DEEP LEARNING FOR ACCELERATED MR IMAGING

A DISSERTATION SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

By
Salman Ul Hassan Dar
February 2021

Deep Learning for Accelerated MR Imaging
By Salman Ul Hassan Dar
February 2021

We certify that we have read this dissertation and that in our opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Tolga Çukur(Advisor)

Emine Ülkü Sarıtaş Çukur

İbrahim Aykut Erdem

İlkay Öksüz

Ayşegül Dündar

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

DEEP LEARNING FOR ACCELERATED MR IMAGING

Salman Ul Hassan Dar
Ph.D. in Electrical and Electronics Engineering
Advisor: Tolga Çukur
February 2021

Magnetic resonance imaging is a non-invasive imaging modality that enables multi-contrast acquisition of an underlying anatomy, thereby supplementing multitude of information for diagnosis. However, prolonged scan duration may prohibit its practical use. Two mainstream frameworks for accelerating MR image acquisitions are reconstruction and synthesis. In reconstruction, acquisitions are accelerated by undersampling in k-space, followed by reconstruction algorithms. Lately deep neural networks have offered significant improvements over traditional methods in MR image reconstruction. However, deep neural networks rely heavily on availability of large datasets which might not be readily available for some applications. Furthermore, a caveat of the reconstruction framework in general is that the performance naturally starts degrading towards higher acceleration factors where fewer data samples are acquired. In the alternative synthesis framework, acquisitions are accelerated by acquiring a subset of desired contrasts, and recovering the missing ones from the acquired ones. Current synthesis methods are primarily based on deep neural networks, which are trained to minimize mean square or absolute loss functions. This can bring about loss of intermediate-to-high spatial frequency content in the recovered images. Furthermore, the synthesis performance in general relies on similarity in relaxation parameters between source and target contrasts, and large dissimilarities can lead to artifactual synthesis or loss of features. Here, we tackle issues associated with reconstruction and synthesis approaches. In reconstruction, the data scarcity issue is addressed by pre-training a network on large readily available datasets, and fine-tuning on just a few samples from target datasets. In synthesis, the loss of intermediate-to-high spatial frequency is catered for by adding adversarial and high-level perceptual losses on top of traditional mean absolute error. Finally, a joint reconstruction and synthesis approach is proposed to mitigate the issues associated with both reconstruction and synthesis approaches in general. Demonstrations on MRI brain datasets of healthy subjects and patients indicate

superior performance of the proposed techniques over the current state-of-the art ones.

Keywords: Magnetic resonance imaging, reconstruction, synthesis, deep neural networks, transfer learning, generative adversarial networks.

ÖZET

TÜRKÇE BAŞLIK

Salman Ul Hassan Dar
Elektrik Elektronik Mühendisliği, Doktora
Tez Danışmanı: Tolga Çukur
Şubat 2021

Manyetik rezonans görüntüleme, aynı anatominin çoklu kontrastlı edinimine olanak sağlayan ve böylece tanı için çok sayıda bilgiyi sağlayan noninvazif bir görüntüleme yöntemidir. Ancak uzun tarama süresi, pratik kullanıma engel olabilir. MR görüntü edinimlerini hızlandırmak için kullanılan iki ana sistem geriçatım ve sentezdir. Geriçatımda, edinimler k-uzayında alt örnekleme ve takibinde geriçatım algoritmaları ile hızlandırılır. Son zamanlarda derin sinir ağları, MR görüntü geriçatımında geleneksel yöntemlerin üstüne önemli gelişmeler sağladı. Bununla birlikte, derin sinir ağları geniş veri setlerinin kullanılabilirliğine bel bağlamaktadır ki bu veri setleri bazı uygulamalar için mevcut olmayabilir. Ayrıca, genel olarak geriçatım yöntemlerinin bir uyarısı şudur ki, daha az veri örnekleminin elde edildiği daha yüksek hızlandırma faktörlerine doğru performans doğal olarak düşmeye başlamaktadır. Alternatif sentez yönteminde, istenilen kontrastların bir alt kümesinin edinimi ve eksik olanların elde edilenlerden kurtarılmasıyla edinimler hızlandırılır. Mevcut sentez yöntemleri, başlıca ortalama kare ve mutlak kayıp fonksiyonlarını en aza indirmek için eğitilmiş derin sinir ağlarına dayanmaktadır. Bu kurtarılan görüntülerde, orta ve yüksek frekans içeriğinin kaybına neden olabilir. Ayrıca sentez performansı genel olarak kaynak ve hedef kontrastları arasındaki relaksasyon parametrelerindeki benzerliğe dayanır ve büyük farklılıklar yapay sentezlere ve özellik kayıplarına yol açabilir. Burada geriçatım ve sentez yaklaşımları ile alakalı sorunları ele alıyoruz. Geriçatımda veri azlığı sorunu, mevcut geniş veri setleri üzerinde bir modelin ön eğitiminin yapılması ve sonrasında hedef veri setinden yalnızca birkaç örnek üzerinde ince ayarlarının yapılmasıyla ele alınmıştır. Sentez durumunda orta ve yüksek frekans kaybı, geleneksel ortalama mutlak hatanın üzerine çekişmeli ve yüksek düzeyde algısal kayıp fonksiyonlarının eklenmesiyle karşılanmaktadır. Son olarak, hem geriçatım hem de sentez yaklaşımlarıyla alakalı sorunları hafifletmek için ortak bir geriçatım ve sentez yaklaşımı önerilmiştir. Sağlıklı deneklerin ve hastaların MR beyin verisetleri üzerinde yapılan gösterimler, önerilen tekniklerin güncel

gelişmiş tekniklere göre üstün performansını göstermektedir.

Anahtar sözcükler: Manyetik rezonans görüntüleme, geriçatım, sentez, derin sinir ağları, öğrenme aktarması, çekişmeli üretici ağlar.

Acknowledgement

First of all I would like to thank Allah for the countless blessings.

