Optimizing data acquisition for deep learning in magnetic resonance imaging

Quinn Q. Torres* **and Marcus L. Wong***
Department of Mathematics, Manhattan College

Abstract. Magnetic Resonance Imaging (MRI) uses large magnets, radio waves, and a computer system to show detailed images inside a person's body. A downside of MRI is that it takes a long time to gather the data. To address the time required to collect the data, we explored three different acquisitions using half and a fourth of the data causing blur, aliasing, and a combination of both. The goal was to produce the best reconstructed image using a Convolution Neural Network (CNN). Our CNN reduced blurring from the blurred image and aliasing from the alias + blur image but we found the CNN is unable to remove aliasing from only aliased images. This shows the importance of data acquisition in reconstruction with neural networks.

Introduction

MRI is a useful tool in the medical field to look at tissues inside the body. However, a downside of MRI is that it is slow and therefore costly. Research is being conducted in order to optimize the speed of MRI. Our research explores how to optimize the speed by choosing less data to collect. Specifically, we divided the research into two parts, in one part we used $2\times$ accelerated data, and in the other part we used $4\times$ accelerated data. We used a convolutional neural network to reconstruct our accelerated images [1]. Our research will illustrate how the reconstructed images perform compared to each different type of data acquisition for $2\times$ and $4\times$ data accelerations in a similar way to previous work in compressed sensing [2].

Background

Fourier transform

The Fourier transform is a mathematical function that converts images into a representations using sines and cosines. Fourier transforms place an image in its frequency domain and display both low and high frequencies. We created our Fourier transforms in such a way where the low frequencies with the larger structural information of the image are closer to the center, and the higher frequencies that contain the smaller details of the image are towards the edge [3].

Masking, Aliasing and Blurring

Masking is an image processing tool to take specific data from the image by setting some pixel values to zero and other pixel values to 1. The pixel values of zero are chosen to not be taken for further processing while the pixel values of one are. Different masks end up in images being processed in different ways (Fig. 1). For example, applying a mask that only processes the

*Research mentored by Angel Pineda, Ph.D.

frequencies in the middle of the Fourier transform results in a blurred image, while a mask that takes every other column of the image, results in an aliased image. Aliasing is an effect that causes different signals to become indistinguishable from one another when sampled. In simpler terms, aliasing displays an image overlapping with itself.

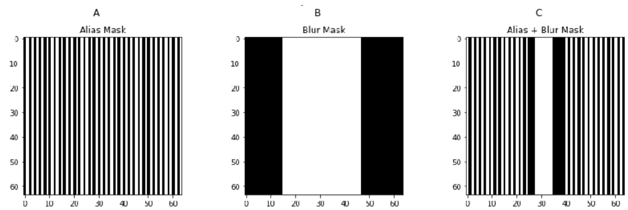

Figure 1. The masks we used for 2× accelerated data acquisition. Image A, every other line is collected; image B, the middle 32 columns is collected, and image C has every other line collected and the middle eight columns

Neural networks

A neural network can be thought of as a black box that takes in one or multiple inputs and then processes them into one or multiple outputs. The black box contains layers that consist of neurons. The goal of the neural network is to find the appropriate weighted connections from one layer to another [4, 5, 6]. One example of a neural network would be having the inputs of a neural network be aspects of a house like number of bedrooms and square footage, and the output be the price of the house based on those inputs. A diagram of a neural network structure can be seen in Fig. 2.

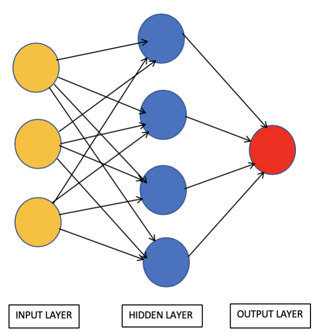

Figure 2. A neural network with three nodes as the input layer, four nodes as the hidden layer, and one node as the output layer. Each node passes along information to the nest layer that have corresponding weights associated with them.

