heavy than DESS, so generator D had to increase the sharpness of the image, while generator C didn't have to.

The model was quite unstable and although the images did improve with more training, convergence was never achieved. Which is evident in the loss plots of the cycle loss (Figure 4) as the loss values never decreased below 2.4 for a prolonged period.

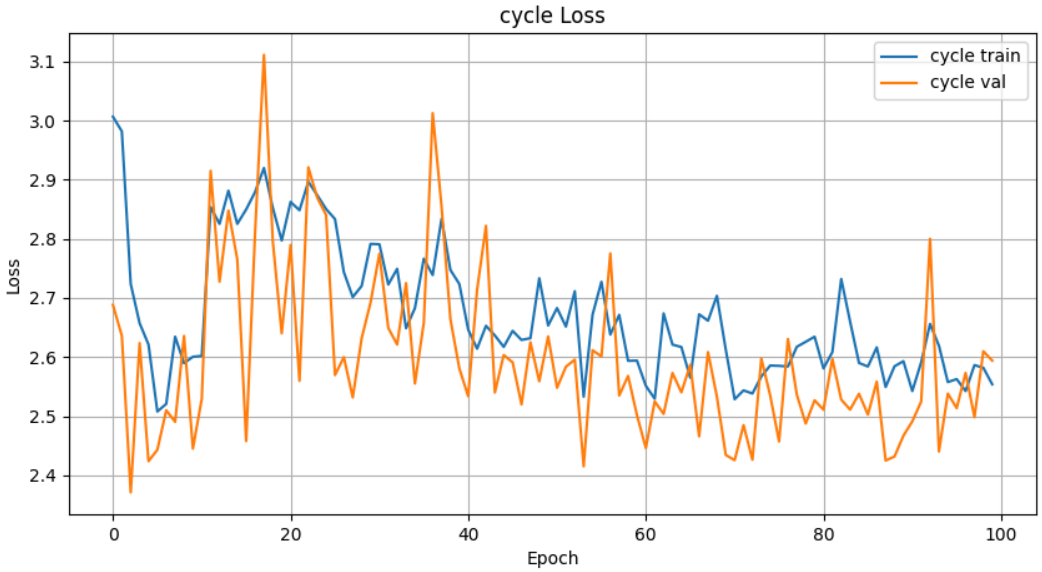
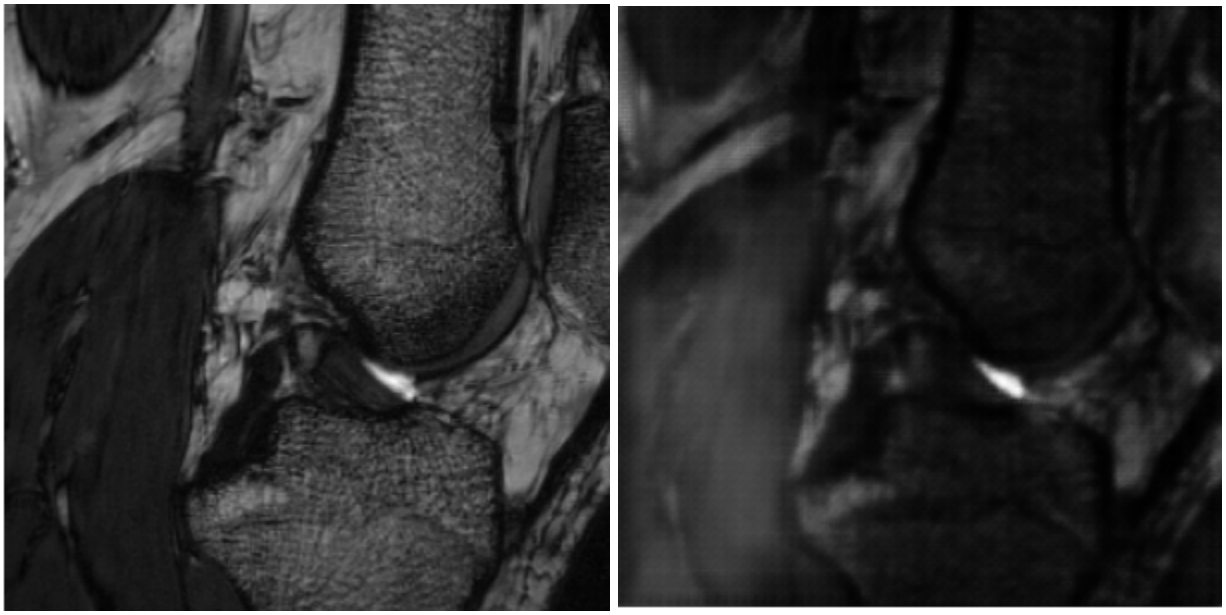


