
ECE 697 Final Project Report: Denoising and Intensity Inhomogeneity Correction of MR Images using Deep Learning

Cameron Craig
MLSP Program
University of Wisconsin – Madison
Madison, WI
ccraig3@wisc.edu

Justin Hiemstra
MLSP Program
University of Wisconsin – Madison
Madison, WI
justinhiemstra@protonmail.com

Abstract

Magnetic Resonance Images captured only from the posterior coil array exhibit a smoothly varying low frequency “bias field” which significantly distorts the intensity of the signal across the image plane. This inhomogeneity can be corrected with existing untrained algorithms, but those algorithms also amplify noise in the image. In this project we explore two distinct approaches to train a convolutional neural network to simultaneously denoise and correct the signal inhomogeneity of MR images. Our first approach uses the original implementation of CycleGAN to correct noisy and inhomogeneous images by learning from unpaired real training data. Our second approach started by creating an algorithm to artificially add inhomogeneity and noise to high-quality MR images. The synthetic data obtained from this algorithm was then used to train a U-Net model in a supervised manner to correct the input images. The CycleGAN approach was able to successfully denoise images, but the inhomogeneity present in the available training data was not conducive to the specific task we were investigating. The supervised approach produced successful results on synthetic test data, but real test data is necessary to more thoroughly evaluate its performance.

1 Introduction

1.1 MRI

MRI is a type of medical imaging used widely in clinical settings for patient diagnosis for a few key reasons; it is better at capturing high-contrast images in soft tissues such as the brain, and it does not expose patients to potentially harmful ionizing radiation, as is required in CT and PET scanning [1]. At its core, the technology exploits the fact that some nuclei like hydrogen are able to absorb RF signals when placed in a very powerful magnetic field (modern MRI scanners use fields on the order of 1.5-6 Tesla [8]). The absorption of these RF signals in the hydrogen nuclei induces a "spin polarization," which can then be detected by additional RF coils to construct an image [17].

While MRI may be less harmful than other types of imaging, it is not necessarily the panacea of medical imaging. The trade off lies in the fact that MRI is often quite uncomfortable for patients. Generating the powerful magnetic fields needed for MRI requires comparatively long exposure periods inside narrow tubes surrounded by magnetic coils that can be quite loud when powered on.

To increase an image’s signal-to-noise ratio (SNR), additional coils are sometimes placed directly on the patient in a way that could be likened to donning a suit of armor. While these additional coils aid in creating crisper, more homogeneous images, they also increase the patient’s discomfort, especially

for obese patients that may find the already-confined spaces very restrictive. The ability to generate low-noise, high-signal images without needing the additional RF coils is highly sought after because it may save time, money, and allow clinicians to make more accurate diagnoses, while also improving the patient's experience.

Even in the absence of accessory RF coils, the scanning surface upon which the patient lies, called the "patient table," has an embedded coil array that is often called the "posterior coil". This coil array serves to increase SNR local to whichever portion of the patient is in contact with the table, leading to a predictable pattern of noise diffusion and signal intensity inhomogeneity in the images. In particular, the noise diffusion of MRI is governed by the *Rician* distribution, which arises when Gaussian noise in various MRI raw signal¹ channels is transformed non-linearly through an inverse fast Fourier transform in the process of converting the signals into an observable image. In these images the noise remains uniform in intensity, while the signal drops off with distance from the posterior coil, resulting in images that become relatively dark toward the patient's anterior.

1.2 Problem Description

Clean, homogeneous images are of the utmost importance to clinicians and radiologists when making diagnoses that may mean life or death for their patients. While posterior coil scans generate images with high SNR local to the posterior coil, the critical nature of radiologists' work means that they will often opt to take MRI scans using multiple coil arrays, guaranteeing high image quality. However, this comes at the expense of the patient's comfort and the radiologist's patient throughput, because configuring the additional coils can be very time consuming and requires the patient to lie motionless on the scanning bed for much longer when compared to posterior coil scans.

Thus, it is desirable for radiologists to generate scans that have the quality obtained from using multiple coil arrays, but with the ease, speed, and simplicity of relying only on the posterior coil. The problem we aim to solve in this project is mapping posterior/"single" coil images to their corresponding multi coil equivalent. This requires correcting both the image's inhomogeneity and denoising it.

1.3 Related Works

While there exist many methods both for denoising MRI scans and correcting signal inhomogeneities, problems arise when these two tools are used in conjunction. Current inhomogeneity correction methods unavoidably increase the intensity of whatever noise may be left in the dark portions of the image. Whether the image is first denoised and then intensity-corrected, or first intensity-corrected and then denoised, excess noise will remain in the most heavily-corrected portions of the image. This leads to the non-uniform SNR across the image when using fewer RF coils.

1.3.1 MRI Denoising

Various approaches using machine learning have been attempted to denoise MRI scans. One popular approach explored by [7] uses a set of stacked patch-based convolutional neural networks (CNNs) in conjunction with a "rotationally invariant non-local means filter" in an attempt to deal with the spatially variant noise in MRI. While this method works well for denoising MRI scans, it does little to deal with the specific types of noise and inhomogeneity created by posterior coil scans.

A significant body of related work also exists in the field of low dose computed tomography (LDCT) denoising. Unlike MRI, CT scans subject the patient to ionizing radiation, which, according to [1], can increase the risk of cell mutation and cancer. This risk can be reduced by exposing the patient to a lower dose of radiation, which comes at the cost of a lower SNR in the image. Numerous deep-learning methods have been used to improve the SNR in LDCT images, including CNNs [2], a U-net inspired wavelet network [6], and RED-CNN [3].

However, these works do not specifically address the type of noise or inhomogeneity found in MRI.

¹The raw signals obtained by the MRI machine live in the Fourier domain, often referred to as "k-space" by radiologists

1.3.2 MRI Intensity Inhomogeneity Correction

Many methods exist for correcting signal intensity inhomogeneity in MRI scans. One relatively recent survey of methods[16] divides these methods into two broad categories: prospective and retrospective approaches. Prospective approaches are those that deal with exploiting knowledge of the MRI scanner to correct inhomogeneity, such as by scanning bags of uniform liquid to generate signal dropoff maps, or by algorithmically smoothing images that come from a combination of volume coils (low-but-uniform SNR) and surface coils (high SNR close to the coil, but lower SNR further away from the coil). Retrospective methods, on the other hand, work primarily in the image/signals-processing realm and are more agnostic to the particular machine used to generate the scan. Such retrospective approaches may include segmentation-based methods, wherein tissues are segmented and given a uniform intensity, or the highly popular N3/N4 (nonparametric nonuniformity normalization)[11] that "seeks the smooth multiplicative field that maximizes the high frequency content of the distribution of tissue intensity" [16]. N4 is of particular importance to our project because it is the current tool favored by clinicians for inhomogeneity correction, and because it will serve as a baseline against which to compare our approaches.