Methods

We used three different data acquisitions, alias, blur, and alias + blur. The purpose of these different data acquisitions is to simulate how the image would look like if we only processed specific data from the Fourier transform of a 64×64 image. We used the data acquisitions for 2× and 4× accelerated data, which means that for two times accelerated data we acquired 32 columns,

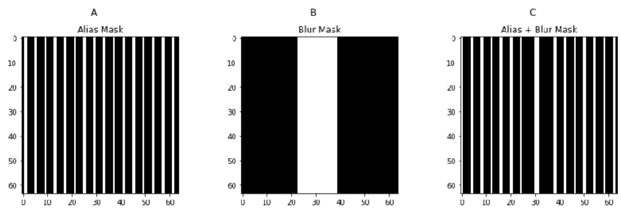

Figure 3. The masks we used for four times accelerated data acquisition. Image A is collecting every other two columns of data, image B is collecting the middle 16 columns of data, and image C is collecting every other 2 columns and the middle 3 columns. The areas that are black represent data not being collected, while the area that is white represents the data being collected.

while for four times accelerated we acquired 16 columns of the Fourier transform. Fig. 3 shows images on how the three different data acquisition masks looked like for 4× accelerated data.

Our data set consisted of 1000 64×64 sub images from an MRI volume. The order of the images was randomized; 800 images were used for training and 200 images were used for testing. We then took the inverse Fourier transform of the MRI data after the specific mask was applied and ran it through our convolutional neural network. A convolutional neural network is a type of neural network used in image recognition and processing. In addition, CNNs consist of convolutional layers that learn specific aspects of the image [4]. Figs. 4 give the structure and number of parameters of the convolutional neural network that was used in our work. For the purposes of our research we decided to keep our network simple.

Layer (type)	Output Shape	Param #
input_20 (InputLayer)	(None, 64, 64, 1)	0
conv2d_28 (Conv2D)	(None, 64, 64, 64)	640
batch_normalization_19 (Batc	(None, 64, 64, 64)	256
conv2d_29 (Conv2D)	(None, 64, 64, 64)	36928
batch_normalization_20 (Batc	(None, 64, 64, 64)	256
conv2d_30 (Conv2D)	(None, 64, 64, 64)	36928
batch_normalization_21 (Batc	(None, 64, 64, 64)	256
conv2d_31 (Conv2D)	(None, 64, 64, 64)	36928
batch_normalization_22 (Batc	(None, 64, 64, 64)	256
conv2d_32 (Conv2D)	(None, 64, 64, 1)	577

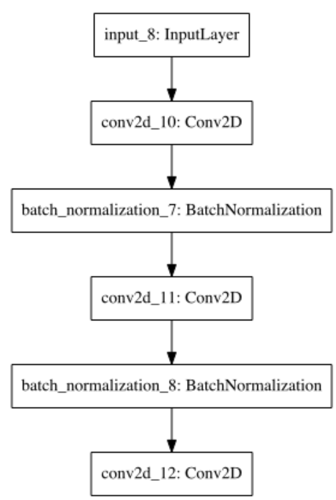

Figure 4. The table shows the progression of each layer in the convolutional neural network, the shape of the resulting output after each layer, and the number of parameters for each layer. The diagram shows the order of each layer in the convolutional neural network we used. It starts with the input layer which takes in the MRI images, then the convolutional layer which extracts features from our images, a batch normalization layer normalizes specified quantities of data to speed up the learning process and then repeats until it reaches the final layer, a convolutional layer, a batch normalization layer, and a convolutional layer for the final layer.

Results

2× Data Acceleration

Fig. 5 shows the results for the 2× aliased acquisition. The image reconstructed using the neural network is a blurred version of the aliased image. The 2× blurred acquisition results are shown in Fig. 6. The reconstructed image removes some of the blurring. In the aliased + blur image (Fig. 7), the reconstructed image removes some of the blurring and aliasing.