Figure 4. Training and validation cycle loss of the cycleGAN model after 100 epochs

Generator C (CISS to DESS)



Generator D (DESS to CISS)

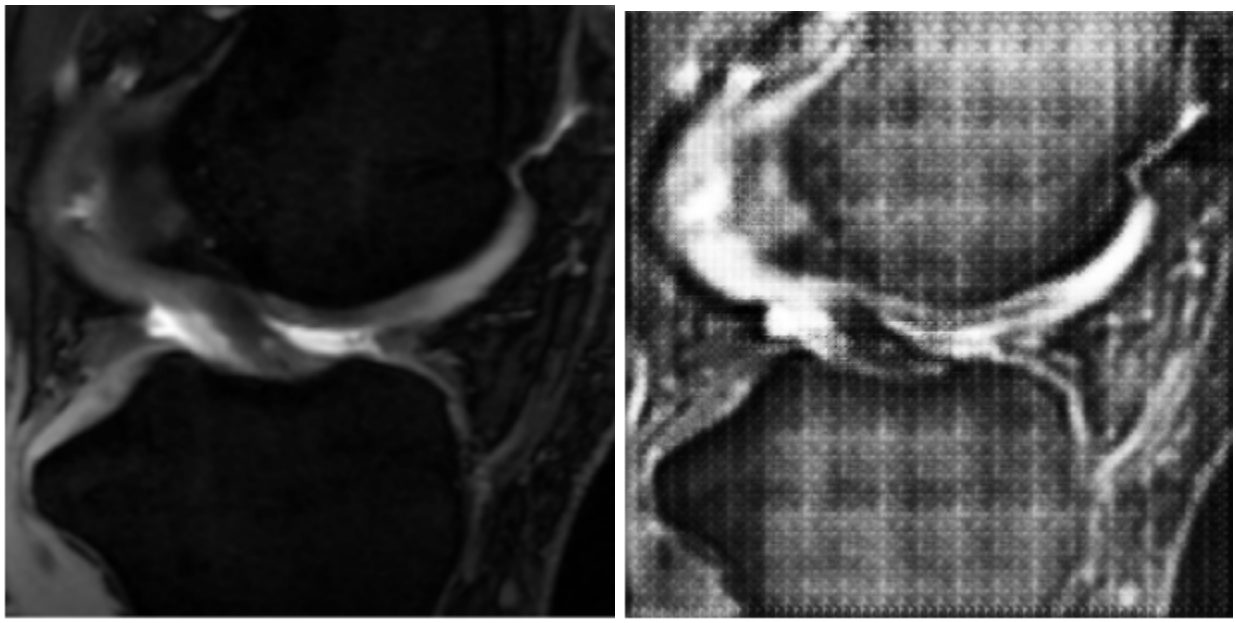
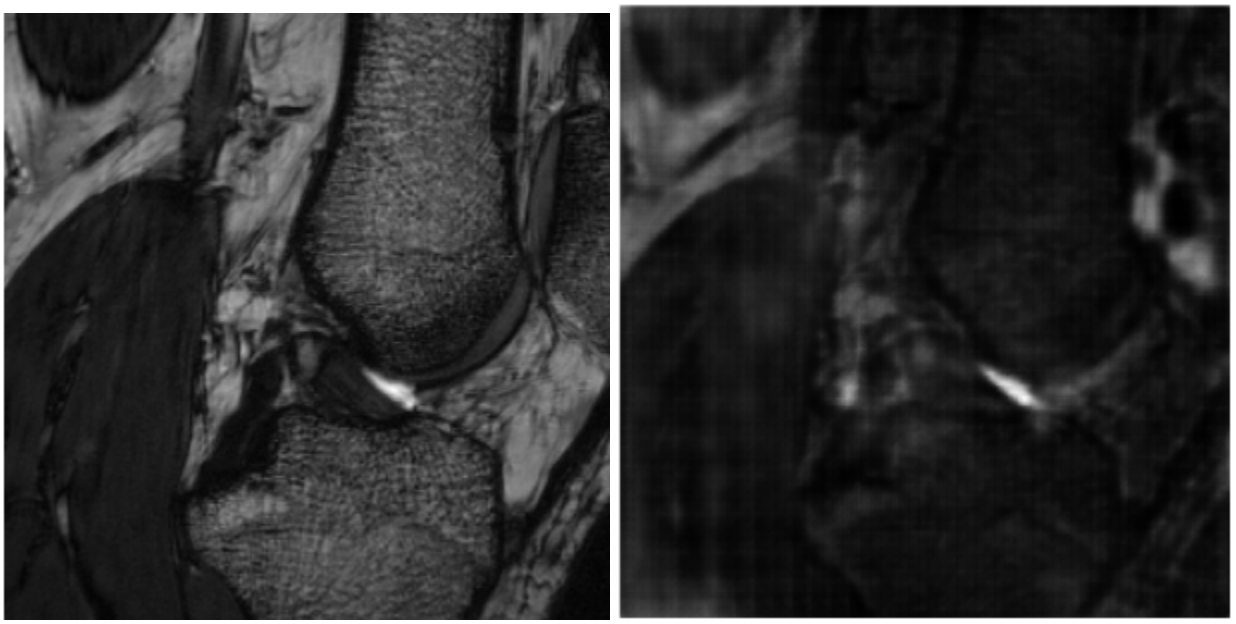