Again, while these tools are well suited to inhomogeneity correction, they do little to denoise images.

1.4 Our Contribution

The primary contribution our project makes that is not present in existing works is our focus on the specific kind of inhomogeneity that occurs when capturing only from the posterior coil array. All of the existing work we found that addresses inhomogeneity approaches the problem from a more general standpoint, aiming to correct all types and patterns of inhomogeneity with a single solution. In contrast, our original goal was to target our solutions at addressing a predominantly unidirectional falloff of the signal in the vertical direction. This kind of inhomogeneity is especially interesting when considering the issue posed by the noise in the image. The greater attenuation of the signal on the top of the image versus the bottom creates a highly uneven signal-to-noise ratio across the image. Inhomogenous images that are created with different receive coil configurations do also exhibit this characteristic, but the effect is generally less severe in those examples. By training a model to correct this specific kind of inhomogeneity and noise, the hope is that the model can specialize its filters to the more narrowly defined task and produce better results than a "jack of all trades" model of similar complexity.

2 Methods and Theory

2.1 Project Approach

Our project explores two approaches to correcting noise and signal inhomogeneity in posterior coil MRI scans. The two approaches we implemented, a CycleGAN trained on real data and a UNet trained on synthetic data, were chosen because they offer a different set of strengths and weaknesses, at least in theory.

2.1.1 CycleGAN Approach

The lack of paired training data makes some form of unsupervised learning appealing in our case. In particular, we need a form of unsupervised learning that can learn from the distributions of images that make up our data set, ie the distributions of "single coil" and "multi coil" images. Because Generative Adversarial Networks are designed to learn from distributions, we chose to train a GAN-style model to accomplish our learning task.

Taking this approach has several potential advantages over our UNet approach. First, because the models learn directly from unedited data, there is no need to make any assumptions about the inhomogeneity or noise profiles present in the single and multi coil scans. This may be beneficial because while we assume Rician noise in our scans, there are other types noise in MRI scans that are much harder to model [5].

Second, because our data set is constructed of scans taken by different machines with different scanning parameters, the trained model should be more device-agnostic. This may make the trained model more applicable in clinical settings where many different types of MRI scanners exist.

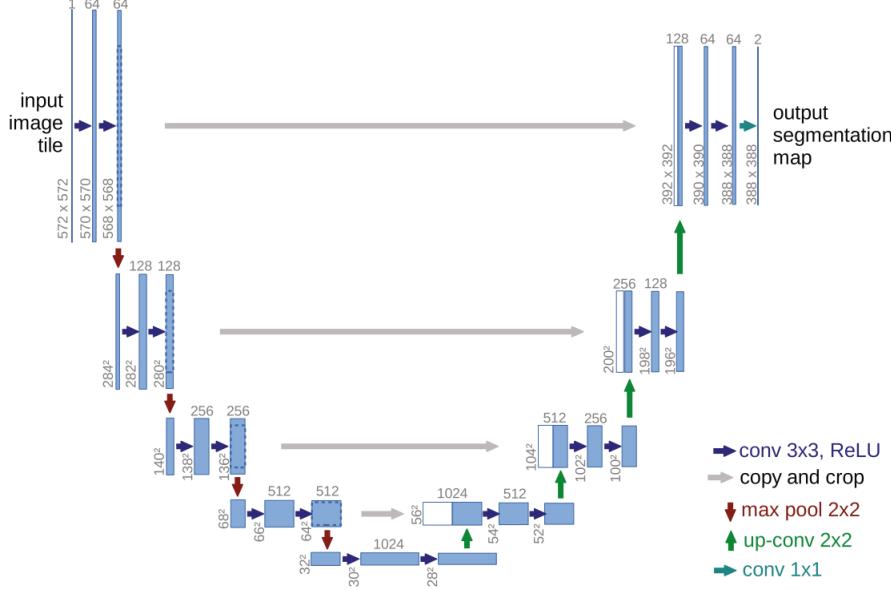


Figure 1: This U-Net architecture was originally proposed by [9] for biomedical image segmentation. We modified the input and output dimensions to re-purpose the architecture for our task.

However, one issue is that GANs, by their nature, make up content. In medical image processing this is often referred to as *image hallucination* [10]. This is obviously undesirable, as one could imagine such scenarios where an unhealthy patient is not correctly diagnosed after a model has obscured their disease. To control the ability of the GAN to hallucinate content, we've chosen to use a CycleGAN model, which augments the standard GAN architecture with a lambda-parameterized "cycle consistency" loss to ensure that the generated output of the model is within some neighborhood of its input. By using a CycleGAN, we can formulate the unsupervised learning problem as learning to map the distribution of single coil MRI scans to the distribution of multi coil scans.

2.1.2 Supervised U-Net Approach

In recent years fully convolutional neural networks have demonstrated state of the art performance on image denoising tasks [13]. It has been shown in [15] that the architecture of an image-to-image deep learning model has a significant impact on the model's ability to perform tasks like deblurring, inpainting, and denoising. Those authors found that models with an "hourglass" shaped architecture are inherently effective at denoising. These networks are able to denoise their input by extracting patterns from the image at multiple different scales. The autoencoder is perhaps the simplest example of such an "hourglass" shaped model. One drawback of the autoencoder architecture is that it bottlenecks the amount of information that can be conveyed from its input to its output. This is a problem because fine details in the input image are important to radiologists, and abstracting those details away into smooth regions is not acceptable.

The U-Net architecture illustrated in figure 1 was originally proposed by Ronneberger et al. in 2015 [9] for the task of segmenting biomedical images. To make the existing PyTorch implementation of the U-Net suitable for denoising images we modified the input and output to be 320×320 pixels, and we changed the downsampling operation from max-pooling to bilinear interpolation.