Then I would like to thank my supervisor Tolga Çukur for his constant support. I don't think I could have had a better PhD supervisor than him. I have learned a lot from him and I am still learning. Whatever I am today as a researcher is because of the time and effort he has put into me. He has always been supportive and very understanding whenever the time came. I hope I can pay him back sometime.

I would also like to thank my jury members, Emine hoca for her time despite her very busy schedule and a baby which can be very difficult to manage, Aykut hoca for his time and his constant guidance in multiple projects, İlkay hoca for agreeing to attend my thesis presentation on a 2-weeks notice although it was not his preferred time, and Ayşegül hoca for being super nice by helping me in getting her signatures while being out of the town.

Next, I would like to thank all my current and old labmates, most of whom are also my really good friends. I would like to thank Emin (one of my best friends), Umit abi (like a big brother and a guide), Osman, Mahmut (like an annoying little brother), Ozgur aga, Muzaffer (especially for the Turkish abstract), Emrullah, Shahdloo, Efe, Ibrahim, and Kubra. I would also like to thank the rest of the UMRAM family members who have made this place wonderful and who have always been nice and helpful throughout. I want to thank Bilal and Mustafa who are some of the nicest people I have come across at Bilkent, Uğur abi, Alper, Cemre, Omer, Toygan, Koray, Ali-reza, Ehsan khan, Aydan hanım, and Ergin hoca.

I have spent around 8 years in Bilkent and throughout this time I have made many friends who have made Bilkent a fun place. I want to thank my earlier dorm mates including Adamu, Muazzam, M. Yunusa, Abubakr Adamu, Redi, Erion, Abdullah Oner, Mustafa Erol and Mehmet Bey. I would also like to thank the Arab gang including Abdur-rehman, Tariq, Ahmed Ouf, Nabeel and Maha. I would also like to thank Muhammad Alser and Nur for being like elder siblings, and their beautiful children Hassan and Adam. Pakistani friends, Anjum bhai and Faiza baji always there for any kind of help, Muhadissa for being super

helpful in my first 2 years, Salahuddin bhai, Umer bhai, Mansoor bhai and Maria baji, Naveed bhai and Bareera, Talha, Umer bhai. All undergrad football friends, Haider, Salu, Shah, Qidwai, Ahad.

Next, I would like to thank my family for their support especially during last year of my PhD. Aji mama, Tai, and Chachi (who have treated my like their own son), Mushtaq uncle, Chacho, Papa, Ghazi bhai, Bawa, Maryam, Nida baji, Samar baji, Motu, Hamza, Atif bhai, and my parents in-law for trusting me with their beautiful daughter and always treating me like their fifth child. I would like to thank Hamza bhai and Maya api who were always the people I looked upto and I always wanted to do as good as them. My brother Shanoo who has been one of my best friends since my childhood and I have spent one of the best times of my life with him.

Then I would like to thank my wonderful and lovely wife Elif, who is the biggest blessing in my life. She has been very very supportive and compromising throughout my PhD, letting me work late nights, weekends and even on vacations (sorry canım), and she has always believed in me. To be honest, I don't thing I deserve her. I would be totally lost without her. She has also provided me with a beautiful and fun daughter Vera, who keeps on adding new things to our lives. I wish I could make Vera understand how she has been a shining light in this past year, which has been the toughest year of my life.

I would like to thank my parents for all they have done for me. I cannot repay even a small percentage of what they have done for me. I started realizing these things more after having my own daughter. Special thanks to my father who taught me how to work hard and with dedication.

I wish my mother were alive to see me finish my PhD. It was her biggest dream. She used to make a lot of Duas for me, before every presentation, deadline, and exam. I still remember she started crying with happiness when I defended my master thesis, and finishing my PhD without her feels incomplete, as if I am graduating without fulfilling all the requirements. She taught me how to read and write. She would carry me all night (I wasn't an easy baby). She would wait for me whenever I would go somewhere, and wouldn't sleep till I return. She would get worried whenever I wouldn't pick her call up. She would wake up at nights, and pray for me and everyone. She taught me how to forgive. I learned

humility from her. She taught me how to be a giver. She taught me how to be selfless. She was a perfect example of a walking angel on earth. She is the best human I have ever known in my life. Whatever good I have is from her, and whatever bad I have is what I couldn't learn from her. I dedicate this thesis to her.

Contents

1 Introduction **1**

2 Fundamentals of Accelerated MRI **4**

2.1 Magnetic resonance imaging (MRI) 4

2.2 Accelerated MRI 6

2.2.1 Reconstruction 7

2.2.2 Synthesis 8

3 A TransferLearning Approach for Accelerated MRI Using Deep Neural Networks **10**

3.1 Introduction 11

3.2 Methods 14

3.2.1 Multi-Coil Data 16

3.2.2 MRI Reconstruction via Neural Networks (NN) 17

3.2.3 Datasets 21

3.2.4 Network training and fine-tuning 26

3.2.5 Network validation 29

3.2.6 Performance analyses 32

3.3 Results 35

3.3.1 Single-coil magnitude data 35

3.3.2 Multi-coil complex data 50

3.3.3 Single-coil complex data 60

3.4 Discussion 61

4 Image Synthesis in Multi-Contrast MRI With Conditional Generative Adversarial Networks **74**

4.1 Introduction 74
4.2 Methods 79
4.2.1 Image Synthesis via Adversarial Networks 79
4.2.2 MRI Datasets 82
4.2.3 Image Registration 85
4.2.4 Network Training 85
4.2.5 Competing Methods 87
4.2.6 Experiments 88
4.3 Results 93
4.3.1 Comparison of GAN-based models 93
4.3.2 Comparison to state-of-the-art methods 96
4.3.3 Spectral density analysis 103
4.3.4 Generalizability 104
4.3.5 Reliability against noise 105
4.4 Discussion 106
4.5 Conclusion 111