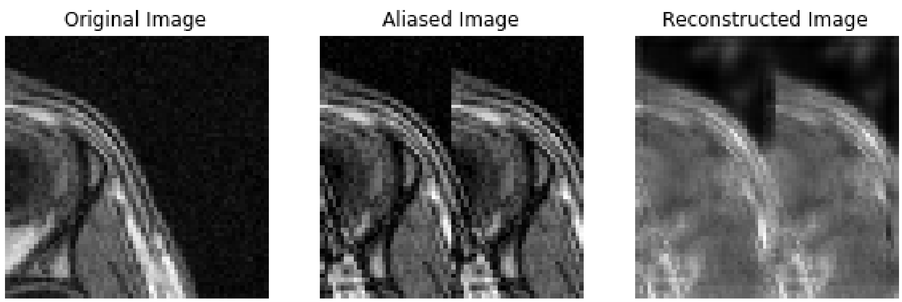

Figure 5. Original image, image from the 2× aliased acquisition and the reconstructed image using the neural network. The RMSE between original and aliased image was 0.101 and the RMSE between original and reconstructed image was 0.102. The reconstructed image blurred the aliased image but had similar RMSE for this specific image.

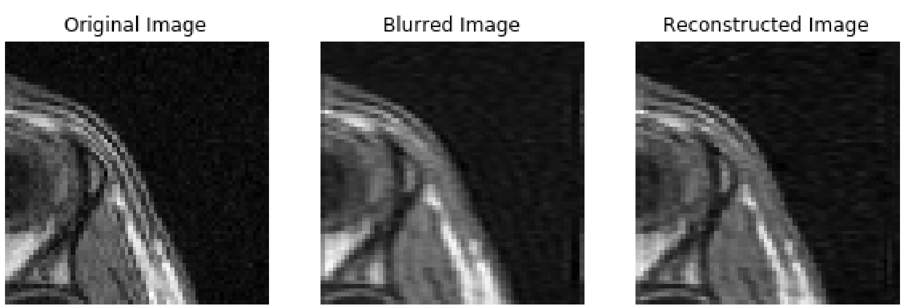

Figure 6. Original image, image from the 2× blurred acquisition and the reconstructed image using the neural network. The RMSE between original and blurred image was 0.025 and the RMSE between original and reconstructed image was 0.029. The reconstructed image removed some of the blur from the blurred image but had a worse RMSE for this specific image.

4× Data Acceleration

The results for a four times (4×) accelerated acquisition magnify the challenges of acquiring less data as seen in Figs. 8, 9, and 10.

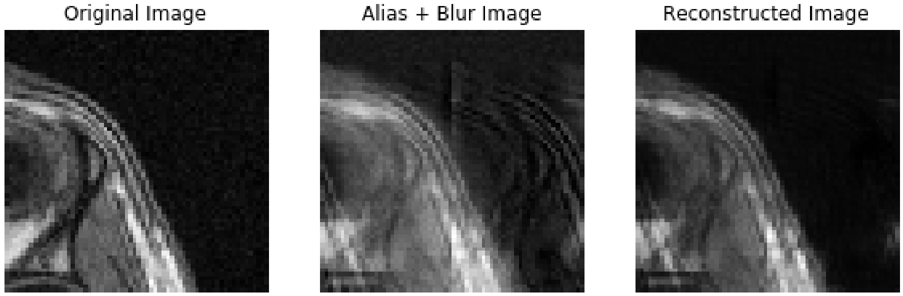

Figure 7. Original image, image from the 2× aliased and blurred acquisition and the reconstructed image using the neural network. The RMSE between original and alias + blur image was 0.041 and the RMSE between original and reconstructed image was 0.037. The reconstructed image removed some of the blur and aliasing from the blurred and aliased image and had a better RMSE for this specific image.