Like the autoencoder, the U-Net exhibits an "hourglass" shape. This model progressively down-samples the input until it is a fraction of the input dimension, and then upsamples that compressed representation progressively to produce an output of similar (or in our case identical) resolution to the input. Unlike the autoencoder, the U-Net includes residual "skip" connections between the downward and upward paths. These connections bypass the bottleneck in the model, allowing information about the input to flow unimpeded towards the output of the model. Ideally, this model should be able to learn the same compressed representation that an autoencoder uses to capture large-scale patterns in the input, while also passing finer details through to the output. With this structure the model



Figure 2: Two rectangular phantoms were stacked vertically to measure the bias field inside the scanner.

should be well-suited to denoise the input without excessive smoothing or loss of detail. Additionally, the compressed version of the input at the middle of the network should make it easy to detect and correct image-wide intensity inhomogeneity that would be missed on the larger images seen at the beginning and end of the network.

A U-Net is a supervised model which requires both an input, and the corresponding ground truth at the output for each training sample. Like other CNNs, the U-Net is a differentiable model and can be trained using stochastic gradient descent by back-propagating gradients from the output to the input. Supervised deep learning models of this type are comparatively faster to train than GANs, and unlike GANs it is usually easy to achieve convergence during training. Unfortunately, this convenience comes at a cost. While a GAN can be trained on unpaired examples of clean and noisy images, the supervised U-Net requires paired training data. For this project we were unable to obtain access to true paired data samples captured at the same time on a single MRI scanner. To overcome this data access limitation we decided to use synthetic data instead.

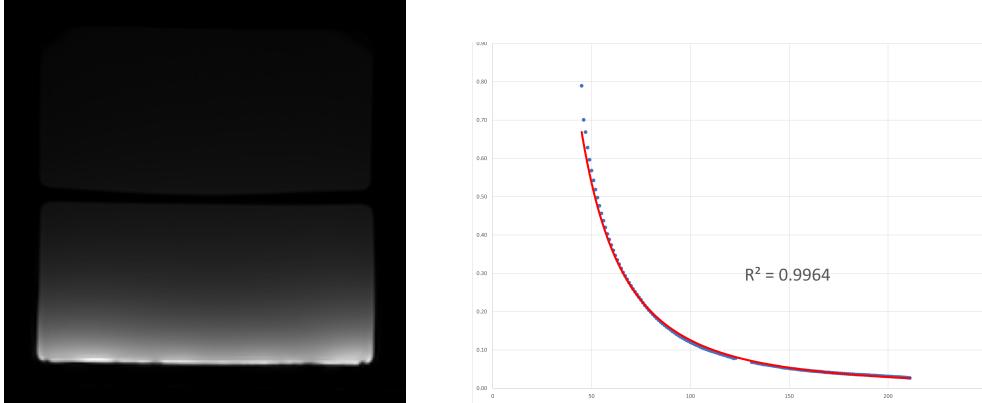
2.1.3 Synthetic Data Generation

The way that an MR image is distorted as a result of the bias field and noise is relatively well understood [5]. The mathematical details are covered in section 2.2.2, but the process of transforming the ideal signal into an observed image can be represented by two major steps which happen in series. First, the ideal signal is biased by the smoothly varying, low frequency bias field in a multiplicative manner. Second, the biased ideal signal undergoes a Rician noising process, where the signal is summed with real and imaginary i.i.d. white Gaussian noise maps. The magnitude of the complex-valued image is taken to produce the final real-valued noisy image.

By performing these two steps in succession on an image, we were able to generate biased and noisy versions of homogeneous multi-coil images. In order to perform the first step of this process we needed a bias field to multiply the image with. As is discussed in section 1.4, for this project we are particularly interested in correcting images that have the particular kind of inhomogeneity that results from using the posterior coil array to receive the signal. In order to write a python script that could generate such a bias field we first needed to study its properties.

In July of 2022 we conducted an experiment at the Wisconsin Institutes for Medical Research (WIMR) Radiology Department. During the experiment we used a GE Healthcare Signa 3T MRI scanner to image two rectangular phantoms (seen in figure 2) using the posterior coil array. The phantoms are designed to produce a uniform response in the image, so by measuring the image of the phantoms we can directly measure the bias field inside the bore of the scanner.

The median values of the rows of figure 3a were fit to a power curve, which can be seen in figure 3b. The equation of the best-fit curve was:



(a) MR Image of rectangular phantoms received by the posterior coil array.
(b) Polynomial best-fit curve for the median intensity of each row of (a).

$$y = \frac{1927.5}{(x + 37)^{2.093}} \quad (1)$$

The power in the denominator of equation 1 is very close to 2, which suggests that the observed signal is predominantly displaying the effects of the inverse square law. Based on this measurement and analysis, we implemented a python function that generates artificial bias field images using this inverse square falloff. The function generates the field by computing the falloff in two dimensions using the distance to the nearest "coil point." The function accepts a list of "coil points" as input, which allows the user to place the simulated coils anywhere inside (or outside) the image plane. To recreate the bias field from the posterior coil array's perspective, we used this function to create a line of evenly spaced "coil points" just below the bottom edge of the image.

One goal in training the U-Net is for the final model to be general enough to work effectively on input images created by MRI scanners of a different make and model than the one we used. In order to prevent the U-Net from over-fitting to the specific bias field strength that we measured, we sampled many parameters randomly from a uniform distribution. These parameters included the intensity at the top edge of the image, the noise gain, and the width and number of simulated coils. 3 independent sets of 5000 different bias fields were generated for training, validation, and testing. Using this algorithm and the pre-generated bias fields, the dataloader created synthetic image pairs in real time as needed during training.

2.2 Mathematical Formulation

2.2.1 CycleGAN Approach

We begin the mathematical formalization of our CycleGAN model by defining two distributions of images. Call them distributions X and Y . In our application, these two distributions correspond to the domain of single coil images and the co-domain of multi coil images. Further, define maps $G : X \rightarrow Y$ and $F : Y \rightarrow X$. These are the model's generators that take in an image from one distribution and produce a corresponding image belonging to the other distribution. Finally, define a pair of discriminators with D_X being the discriminator that determines if images belong to distribution X and D_Y being the equivalent for distribution Y . The set (G, F, D_X, D_Y) constitute the model's learnable parameters. However, to train our CycleGAN it is necessary to first understand its loss function.