5 Prior-Guided Image Reconstruction for Accelerated Multi-Contrast MRI via Generative Adversarial Networks 120
5.1 Introduction 121
5.2 Theory and Methods 125
5.2.1 Accelerated MRI 125
5.2.2 Joint Reconstruction-Synthesis via Conditional GANs . . . 128
5.2.3 Competing Methods 132
5.2.4 MRI Datasets 133
5.2.5 Image Registration 135
5.2.6 Undersampling Patterns 136
5.2.7 Model Training Procedures 136
5.2.8 Experiments 139
5.3 Results 144
5.3.1 Main Experiments 144
5.3.2 Control Experiments 147
5.4 Discussion 149

5.5 Conclusion 153

6 Conclusion **173**

List of Figures

2.1 Sample T_1- and T_2-weighted images of the human brain. 6
2.2 In MR image reconstruction, acquisitions are undersampled in k-space followed by a tailored reconstruction algorithm. 8

3.1 Proposed transfer-learning approach for NN-based reconstructions of multi-coil (N_c coils) undersampled acquisitions. A deep architecture with multiple subnetworks is used. The subnetworks consist of calibration consistency CC and convolutional neural network CNN blocks, each followed by a data consistency block DC. (a) Each CNN block is trained sequentially to reconstruct synthetic multi-coil natural images from ImageNet, given zero-filled Fourier reconstructions of their undersampled versions. Due to differences in the characteristics of natural and MR images, the ImageNet-trained network will yield suboptimal performance when directly tested on MR images. (b) For domain transfer, the ImageNet-trained network is fine-tuned end-to-end in the testing domain using few tens of images. This approach enables successful domain transfer between natural and MR images. 13

3.2 Demonstration of (a) convolutional neural network (CNN), (b) calibration consistency (CC) and (c) data consistency (DC) blocks given a multi-coil image x as an input. (a) The CNN block first combines undersampled multi-coil images using coil-sensitivity maps A, estimated via ESPIRiT. Real and imaginary parts of the coil-combined image are then reconstructed using two separate networks. The outputs of the real and imaginary networks are joined to form a complex image, which is then back projected onto individual coils again using the coil sensitivity maps. (b) The CC block transforms the input image into Fourier domain, applies the interpolation operator on multi-coil k-space data, and converts the image back into image domain. (c) The DC block performs a weighted combination of samples recovered by the previous block (CNN or CC) and the originally-acquired samples. 15

3.3 Representative synthetic complex multi-coil natural images. Complex multi-coil natural images were simulated from magnitude images in ImageNet (see Methods for details). Magnitude and phase of two simulated multi-coil natural images (a and b) are shown along with their reference magnitude images. 24

3.4 Percentage change in validation error as a function of number of epochs for T_2 to T_1 domain transfer at acceleration factor R=4. Results are shown for sequential training of individual CNN blocks (a-e), end-to-end training of the complete network (f) and fine-tuning of the complete network (g). Initial CNN block training was performed on 2000 T_2-weighted images, end-to-end training was performed on 100 T_2-weighted images, and fine-tuning was performed on 20 T_1-weighted images. 30

3.5 Percentage change in validation error as a function of number of epochs for ImageNet to T_1 domain transfer at acceleration factor R=4. Results are shown for learning rates (lr) equal to (a) 10^{-4}, (b) 10^{-5} and (c) 10^{-6}. Initial CNN block training was performed on 2000 ImageNet images, end-to-end training was performed on 100 ImageNet images, and fine-tuning was performed on 20 T_1-weighted images. Learning rate equal to 10^{-5} facilitates convergence while preventing undesirable oscillations in the validation error. A learning rate of 10^{-5} ensures both stable fine-tuning and faster convergence. In contrast, a higher learning rate of 10^{-4} leads to oscillatory behavior in validation error, potentially suggesting overfitting to fine-tuning samples. While fine-tuning is relatively stable for a lower learning rate of 10^{-6}, network convergence is noticeably slower. 31

3.6 Representative reconstructions of a T_1-weighted acquisition at acceleration factor R=4. Reconstructions were performed via the Zero-filled Fourier method (ZF), and ImageNet-trained, T_2-trained, and T_1-trained networks. (a) Reconstructed images and error maps for raw networks (see colorbar). (b) Reconstructed images and error maps for fine-tuned networks. The fully-sampled reference image is also shown. Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_1-weighted images. Following fine-tuning with few tens of samples, ImageNet-trained and T_2-trained networks yield reconstructions of highly similar quality to the T_1-trained network. . . 36

3.7 Reconstructions of a T_1-weighted acquisition with R=4 via ZF, conventional compressed-sensing (CS), and ImageNet-trained, T_1-trained and T_2-trained networks along with the fully-sampled reference image. Error maps for each reconstruction are shown below (see colorbar). Networks were trained on 2000 images and fine-tuned on 20 images acquired with the test contrast. The domain-transferred networks maintain nearly identical performance to the networks trained directly in the testing domain. Furthermore, the domain-transferred networks reconstructions outperform conventional CS in terms of image sharpness and residual aliasing artifacts. 37

3.8 Reconstruction performance was evaluated for undersampled T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained on 4k images and fine-tuned on 100 images), ImageNet-trained networks (trained on 500, 1000, 2000, or 4000 images), and T_2-trained network (trained on 4000 images). Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms all domain-transferred networks. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. Domain-transferred networks trained on fewer samples require more fine-tuning samples to yield similar performance consistently across R. 38

3.9 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil magnitude images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained networks trained on (a) 500, (b) 1000, (c) 2000, and (d) 4000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network. Domain-transferred networks trained on fewer samples require more fine-tuning samples for the PSNR values to converge. Furthermore, at higher values of R, more fine-tuning samples are required for convergence. 39