Figure 5. Input images (left column) and generated images (right column) from generator C and generator D after 10 Epochs

Generator C (CISS to DESS)



Generator D (DESS to CISS)

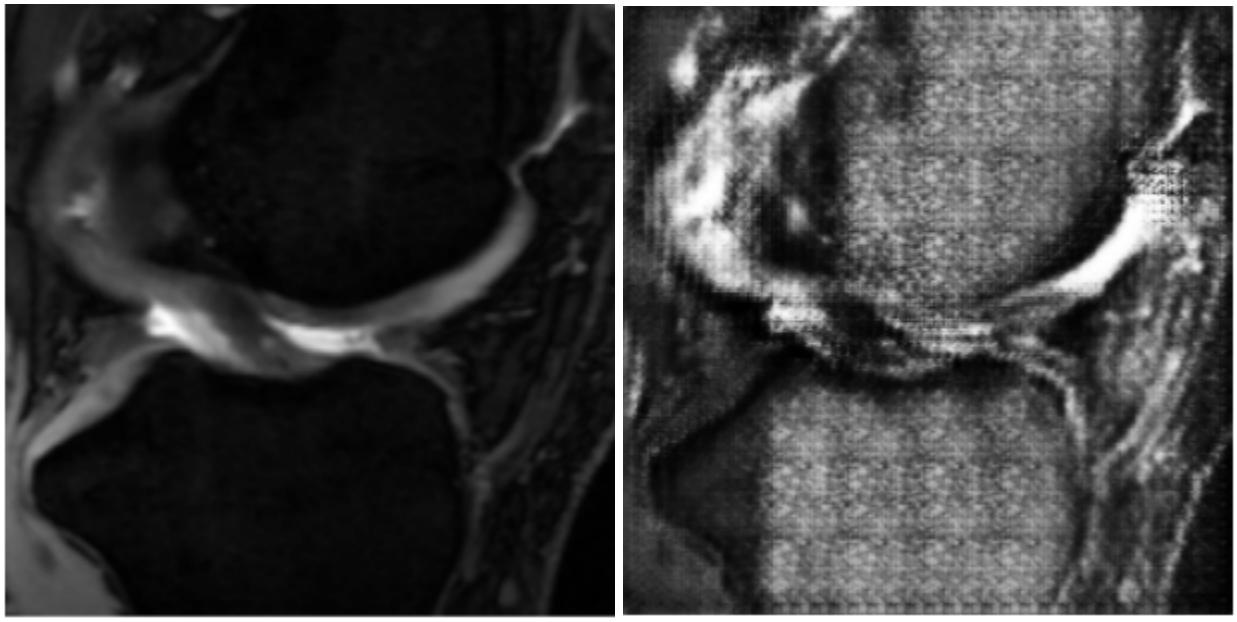
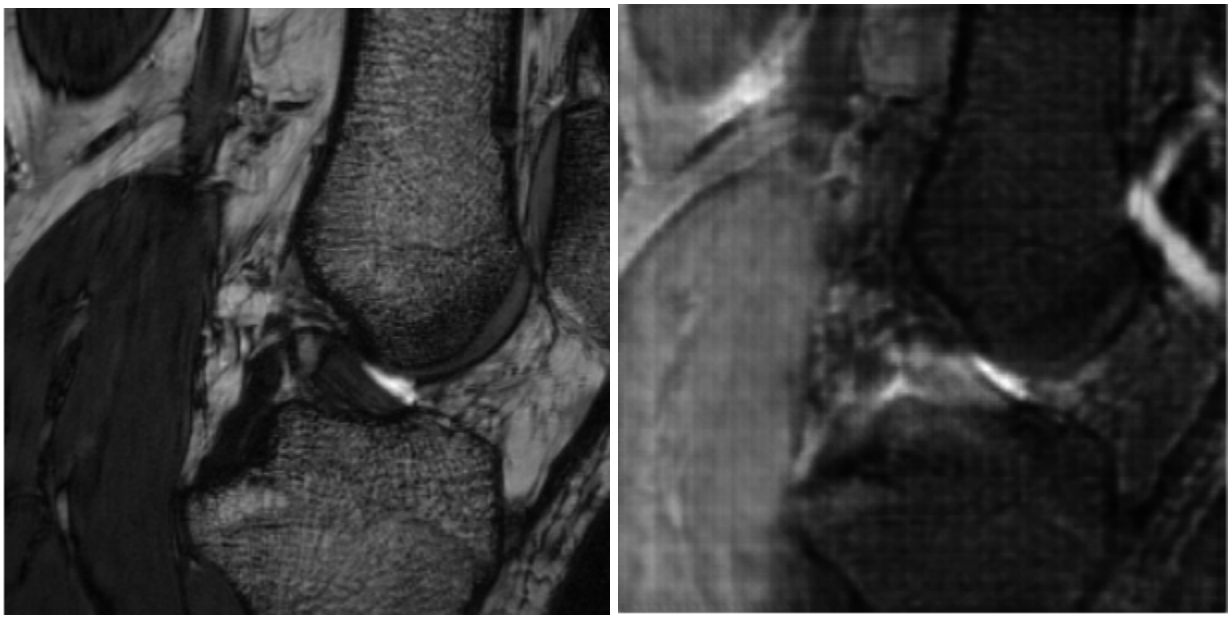