A CycleGAN functions much like a standard GAN, where a generator is trained adversarially to generate images belonging to a particular distribution, and a discriminator is trained to determine whether or not those images are real. However, unlike a vanilla GAN, CycleGAN models must learn discriminators for both an input and output distribution, as well as generator maps between those distributions. As such, the model has a distribution-symmetric adversarial loss term:

$$\mathcal{L}_{GAN} = \mathbb{E}_{y \sim Y} [\log D_Y(y)] + \mathbb{E}_{x \sim X} [\log(1 - D_Y(G(x)))] \quad (2)$$

$$+ \mathbb{E}_{x \sim X} [\log D_X(x)] + \mathbb{E}_{y \sim Y} [\log(1 - D_X(G(y)))] \quad (3)$$

[18]

The main innovation of a CycleGAN, however, is its addition of a lambda-parameterized "cycle consistency" loss [18]. This is the loss term that guarantees the model's output image is similar to the input image, and limits the model's ability to hallucinate content.

Mathematically this loss is defined as:

$$\mathcal{L}_{Cyc}(G, F) = \lambda \left(\mathbb{E}_{x \sim X} [|F(G(x)) - x|_1] + \mathbb{E}_{y \sim Y} [|G(F(y)) - y|_1] \right) [18]$$

However, for the sake of intuition it is most useful to view this loss pictorially.

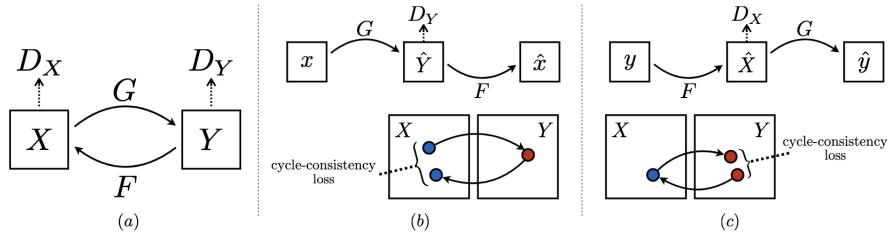


Figure 4: CycleGAN Architecture [18]

Note that in box (b), image $x \sim X$ is mapped to an output image $\hat{y} \sim Y$ and back to $\hat{x} \sim X$. The notion of cycle consistency is to use the L_1 norm to measure the distance between \hat{x} and x . By forcing this distance to be small, the model is prevented from deviating too far from the input image in a way that would destroy input image information².

When combined, these different losses yield the following loss

$$\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{GAN} + \lambda \mathcal{L}_{Cyc}$$

with an overall objective function to find

$$G^*, F^* = \arg \min_{G, F} \max_{D_X, D_Y} \mathcal{L}(G, F, D_X, D_Y) [18].$$

2.2.2 U-Net Approach

The following equation describes the noise model we used for generating the synthetic data:

$$\mathbf{v}(x) = \mathbf{n}(\mathbf{u}(x)\mathbf{f}(x)) \quad (4)$$

Here, $\mathbf{v}(x)$ is the observed biased and noisy image, $\mathbf{u}(x)$ is the ideal image, $\mathbf{f}(x)$ is the bias field, and $\mathbf{n}(x)$ is the function that applies Rician distributed noise to its input. The key observation to make here is that given an ideal image $\mathbf{u}(x)$ it is fairly straightforward to produce a noisy image $\mathbf{v}(x)$. The process to convert from $\mathbf{v}(x)$ to $\mathbf{u}(x)$, however, is less clear, as there isn't a closed-form solution to invert the noising function $\mathbf{n}(x)$. This is essentially the crux of this entire approach; our ultimate end goal of producing clean versions of noisy images is difficult, so we instead train a neural network (which is a universal function approximator) to approximate the inverse of this function for us.

²It has actually been shown that CycleGANs are, in effect, learning a form of steganography[4]

2.3 Project Developments

The initial idea for this project was presented to us in class by Dr. Alan McMillan. Dr. McMillan works at WIMR, and he thought that the problem we are addressing was both relevant for clinical use, and currently unsolved. Originally we thought we might receive paired training data from Dr. McMillan, but unfortunately it wasn't possible to acquire the data in time for this summer project. We then turned our focus to searching for publicly available MRI data online, where we found the FastMRI data set from NYU. While the data set does not contain the paired training samples we were hoping for, it does contain both singlecoil and multicoil MR images saved as raw k-space (frequency domain) data. After finding this data and brainstorming solutions to the problem, we arrived at the two different approaches we have presented in this report. One of the primary reasons we chose to approach the same problem from two very different angles was that we wanted to balance risk with possible reward. The CycleGAN approach was very interesting given its flexible data requirements, but we (correctly) anticipated early on that this approach might not be fruitful for a variety of reasons, and as a backup we pursued the U-Net approach in parallel.

At this phase of the project Cameron worked on developing the supervised dataloader, stochastic noise model, and U-Net PyTorch implementation. Justin worked on reconstructing images from the k-space data, re-training the original CycleGAN implementation, and establishing a high-performance training pipeline using resources provided by the Center for High-Throughput Computing (CHTC). Developing the CHTC training pipeline proved to be challenging, so we began training a preliminary version of the U-Net model in Google Colab. During this training attempt numerous bugs in the training code were revealed, and it became apparent that the main training loop code was getting too messy and difficult to maintain. In an effort to make further training easier, and to enable multi-gpu training, Cameron began to refactor the existing PyTorch code to use the PyTorch Lightning library. This library is fully cross-compatible with PyTorch, and allows the user to abstract away technical details like moving tensors between cpu and gpu, and turning gradients on and off. The result was a cleaner and more organised codebase, better training performance, and automatic compatibility with multi-gpu systems.

For this final report we hoped to train multiple U-Net models with different hyperparameters in order to analyze their affect, and to arrive at final hyperparameter recommendations for optimal performance. Unfortunately it was not possible to complete the training of these models due to an untimely CHTC maintenance outage. Despite this setback we were still able to gather and analyze results from the UNet model trained on Google Colab. A more thorough hyperparameter sweep would be a good candidate for future work studying this approach.