3.10 Representative reconstructions of a T_2-weighted acquisition at acceleration factor R=4. Reconstructions were performed via the Zero-filled Fourier method (ZF), and ImageNet-trained, T_2-trained, and T_1-trained networks. (a) Reconstructed images and error maps for raw networks (see colorbar). (b) Reconstructed images and error maps for fine-tuned networks. The fully-sampled reference image is also shown. Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_2-weighted images. Following fine-tuning with few tens of samples, ImageNet-trained and T_1-trained networks yield reconstructions of highly similar quality to the T_2-trained network. . . 42

3.11 Reconstructions of a T_2-weighted acquisition with R=4 via ZF, conventional compressed-sensing (CS), and ImageNet-trained, T_1-trained and T_2-trained networks along with the fully-sampled reference image. Error maps for each reconstruction are shown below (see colorbar). Networks were trained on 2000 images and fine-tuned on 20 images acquired with the test contrast. The domain-transferred networks maintain nearly identical performance to the networks trained directly in the testing domain. Furthermore, the domain-transferred networks reconstructions outperform conventional CS in terms of image sharpness and residual aliasing artifacts. 43

3.12 Reconstruction performance was evaluated for undersampled T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained on 4k images and fine-tuned on 100 images), ImageNet-trained networks (trained on 500, 1000, 2000, or 4000 images), and T_1-trained network (trained on 4000 images). Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. Domain-transferred networks trained on fewer samples require more fine-tuning samples to yield similar performance consistently across R. 44

3.13 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained networks trained on (a) 500, (b) 1000, (c) 2000, and (d) 4000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network. Domain-transferred networks trained on fewer samples require more fine-tuning samples for the PSNR values to converge. Furthermore, at higher values of R, more fine-tuning samples are required for convergence. 45

3.14 Reconstruction performance was evaluated for undersampled multi-coil T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. 47

3.15 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on multi-coil complex images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network (see Figure 6). At higher values of R, more fine-tuning samples are required for convergence. 48

3.16 Representative reconstructions of a multi-coil T_1-weighted acquisition at acceleration factor R=10. Reconstructions were performed via ZF, ImageNet-trained and T_1-trained networks, and SPIRiT (top row). Corresponding error maps are also shown (see colorbar; bottom row) along with the fully-sampled reference (top row). Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_1-weighted images. The ImageNet-trained network maintains similar performance to the T_1-trained network trained directly on the images from the test domain. Furthermore, the domain-transferred network outperforms conventional SPIRiT in terms of residual aliasing artifacts. 49

3.17 Reconstruction performance was evaluated for undersampled multi-coil T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_2-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. 51

3.18 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on multi-coil complex images) to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network. At higher values of R, more fine-tuning samples are required for convergence. 52

3.19 Representative reconstructions of a multi-coil T_2-weighted acquisition at acceleration factor R=10. Reconstructions were performed via ZF, ImageNet-trained and T_2-trained networks, and SPIRiT (top row). Corresponding error maps are also shown (see colorbar; bottom row) along with the fully-sampled reference (top row). Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_2-weighted images. The ImageNet-trained network maintains similar performance to the T_2-trained network trained directly on the images from the test domain. Furthermore, the domain-transferred network outperforms conventional SPIRiT in terms of residual aliasing artifacts. 53

3.20 Percentage change in network weights as a function of network depth for multi-coil ImageNet to (a) T_1 and (b) T_2 domain transfer averaged across acceleration factors (R=4-10). Red dots correspond to the percentage change, and blue dashed lines correspond to a linear least squares fit to the percentage change. Overall, the percentage change in weights is higher for earlier versus later layers of the network. For ImageNet to T_1 domain transfer, percentage change varies from 2.27% to 0.56%, and for ImageNet to T_2 domain transfer percentage change varies from 3.28% to 0.47%. Note that the layer number ranges from 1 to 25. This is because the CNN architecture used in this study consists of 5 cascades and each cascade consists of 5 layers. 54

3.21 Reconstruction performance was evaluated for undersampled single-coil complex T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. 56

3.22 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil complex images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network. 57

3.23 Reconstruction performance was evaluated for undersampled single-coil complex T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_2-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. 58

3.24 Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil complex images) to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network. 59

4.1 The pGAN method is based on a conditional adversarial network with a generator G, a pre-trained VGG16 network V, and a discriminator D. Given an input image in a source contrast (e.g., T_1-weighted), G learns to generate the image of the same anatomy in a target contrast (e.g., T_2-weighted). Meanwhile, D learns to discriminate between synthetic (e.g., T_1-G(T_1)) and real (e.g., T_1-T_2) pairs of multi-contrast images. Both subnetworks are trained simultaneously, where G aims to minimize a pixel-wise, a perceptual and an adversarial loss function, and D tries to maximize the adversarial loss function. 77

4.2 The cGAN method is based on a conditional adversarial network with two generators (G_{T_1}, G_{T_2}) and two discriminators (D_{T_1}, D_{T_2}). Given a T_1-weighted image, G_{T_2} learns to generate the respective T_2-weighted image of the same anatomy that is indiscriminable from real T_2-weighted images of other anatomies, whereas D_{T_2} learns to discriminate between synthetic and real T_2-weighted images. Similarly, G_{T_1} learns to generate realistic a T_1-weighted image of an anatomy given the respective T_2-weighted image, whereas D_{T_1} learns to discriminate between synthetic and real T_1-weighted images. Since the discriminators do not compare target images of the same anatomy, a pixel-wise loss cannot be used. Instead, a cycle-consistency loss is utilized to ensure that the trained generators enable reliable recovery of the source image from the generated target image. 78

4.3 The proposed approach was demonstrated for synthesis of T_2-weighted images from T_1-weighted images in the MIDAS dataset. Synthesis was performed with pGAN, cGAN trained on registered images (cGANreg), and cGAN trained on unregistered images ($cGAN_{unreg}$). For pGAN and cGANreg, training was performed using T_2-weighted images registered onto T_1-weighted images ($T_1 \rightarrow T_2\#$). Synthesis results for (a) the single cross-section, and (b) multi cross-section models are shown along with the true target image (reference) and the source image (source). Zoomed-in portions of the images are also displayed. While both pGAN and cGAN yield synthetic images of striking visual similarity to the reference, pGAN is the top performer. Synthesis quality is improved as information across neighboring cross sections is incorporated, particularly for the pGAN method. 91