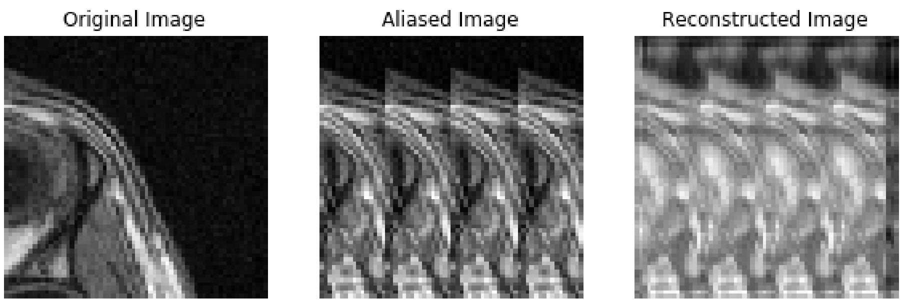

Figure 8. Original image, image from the 4× aliased acquisition, and the reconstructed image using the neural network. The RMSE between original and aliased image was 0.112 and the RMSE between original and reconstructed image was 0.134. The reconstructed image blurred the aliased image and had a larger RMSE for this specific image.

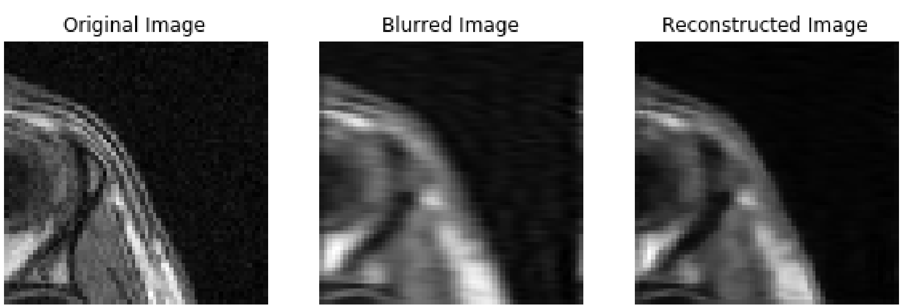

Figure 9. Original image, image from the 4× blurred acquisition, and the reconstructed image using the neural network. The RMSE between original and blurred image was 0.040 and the RMSE between original and reconstructed image was 0.045. The reconstructed image removed some of the blur from the blurred image but had a worse RMSE for this specific image.

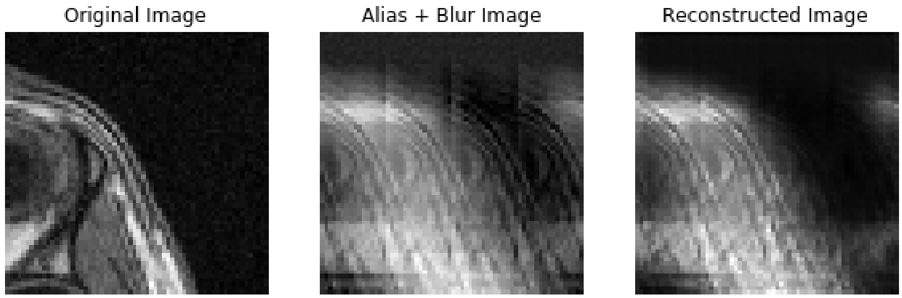

Figure 10. Original image, image from the 4× aliased, and blurred acquisition and the reconstructed image using the neural network. The RMSE between original and alias + blur image was 0.066 and the RMSE between original and reconstructed image was 0.063. The reconstructed image removed some of the blur and aliasing from the blurred and aliased image and had a slightly better RMSE for this specific image. There were significantly more artifacts in this image than in the 2× acceleration.

Average mean squared error

To better assess the performance of our methods, we computed the average mean squared error (MSE) using our testing set for each of the acquistions (Table 1). We see that the 2× error is consistently less than the 4× error. The acquisition with the smallest MSE was the 2× blur and the one with the largest MSE was the 4× alias.

Table 1. The average mean squared error results for the three different data acquisitions for 2× and 4× accelerated data.

	2×	4×
Alias	8.5×10^{-3}	1.0×10^{-2}
Blur	5.0×10^{-4}	1.2×10^{-3}
Alias + Blur	1.7×10^{-3}	4.6×10^{-3}

Training performance of CNN

Fig. 11 shows the root mean square error (RMSE) for the training of the CNN for the 2× alias and blur acquisition as a funtion of the number of passes through the data (epochs). This plot is representative of the training of the neural network for other acquisitions. We used 50 epochs for all of our results.