3 Project Results

3.1 CycleGAN Approach

Our CycleGAN was trained using the implementation proposed by [18]³ for 100 epochs with a learning rate of $lr = 0.0002$, and an additional 100 epochs of exponential decay learning. Trained on 2xA100 GPUs with 80GB of GPU RAM, 128GB of computer RAM, and 2.2TB of high-speed NVMe disk, all available through CHTC's high-throughput system, the model took approximately 1.5 days to train.

Initial visual inspection of images produced by the trained model (figure 5) yielded interesting results. From these images, it was clear that the model was learning to denoise images from the single coil domain and reintroduce that noise to images from the multi coil domain.

However, we did not quantify the success of that denoising because a major shortcoming of the CycleGAN approach had become apparent at this point. The data set we used from NYU contained labeled single and multi coil images that exhibited the noise we expected, but the signal inhomogeneity in the single coil images did not appear to be generated by a posterior coil array. Because we are trying to correct a particular type of inhomogeneity, the distributions of inhomogeneity in our raw data are not conducive to our original learning task.

³Code source: <https://github.com/junyanz/CycleGAN>

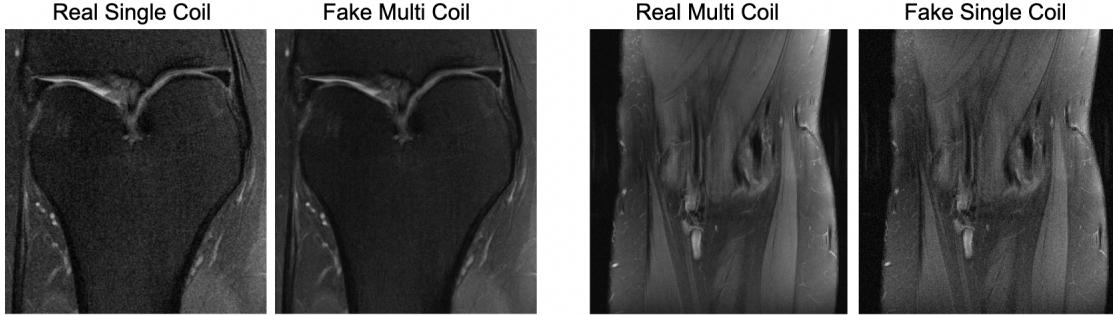


Figure 5: CycleGAN results

At this point, we had seen promising results from our UNet model for both inhomogeneity correction and denoising, so we decided to abandon the CycleGAN approach and focus our entire efforts on training UNet models.

3.2 Supervised U-Net Approach

3.2.1 Baseline for Comparison

When discussing the project early on Dr. McMillan explained that he had previously tried using a bias correction algorithm called N4ITK [14] to correct whole body scans taken with just the posterior coil array. This algorithm is an improvement of the non-parametric nonuniform intensity normalization (N3) algorithm, which was published in 1998 [12]. N3 was an extremely popular choice for correcting bias in MR images for many years, likely in part because it was open source. N4ITK solves a number of problems that arose in the use of N3 [14], and generally speaking does a good job of correcting intensity inhomogeneity in MR images. We chose to use N4 as a baseline to compare our model against because it is a popular and effective tool, and because Dr. McMillan had experience with it. It is important to note, however, that N4 only aims to correct intensity inhomogeneity, as it is not a denoising algorithm.

3.2.2 Evaluation Methodology

A test set of 4,674 MR images of brains were used to evaluate both N4 and the trained model. The synthetic noisy inputs were generated using a noise semi-amplitude of 6% dynamic range, and a maximum attenuation of 5.5% dynamic range. The predictions from N4 and the model were compared to the ground truth images using Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Multi-Scale Structural Similarity Index Measure (MS-SSIM), and the training criterion, which was $L_1 + L_2$ loss.

3.2.3 U-Net Results

Performance on Test Set				
Algorithm or Model	PSNR (dB)	SSIM [0, 1]	MS-SSIM [0, 1]	$L_1 + L_2$ Loss
N4ITK	23.7	0.407	0.894	0.0510
U-Net	28.8	0.733	0.974	0.0235

Table 1: Performance comparison between N4ITK and our trained U-Net on the test data set.

Our trained U-Net outperformed the existing algorithm on all of the metrics we tested. As of the conclusion of this project we were not able to obtain any real paired test data. While these results are impressive and seem to indicate that we have created an effective solution to the problem we were investigating, it will be necessary to re-evaluate the model using real data to make stronger conclusions about its true effectiveness.

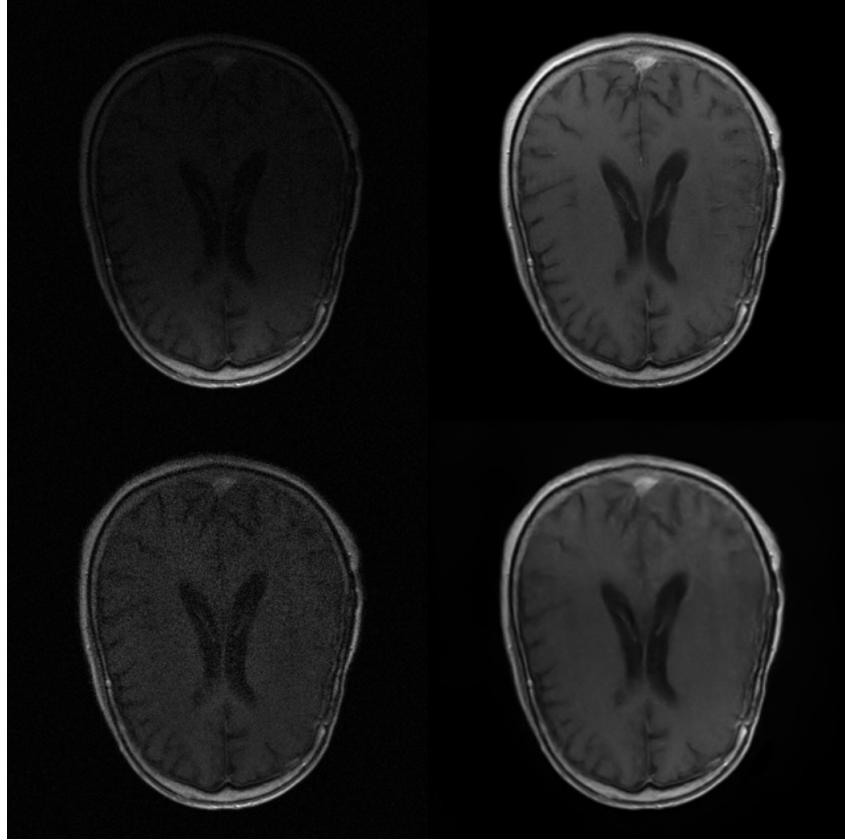


Figure 6: Example result from the test set. Top left: synthetic input, top right: ground truth, bottom left: N4ITK output, bottom right: U-Net prediction.