4.4 The proposed approach was demonstrated for synthesis of T_1-weighted images from T_2-weighted images in the MIDAS dataset. Synthesis was performed with pGAN, cGAN trained on registered images (cGANreg), and cGAN trained on unregistered images ($cGAN_{unreg}$). For pGAN and cGANreg, training was performed using T_2-weighted images registered onto T_1-weighted images ($T_2\# \rightarrow T_1$). Synthesis results for (a) the single cross-section, and (b) three cross-section models are shown along with the true target image (reference) and the source image (source). Zoomed-in portions of the images are also displayed. While both pGAN and cGAN yield synthetic images of striking visual similarity to the reference, pGAN is the top performer. Synthesis quality is improved as information across neighboring cross sections is incorporated for the pGAN method. 92

4.5 The proposed approach was demonstrated for synthesis of T_2-weighted images and T_1-weighted images in the MIDAS dataset. $T_1 \rightarrow T_2$# and T_2# $\rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2$#, and (b) T_2# $\rightarrow T_1$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions near issue boundaries that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring. 94

4.6 The proposed approach was demonstrated for synthesis of T_2-weighted and T_1-weighted images in the all datasets. $T_1 \rightarrow T_2$ and $T_2 \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_2 \rightarrow T_1$ are shown along with the true target image (reference) and the source image (source) for each dataset. The proposed method outperforms competing methods in terms of synthesis quality. The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in synthesized images compared to Multimodal and Replica that show some degree of blurring. 95

4.7 The proposed approach was demonstrated for synthesis of T_2-weighted and T_1-weighted images. $T_1 \rightarrow T_2$ and $T_2 \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_2 \rightarrow T_1$ are shown along with the true target image (reference) and the source image (source) for each dataset. The proposed method outperforms competing methods in terms of synthesis quality. The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in synthesized images compared to Multimodal and Replica that show some degree of blurring. 97

4.8 The proposed approach was demonstrated for synthesis of T_1-weighted images from T_2-weighted images in the IXI dataset. $T_2 \rightarrow T_1\#$ and $T_2\# \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_2 \rightarrow T_1\#$, and (b) $T_2\# \rightarrow T_1$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring. 98

4.9 The proposed approach was demonstrated for synthesis of T_2-weighted images from T_1-weighted images in the IXI dataset. $T_1\# \rightarrow T_2$ and $T_1 \rightarrow T_2\#$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1\# \rightarrow T_2$, and (b) $T_1 \rightarrow T_2\#$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions near issue boundaries that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring. 100

4.10 The proposed approach was demonstrated on glioma patients for synthesis of T_2-weighted images from T_1-weighted images, and T_2-weighted images from T_1-weighted images in the BRATS dataset. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_1 \rightarrow T_2$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). Regions of inaccurate synthesis with Replica and Multimodal are observed near pathologies (marked with arrows). Meanwhile, the pGAN method enables reliable synthesis with visibly improved depiction of intermediate spatial frequency information. 101

4.11 The T_1-weighted image of a sample cross-section from the MIDAS dataset was processed with an ideal filter in k-space. The filter was broadened sequentially to include higher frequencies (0-25%, 0-50%, 0-75%, 0-100% of the maximum spatial frequency). The filtered images respectively show the contribution of low, intermediate, high-intermediate and high frequency bands. The bulk shape and contrast of the imaged object is captured in the low frequency band, whereas the fine structural details such as edges are captured in the intermediate and partly high-intermediate frequency bands. There is no apparent contribution from the high frequency band. 102

4.12 Synthesis results are shown for a sample cross section from the IXI dataset along with the true target (reference) and the source image (source). Images shown in (a) the spatial domain (b) the spatial-frequency (k-space) domain. White circular boundaries in the k-space representation of the source delineate the boundaries of the low, intermediate, high-intermediate and high frequency bands. The pGAN method more accurately synthesizes the target image as evidenced by the better match in energy distribution across k-space 102

4.13 To examine reliability of the synthesis methods against noise, pGAN was trained on noisy images generated from the IXI dataset by adding artificial Rician noise. A sample cross-section from T_1-weighted (a) and T_2-weighted (b) acquisitions along with their noise-added versions are displayed. 104

4.14 Spectral distribution of k-space intensity values was examined for T_1-weighted (a) and T_2-weighted (b) images in the IXI dataset relative to the artificial noise added to the images. Average k-space intensity was calculated within low (0-25%), intermediate (25-50%), high-intermediate (50-75%) and high (75-100%) spatial frequency bands. Bar plots show the results averaged across the test set. The contribution of added noise to low and intermediate frequency bands is small relative to the original image. In contrast, noise more dominantly contributes to high-intermediate and high frequency bands. 105

5.1 The proposed rsGAN method synergistically recovers undersampled multi-contrast MRI acquisitions by complementarily using three priors: shared high-frequency priors available in fully-sampled or lightly undersampled acquisitions of one or more source contrasts to preserve high-spatial-frequency details, low-frequency priors available in highly undersampled acquisitions of one or more target contrasts to prevent feature leakage/loss, and a perceptual prior to improve recovery of high-level features. The input-to-output mapping is implemented using a conditional adversarial network with a generator and a discriminator. The generator learns to recover realistic high-quality target-contrast images by minimizing a pixel-wise, a perceptual and an adversarial loss function. The discriminator learns to discriminate between synthetic and real pairs of multi-contrast images by maximizing the adversarial loss function. 122

5.2 Examples of undersampling patterns for acquisitions accelerated in a broad range (R= 5x, 10x, 20x, 30x, 40x, 50x) 136