Figure 6. Input images (left column) and generated images (right column) from generator C and generator D after 70 Epochs

Generator C (CISS to DESS)



Generator D (DESS to CISS)

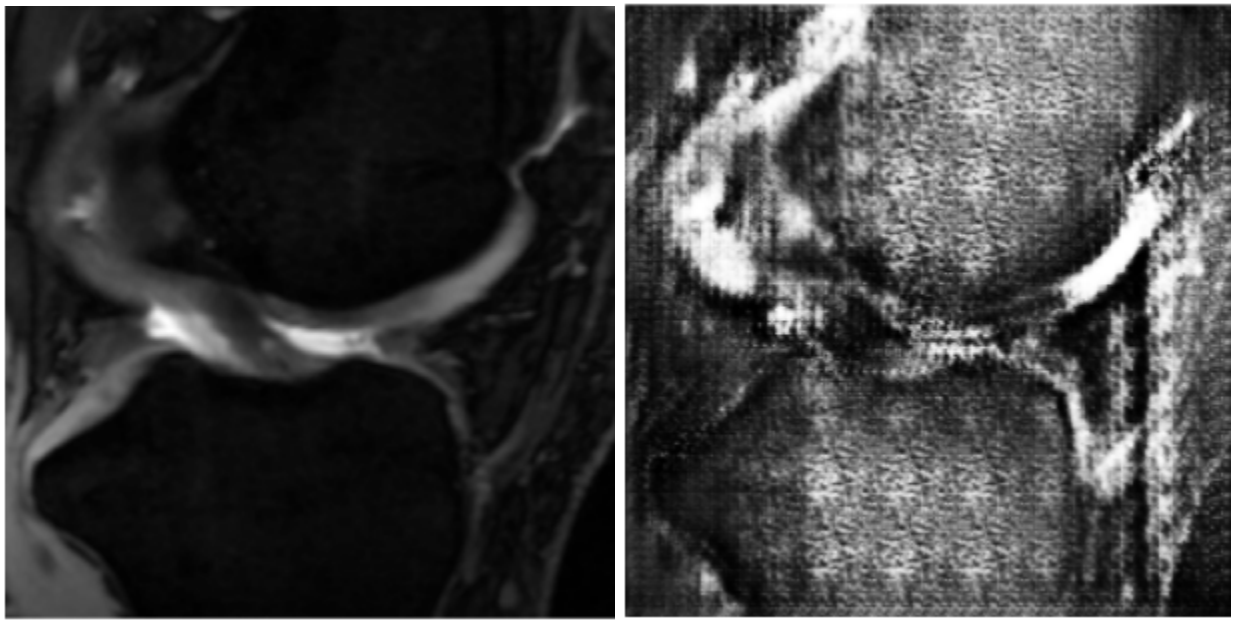


Figure 5. Input images (left column) and generated images (right column) from generator C and generator D after 100 Epochs

4. Discussion

4.1. Challenges

Selecting a model for this task was difficult, as we learned in class GANs are unstable and diffusion models have proven to be more stable while training for image generation tasks. However, diffusion models are computationally extensive and very intensive to implement. Therefore, we decided to begin with a cycleGAN due to its ability to achieve image generation with unpaired images. Although we did not have a large dataset, (n=164

total MRIs) the MRI volumes are 3D which meant we had to re-implement the cycleGAN architecture to work with 3D volumes. This was a difficult task to ensure shapes were consistent when running the generator and discriminator models.

Another challenge we faced was determining an early stopping point. We noticed that any epoch beyond 100 produced suboptimal results. The image quality began to diminish and anatomical structures were difficult to identify. Therefore, we decided to reduce the amount of epochs to 100. However, we believe the epochs could be reduced further. Generator C performed better because the task did not require the generated image to have more detail, rather, the generated image is reducing the amount of detail within the image due to the transformation from CISS to DESS. Generator D had the opposite task, which was to add detail to an image that was lacking detail in the input. In DESS images, the bone is very dark, whereas the soft tissue such as cartilage and ligaments are bright. Generator D worked to achieve the bone texture seen in CISS images. However, that resulted in a tradeoff between increasing the anatomical similarity to the bones in the CISS images and decreasing the detail of other anatomical features such as the ACL.

4.2. Reflection

The cycleGAN was able to achieve results that visually worked well. The figures shown (Figures 5-7) have demonstrated how the cycleGAN achieves the transformation between MRI modalities. However, it is important to note that the generated images should only be used for training purposes and should not be used to make clinical decisions.