Discussion

Based on Figs. 5 and 8, we can see that our reconstructed image does not significantly reduce the aliasing from a purely aliased acquisition. Based on the images from Figs. 6 and 9, our CNN reduces some blurring from the purely blurred acquisitions. Figs. 7 and 10 show that our CNN reduces aliasing and blur from the aliased and blurred acquisition. These images suggest that the

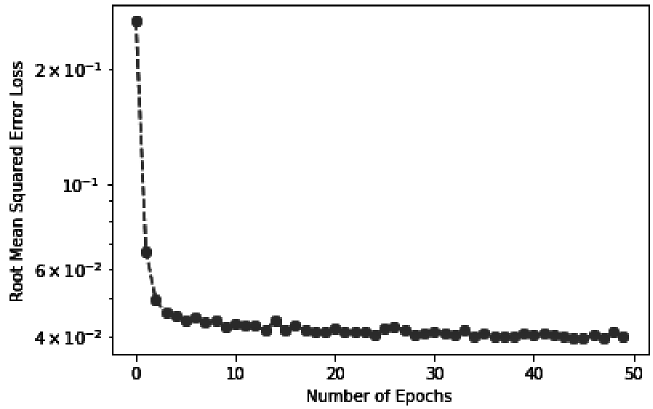

Figure 11. A graph of the training performance of the convolutional neural network with the number of epochs on the x-axis and the average root mean squared error on the y-axis. As the number of epochs increases, the average root mean squared error decreases and plateaus around 45-50 epochs.

neural network works better when low frequencies (large scale features) are included in the data set.

Table 1 shows that for $2\times$ and $4\times$ data acceleration, the lowest mean square error was for the blurred images. However, MSE may not be the best metric for image comparison and for future work we would like to look into other metrics like structural similarity (SSIM) [7]. Ultimately, we would like to explore the performance of these methods on clinical tasks [8]. In addition, we would also like to test different data acquisitions like spokes, spirals, and more complex acquisitions [2].

Conclusion

We found that the reconstructed image decreases blurring from the blurred image and aliasing from the alias + blur image. In addition, we found the reconstructed image is unable to remove aliasing from the purely aliased data. In this work we showed the importance of data acquisition in the performance of neural network reconstructions.

Acknowledgements

This project was supported by the School of Science Summer Research Scholar Program. We thank our advisor Dr. Pineda for his mentoring and Brother Daniel Gardner for coordinating the summer scholars program ctivities.

References

[1] Wong ML, Torres QQ, Optimizing Network Architecture for Deep Learning in Magnetic Resonance Imaging (MRI), *The Manhattan Scientist*, 2019, 6, .

[2] Ortega E, Saenz RV, Importance of Sampling Pattern and Regularization in Under-Sampled Magnetic Resonance Imaging (MRI), *Dimensions*, 2015, 101-112.

[3] Gonzalez RC, Woods RE, Digital Image Processing 4ed, 2017, Pearson, Hoboken, NJ.

[4] Hyun CM, Kim HP, Lee SM, Lee S, Seo JK, Deep Learning for Undersampled MRI Reconstruction, *Physics in Medicine and Biology*, 2018, 63 (13), 135007.

[5] Ronneberger O, Fischer P, Brox T, U-Net: Convolutional Networks for Biomedical Image Segmentation, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Springer, LNCS, 2015, 9351, 234–241.

[6] Knoll F, Hammernik K, Kobler E, Pock T, Recht MP, Sodickson DK, Assessment of the Generalization of Learned Image Reconstruction and the Potential for Transfer Learning, *Magnetic Resonance in Medicine*, 2018, 81, 116–128.

[7] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP, Image Quality Assessment: From Error Visibility to Structural Similarity, *IEEE Transactions on Image Processing* , 2004, 13 (4), 600-612.

[8] Miedema H, Classification of Magnetic Resonance Imaging (MRI) Data Using Small Sample Sizes, *The Manhattan Scientist*, 2017, 4, 235-242.