4 Conclusion

Several conclusions can be drawn from this project. First, we conclude that our approach of using a stochastic noise/bias field model to generate synthetic data that we then use to train a UNet yields promising results when analyzed using PSNR, SSIM, MS-SSIM, and $L_1 + L_2$ loss metrics. However, at this stage it is difficult to make more concrete statements about the clinical efficacy of our model. While our model outperforms the baseline on all of our metrics, it is ultimately a radiologist who decides what constitutes a "good" MRI scan. Further testing must be done to assess whether radiologists approve of our model's output. Moreover, because our model is not "explainable" in the same way something like a decision tree or a spline-fitting algorithm is explainable, FDA approval of our model's use in a clinical setting seems unlikely.

Second, while our model appears to be performing the intended task, we must note that we are currently unable to test the model on non-synthetic data. The fact that the model already appears to show promise is all the more reason that a genuine data set of matched training pairs should be proactively collected to guarantee our model's robustness.

Finally, we point out that there is room for future work. While our model was trained using 2D image slices from MRI scans of knees, it must be noted that MRI scans produce inherently 3D raw data. By potentially learning from 3D representations of the data, our model may be able to pick up on additional information about the noise and inhomogeneity profiles in the scans. We were unable to explore this approach due to data limitations, and the time limitations of the semester.

5 Acknowledgements

We would like to thank and acknowledge several people for supporting us throughout our project.

First, thank you to professor Alan McMillan for giving us the idea for our project and for helping us obtain phantom scans at WIMR. They were invaluable for us to generate realistic inhomogeneity bias fields. Second, thank you to professor Kassem Fawaz for helping us clarify our project early on and for providing us with necessary computing resources, like Colab Pro+.

Finally, thank you to CHTC and its staff for their support and computational resources.

5.1 Summary of Individual Contributions

5.1.1 Cameron

Cameron was responsible for building the bias field model from our scans at WIMR and for applying those bias fields and Rician noise to our data for training with a bespoke UNet on Colab. The models presented in the results section for the UNet approach come from a model trained in Colab.

5.1.2 Justin

Justin was responsible for the initial attempts to train a CycleGAN on our data set. When we realized this approach was limited by low-quality data, Justin switched over to retooling Cameron's UNet code for CHTC's high-throughput GPU cluster. While competition for limited resources and system-wide upgrades in CHTC meant running code was frustrating, when code was able to run we saw model training times drop from ~1 day to ~10 minutes.

References

- [1] D. J. Brenner and E. J. Hall. Computed tomography—an increasing source of radiation exposure. *New England Journal of Medicine*, 357:2277–2284, November 2007.
- [2] H. Chen. Low-dose ct denoising with convolutional neural network, 2016. URL <https://arxiv.org/abs/1610.00321>.
- [3] H. Chen. Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE Transactions on Medical Imaging*, 36:2524–2535, December 2017.
- [4] Casey Chu, Andrey Zhmoginov, and Mark Sandler. Cyclegan, a master of steganography, 2017. URL <https://arxiv.org/abs/1712.02950>.
- [5] Hákon Gudbjartsson and Samuel Patz. The rician distribution of noisy mri data. *Magnetic resonance in medicine*, 34(6):910–914, 1995.
- [6] Kang. A deep convolutional neural network using directional wavelets for low-dose x-ray ct reconstruction. *Medical Physics*, 44,10:e360–e375, 2017.
- [7] José Manjon and Pierrick Coupé. Mri denoising using deep learning. 01 2018.
- [8] National Institute of Health. Magnetic resonance imaging. online, 2020. <https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri>.
- [9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [10] Kyle Shelton. Investigating medical imaging hallucinations. online, January 2022. <https://beckman.illinois.edu/about/news/article/2022/01/07/investigating-medical-imaging-hallucinations>.
- [11] John Sled. A nonparametric method for automatic correction of intensity nonuniformity in mri data. In *IEEE Transactions on Medical Imaging*, volume 17, pages 87–97, 1998.
- [12] John G Sled, Alex P Zijdenbos, and Alan C Evans. A nonparametric method for automatic correction of intensity nonuniformity in mri data. *IEEE transactions on medical imaging*, 17(1):87–97, 1998.
- [13] Chunwei Tian, Yong Xu, Lunke Fei, and Ke Yan. Deep learning for image denoising: A survey. In Jeng-Shyang Pan, Jerry Chun-Wei Lin, Bixia Sui, and Shih-Pang Tseng, editors, *Genetic and Evolutionary Computing*, pages 563–572, Singapore, 2019. Springer Singapore. ISBN 978-981-13-5841-8.
- [14] Nicholas J Tustison, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich, and James C Gee. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310–1320, 2010.
- [15] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018.
- [16] Uros Vovk. A review of methods for correction of intensity inhomogeneity in mri. In *IEEE Transactions on Medical Imaging*, volume 26, pages 405–421, 2007. doi: 10.1109/TMI.2006.891486.
- [17] G.A. Wright. Magnetic resonance imaging. *IEEE*, 14:56–66, January 1997.
- [18] Jun-Yan Zhu and Taesung Park. Unpaired image-to-image translation using cycle-consistent adversarial networks. August 2020. URL <https://arxiv.org/pdf/1703.10593.pdf>.