5.3 The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1- and T_2-weighted images from the MIDAS dataset. The acquisition for the source contrast was fully sampled, and the acquisition for the target contrast was undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. PSNR was measured between recovered and fully-sampled reference target-contrast images. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_1 is the source contrast and T_2 is the target contrast. **(b)** PSNR (mean±standard error) when T_2 is the source contrast and T_1 is the target contrast. The performance of sGAN remains constant across R since it does not use any evidence from the target-contrast acquisitions. As expected, the performance of rGAN, jGAN, and rsGAN gradually decreases for higher values of R where the evidence from the target contrast becomes scarce. However, rsGAN performs well even at very high acceleration factors.. 141

5.4 T_2-weighted images in the MIDAS dataset were recovered from heavily undersampled acquisitions (R=10x, 20x, 30x, 40x, 50x). The acquisition for the source contrast (T_1-weighted) was fully sampled. Target-contrast images recovered by ZF (zero-filled Fourier reconstruction), sGAN, jGAN, and rsGAN are shown with the fully-sampled reference image. As the value of R increases the performance of jGAN degrades significantly. Meanwhile, rsGAN maintains high-quality recovered images due to use of additional information from the source contrast. Regions with enhanced recovery in rsGAN are marked with arrows. 154

5.5 T_1-weighted images in the MIDAS dataset were recovered from heavily undersampled acquisitions (R=10x, 20x, 30x, 40x, 50x). The acquisition for the source contrast (T_2-weighted) was fully sampled. Target-contrast images recovered by ZF, sGAN, jGAN, and rsGAN are shown with the fully-sampled reference image. Regions with enhanced recovery in rsGAN are marked with arrows. 155

5.6 Multi-contrast images in the MIDAS dataset were recovered, where the source contrast was fully sampled and the target contrast was undersampled at R=50x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images are shown along with the fully-sampled reference image and the source-contrast image. **(b)** Recovered T_1-weighted images are shown along with the fully-sampled reference image and the source-contrast image. rsGAN yields visually accurate recovery of the target-contrast image compared to sGAN and jGAN. Sample regions that are better recovered by rsGAN are marked with arrows. 156

5.7 The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1- and T_2-weighted images from the BRATS dataset. The acquisition for the source contrast was fully sampled, and the acquisition for the target contrast was undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. PSNR was measured between recovered and fully-sampled reference target-contrast images. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_1 is the source contrast and T_2 is the target contrast. **(b)** PSNR (mean±standard error) when T_2 is the source contrast and T_1 is the target contrast. 160

5.8 Multi-contrast images in the BRATS dataset were recovered, where the source contrast was fully sampled and the target contrast was undersampled at R=50x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images along with the fully-sampled reference image and the source-contrast image. **(b)** Recovered T_1-weighted images along with the fully-sampled reference image and the source-contrast image. rsGAN yields visually superior images compared to sGAN and jGAN. Note that sGAN suffers from either loss of features in the target contrast or synthesis of artefactual features. Meanwhile, jGAN suffers from excessive loss of high spatial frequency information. Sample regions that are more accurately recovered by rsGAN are marked with arrows. 161

5.9 The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1-, T_2- and PD-weighted images from the IXI dataset. The acquisition for the source contrast (T_1-weighted) was lightly undersampled by R_{T_1} =1x, 2x, 3x, and the acquisitions for the target contrasts (T_2- and PD-weighted) were heavily undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_2 is the target contrast. **(b)** PSNR (mean±standard error) for sGAN, rGAN, jGAN, and rsGAN when PD is the target contrast. As expected, rsGAN outperforms sGAN, rGAN, and jGAN at all R. At the same time, performance of rsGAN is highly similar for distinct values of R_{T_1} . 162

5.10 Multi-contrast images in the IXI dataset were recovered, where the source contrast (T_1-weighted) was lightly undersampled at R_{T_1} =2x, and the target contrasts (T_2- and PD-weighted) were heavily undersampled at R=30x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images. **(b)** Recovered PD-weighted images. Samples regions where rsGAN yields sharper images and improved suppression of aliasing artifacts are marked with arrows. 163

5.11 The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1-, T_2- and PD-weighted images from the multi-coil dataset. The acquisition for the source contrast (T_1-weighted) was lightly undersampled by R_{T_1} =1x, 2x, 3x, and the acquisitions for the target contrasts (T_2- and PD-weighted) were heavily undersampled by R=5x, 10x, 15x, 20x, 25x, 30x. **(a)** PSNR (mean±standard error) across the test images (coronal cross-sections) for rsGAN, rGAN, jGAN, and sGAN when T_2 is the target contrast. **(b)** PSNR (mean±standard error) for sGAN, rGAN, jGAN, and rsGAN when PD is the target contrast. As expected, rsGAN outperforms sGAN, rGAN, and jGAN at high values of R. At the same time, performance of rsGAN is highly similar for distinct values of R_{T_1}. 165

5.12 Multi-contrast images in the multi-coil dataset were recovered, where the source contrast (T_1-weighted) was fully sampled, and the target contrasts (T_2- and PD-weighted) were heavily undersampled at R=10x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images. **(b)** Recovered PD-weighted images. Sample regions that are better recovered by rsGAN are marked with arrows. 165

5.13 FLAIR images in the BRATS dataset were recovered, where the source contrast (T_1) was fully sampled and the target contrast (FLAIR) was undersampled at R=50x. Images recovered using ZF, rGAN, jGAN and rsGAN along with the fully-sampled reference image and the source-contrast image are shown. rsGAN yields visually superior images compared to rGAN and jGAN. Both rGAN and jGAN suffer from excessive loss of high spatial frequency information. Sample regions that are more accurately recovered by rsGAN are marked with arrows. 172