Our target goal was to optimize the model and receive outputs that were reasonable, which was achieved. However, the produced images could have better resolution and anatomical features which could be used for clinical evaluation. A limitation of this work was the instability found in the cycleGAN models. Latent diffusion algorithms require pre-trained models, such as varational autoencoders or GANs, that compress the images into a vector that captures the important features.[4] Therefore, future work could be to implement a latent diffusion model using the presented cycleGAN as a pre-trained model for extracting important features from the images. The latent diffusion model will provide more stability and potentially better images than our currently implemented cycleGAN. Increasing the dataset and incorporating more clinical trials would also allow for the model to learn from a diverse dataset making the results more robust. From this project we were able to learn how to implement a model from the beginning and how to preprocess 3D volumes to be used as inputs for convolutional layers. We were able to verify how GANs may be unstable for image generation tasks, but the generated images do align well with the goal of creating images that share characteristics between datasets. This class, along with the final project, has allowed us to develop a deeper understanding of deep learning architectures and the building blocks needed to create functional networks.

5. References

[1] Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2223-2232).

[2] Murray, M. M., Flutie, B. M., Kalish, L. A., Ecklund, K., Fleming, B. C., Proffen, B. L., & Micheli, L. J. (2016). The bridge-enhanced anterior cruciate ligament repair (BEAR) procedure: an early feasibility cohort study. *Orthopaedic Journal of Sports Medicine*, *4*(11), 2325967116672176.

[3] Murray, M. M., Fleming, B. C., Badger, G. J., BEAR Trial Team, Freiberger, C., Henderson, R., ... & Yen, Y. M. (2020). Bridge-enhanced anterior cruciate ligament repair is not inferior to autograft anterior cruciate ligament reconstruction at 2 years: results of a prospective randomized clinical trial. *The American journal of sports medicine*, *48*(6), 1305-1315.

[4] Kim, J., & Park, H. (2024). Adaptive latent diffusion model for 3d medical image to image translation: Multi-modal magnetic resonance imaging study. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 7604-7613).