A Additional Results from the Test Data Set

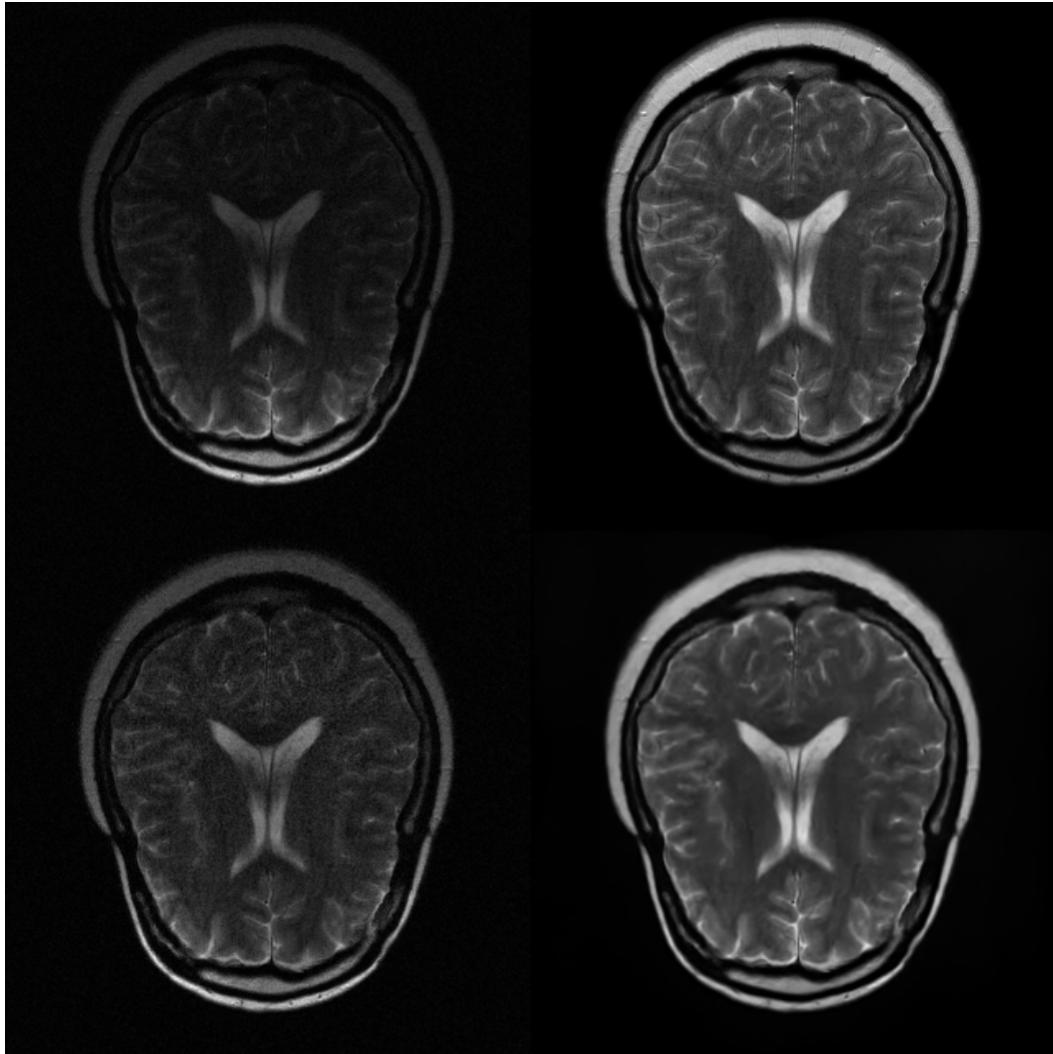


Figure 7: Example result from the test set. Top left: synthetic input, top right: ground truth, bottom left: N4ITK output, bottom right: U-Net prediction.

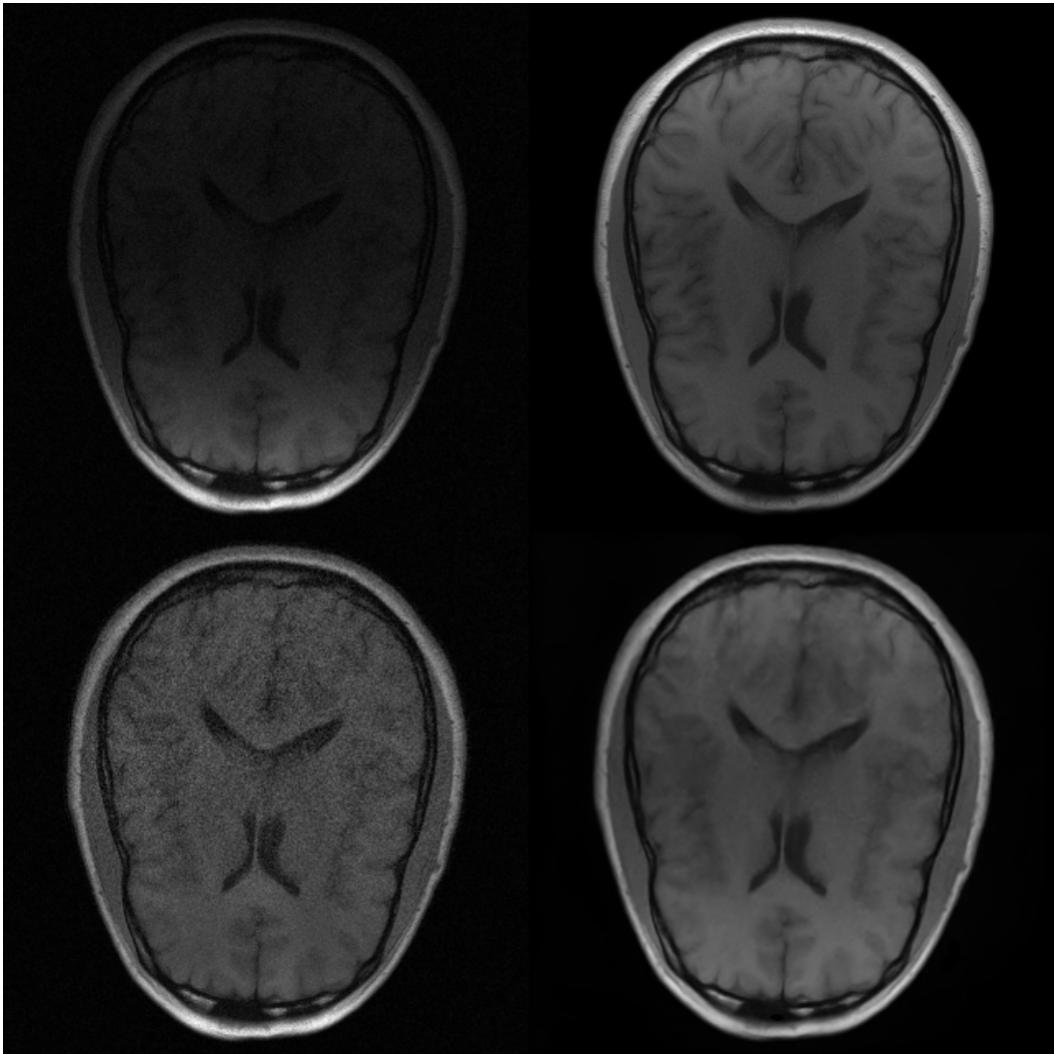


Figure 8: Example result from the test set. Top left: synthetic input, top right: ground truth, bottom left: N4ITK output, bottom right: U-Net prediction.