List of Tables

3.1 Quality of recovered single-coil magnitude T_1-weighted images . . 41
3.2 Quality of recovered single-coil magnitude T_1-weighted images . . 67
3.3 Quality of recovered single-coil magnitude T_1-weighted images . . 68
3.4 Quality of recovered single-coil magnitude T_1-weighted images . . 69
3.5 Quality of recovered single-coil magnitude T_2-weighted images . . 70
3.6 Quality of recovered multi-coil complex T_1-weighted images 71
3.7 Quality of recovered multi-coil complex T_2-weighted images 71
3.8 Quality of recovered single-coil complex T_1-weighted images . . . 72
3.9 Quality of recovered single-coil complex T_2-weighted images . . . 73

4.1 Quality of Synthesis: Perceptual loss 112
4.2 Quality of Synthesis in the MIDAS dataset: Single cross-section models 112
4.3 Quality of Synthesis: Number of cross-sections for pGAN 112
4.4 Quality of Synthesis: Number of cross-sections for $cGAN_{reg}$ 113
4.5 Quality of Synthesis in the MIDAS dataset: Multiple cross-section models (k=3) 113
4.6 Quality of Synthesis in the MIDAS dataset 113
4.7 Quality of Synthesis 114
4.8 Quality of Synthesis in the IXI dataset 114
4.9 Quality of Synthesis in the BRATS dataset 114
4.10 Quality of Synthesis: Modal Comparison on background removed images 115
4.11 Quality of Synthesis in the MIDAS dataset: Control analysis . . . 116
4.12 Quality of Synthesis in the IXI dataset: Control analysis 117

4.13 Quality of Synthesis in the BRATS dataset: Control analysis . . . 118
4.14 Quality of Synthesis in the Midas dataset: Model generalization . 118
4.15 Quality of Synthesis in the IXI dataset: Reliability against noise . 119

5.1 Quality of Recovered Images in the MIDAS Dataset 143
5.2 Quality of Recovered Images in the MIDAS Dataset 157
5.3 Quality of Recovered Images in the BRATS Dataset 158
5.4 Quality of Recovered Images in the BRATS Dataset 159
5.5 Quality of recovered T_2-weighted images in the IXI dataset 164
5.6 Quality of recovered PD-weighted images in the IXI dataset . . . 164
5.7 Quality of Recovered T_2 Images in the MIDAS Dataset 166
5.8 Quality of Recovered T_1 Images in the MIDAS Dataset 166
5.9 Quality of Recovered T_2 Images in the BRATS Dataset 167
5.10 Quality of Recovered T_1 Images in the BRATS Dataset 167
5.11 Quality of Recovered T_2 Images in the IXI Dataset 168
5.12 Quality of Recovered PD Images in the IXI Dataset 168
5.13 Quality of recovered T_2-weighted images in the multi-coil dataset 169
5.14 Quality of recovered PD-weighted images in the multi-coil dataset 169
5.15 Quality of Recovered Images in the IXI Dataset 170
5.16 Quality of Recovered T_2 Images in the BRATS Dataset 170
5.17 Quality of Recovered T_2 Images in the BRATS Dataset 171
5.18 Quality of Recovered FLAIR Images in the BRATS Dataset . . . 171
5.19 Quality of Recovered T_2 and FLAIR Images in the BRATS Dataset 172

Chapter 1

Introduction

Exceptional soft tissue contrast and non-invasive nature has rendered Magnetic Resonance Imaging (MRI) a preferred imaging modality over its counterparts. Despite all the positive aspects of MRI, long scan duration coupled with incooperative patients may prohibit its practical use. This makes acceleration of MR acquisitions a desired alternative.

One predominant framework to accelerate MR acquisitions is to undersample in k-space, and reconstruct the missing samples using reconstruction algorithms. Owing to the inherent compressible nature of MR images, one of the most powerful techniques for recovery of undersampled acquisitions is compressive sensing (CS) [1, 2]. In CS, the ill-posed image recovery problem is solved by imposing regularization in the form of sparsity in known transform domains. Although CS has offerend increased scan efficieny in various applications, pre-defined sparsifying transform domains and poor hyperparamter selection can lead to sub-optimal recovery performance. Recently, deep neural networks (DNNs) have been demonstrated for MR image reconstruction [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]. Unlike CS, DNNs learn the transform domain using large datasets and do not require online hyperparameter tuning. Although DNNs show remarkable recovery performance, they rely heavily on availability of large datasets, which might not be readily available

in some cases. Another issue with the reconstruction approach in general (not specific to DNNs) is the degraded performance at very high acceleration rates due to fewer acquired data samples.

An alternative framework is to form acquisitions of a desired contrast, and synthesize the missing contrasts from the acquired ones. This is typically performed via intensity mappings learned using data-driven methods. One such method is based on learning sparse linear combination between source and target patches [21, 22]. These local methods are unable to capture highly non linear mappings. Other set of techniques rely on non-linear regression algorithms such as DNNs [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38]. While DNNs show exceptional recovery performance, they are trained based on mean absolute or square error which can lead to loss of intermediate-to-high spatial frequency information. Another caveat of the synthesis approach which also limits its diagnostic applications is that the differences in relaxation parameters between source and target contrasts can lead to loss or artifactual synthesis of features.

Contributions

In this thesis we address the aforementioned issues. The contribution of this thesis are as follows: First, we address the issue of data scarcity in deep neural networks for MR image reconstruction. To do this, we pre-train a deep neural network on natural images by simulating pairs of undersampled and fully sampled acquisitions [18]. We then fine-tune the trained network on a few images from the target domain. Our results suggest that a pre-trained network fine-tuned on just a few ten images from the target domain achieves a performance level comparable to the network trained on thousands of images from the target domain.

Next, we tackle the issue of loss of intermediate-to-high spatial frequency while synthesizing MR images of a target contrast given images of a source contrast [27] . We propose conditional generative adversarial networks (GANs) for this purpose. Our results suggest that GANs are able to efficiently recover intermediate-to-high spatial frequency information while reliably recovering the low-spatial-frequency

information.