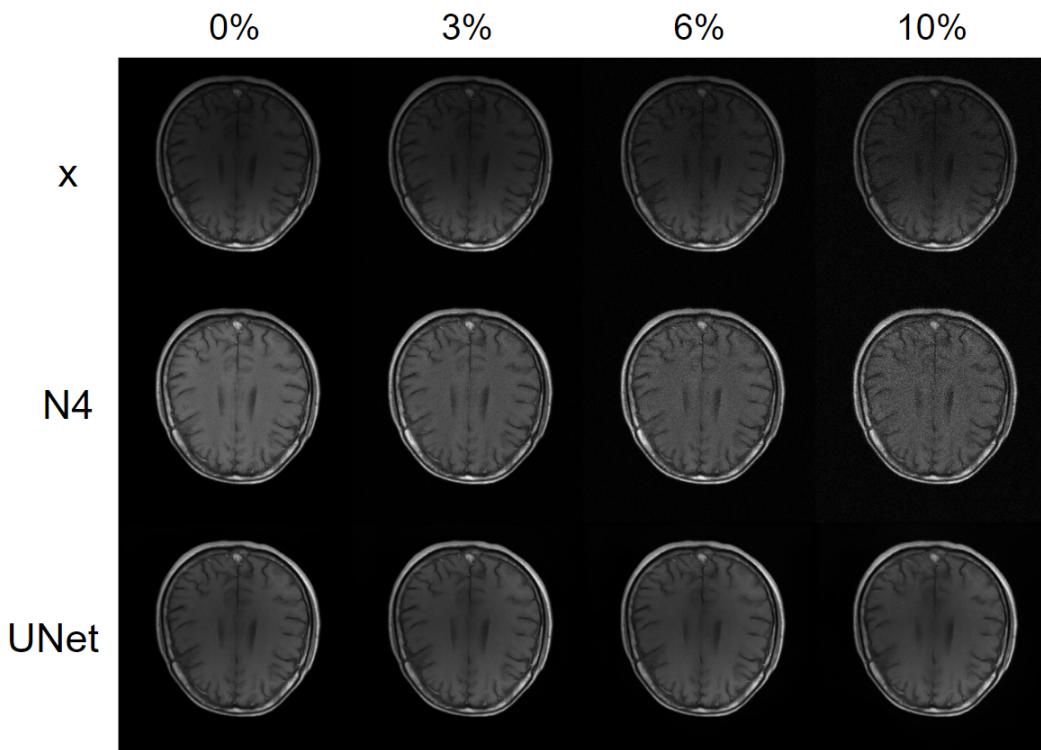


Figure 9: Results from a sweep through different input noise intensities.

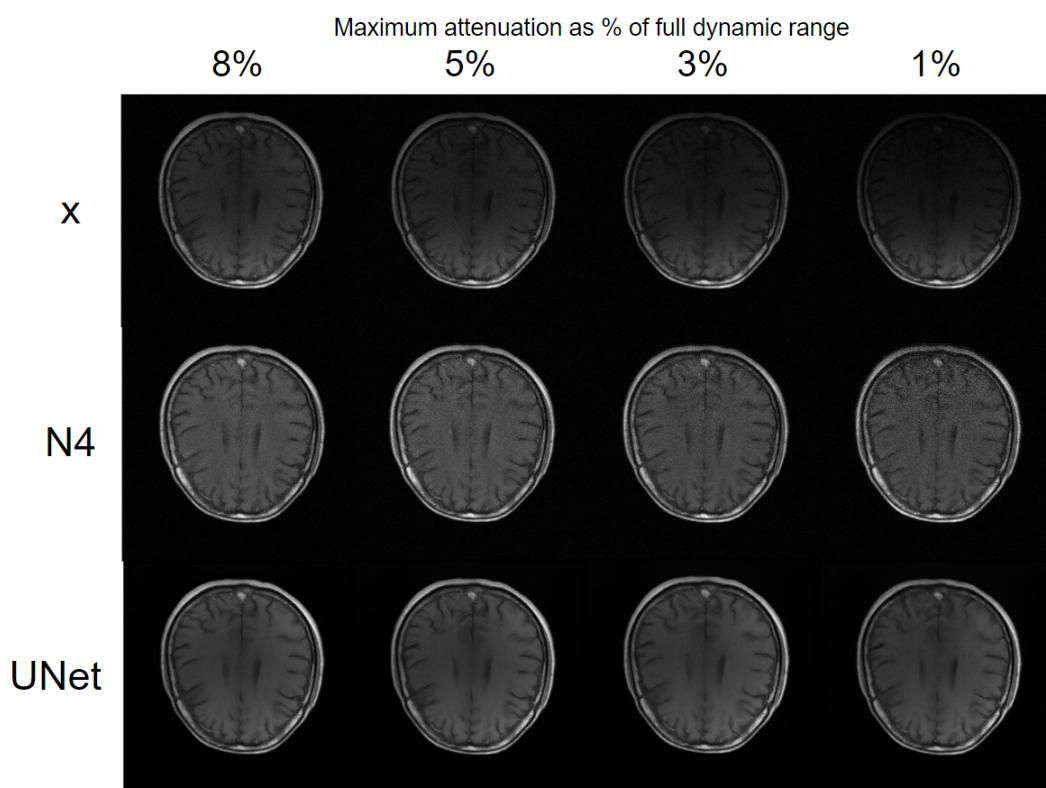


Figure 10: Results from a sweep through different input bias field intensities.