At the end, we address the general issue associated with isolated reconstruction and synthesis approaches [39]. Reconstruction performance starts degrading towards high acceleration rates. On the other hand, synthesis performance might be subjective to the level of similarity in tissues properties between both source and target contrasts. A high dis-similarity might lead to loss or artifactual synthesis of features. To mitigate these issues, we propose a joint reconstruction and synthesis approach for accelerated MR imaging, where images of the target contrast are highly undersampled, and are complemented with a structural prior in form of lightly undersampled images of the source contrast. The proposed approach enables recovery of images accelerated at high rates, i.e., up to 50x, and is immune to loss or artifactual synthesis of features.

Outline

The thesis is organized as follows. Chapter 2 briefly provides fundamentals of MRI data acquisition and accelerated MRI. Chapters 3, 4 and 5 in this thesis are based on the aforementioned contributions. In chapter 3 we show the transfer learning approach for accelerated MR imaging using deep neural networks. In chapter 4, we demonstrate conditional GANs for multi-contrast MR image synthesis. In chapter 5, we show joint reconstruction and synthesis approach for accelerated MR imaging. Finally, we present concluding remarks and discuss possible future directions in chapter 6.

Chapter 2

Fundamentals of Accelerated MRI

2.1 Magnetic resonance imaging (MRI)

MR imaging primarily relies on atomic nuclei possessing magnetic and angular momentum. In default state, these nuclei are randomly aligned with the net magnetization approximating to zero. Once an external static magnetic field (B_0) is applied, the nuclei start precessing around the applied magnetic field in either parallel or anti-parallel direction. The frequency with which they precess is specific to each nucleus, commonly known as the Larmor frequency.

$$\omega_0 = B_0\gamma \tag{2.1}$$

where ω_0 is the Larmor frequency, B_0 represents the applied magnetic field and γ denotes the nucleus specific gyro-magnetic ratio. More nuclei precess in the parallel direction, also referred to as the longitudinal direction, thus giving rise to a net magnetic field. An external radio-frequency (RF) field having same frequency as the Larmor frequency of the nucleus of interest can tip the net magnetization field towards the plane perpendicular to the longitudinal direction, also known as the transverse plane. Given considerably excessive amount of

water in the human body, hydrogen nuclei or protons are the most commonly imaged nuclei. In MRI, application of an RF field at the Larmor frequency of the protons tips the net magnetization of protons into the transverse direction. The duration for which the RF pulse is applied quantifies the magnetization along the longitudinal and transverse directions. Upon turning the external RF field off, the net transverse magnetization starts decaying with a time constant T_2, and the net longitudinal magnetization starts recovering with a time constant T_1 [40]. The interaction between the magnetic field and the net magnetization can be explained by the Bloch equation as follows:

$$\frac{d\mathbf{M}}{dt} = \mathbf{M} \times \gamma\mathbf{B} - \frac{M_x\mathbf{i} + M_y\mathbf{j}}{T_2} - \frac{M_z\mathbf{i} + M_0\mathbf{k}}{T_1} \tag{2.2}$$

where $\mathbf{M}$ and $\mathbf{B}$ are vectors representing the net magnetization and applied magnetic field, M_0 is the magnetic field at the equilibrium state, and M_x, M_y, and M_z denote the x, y, and z components of the net magnetization. Since the amount of water varies from one tissue to another, T_1 and T_2 values also vary across tissues. This forms the basis of MR imaging.

MRI consists of receiver coils to detect change in flux along the transverse direction. The received signal, however, does not contain any spatial information, and just reflects the changes in overall flux throughout the volume. Therefore, for spatial encoding additional gradient magnetic fields (G_x, G_y, G_z) are also added along (x, y, z) axes on top of B_0. This creates a variation in the magnetic field as a function of spatial location, and manipulating the gradient fields can enable us to resolve spatial distribution of the magnetization. The signal recorded by the receiver coil is given as:

$$s(t) = \int_x \int_y \int_z M^0(x, y, z) e^{-t/T_2(\mathbf{r})} exp(-i\gamma \int_0^t \mathbf{G}(\tau).\mathbf{r}\, d\tau) dx dy dz \tag{2.3}$$

where $M^0(x, y, z)$ is the transverse magnetization at (x, y, z), $\mathbf{r} = [x, y, z]$, and $\mathbf{G} = [G_x, G_y, G_z]$. Conventionally, the integral containing the gradient terms are replaced by k_x, k_y, and k_z respectively, forming what is commonly known as the k-space. Equating $k_x(t) = \frac{\gamma}{2\pi} \int_0^t G_x(\tau)\, d\tau$, $k_y(t) = \frac{\gamma}{2\pi} \int_0^t G_y(\tau)\, d\tau$, and $k_z(t) = \frac{\gamma}{2\pi} \int_0^t G_z(\tau)\, d\tau$, the signal received at each time can be expressed as [40]:

$$s(t) = \int_x \int_y \int_z M^0(x, y, z) e^{-t/T_2(\mathbf{r})} e^{-i2\pi[k_x(t)+k_y(t)+k_z(t)]} dx dy dz \tag{2.4}$$

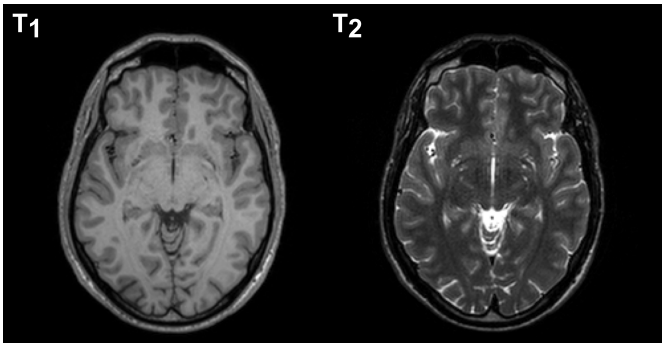

Figure 2.1: Sample T_1- and T_2-weighted images of the human brain.

Note that this is also the 3D Fourier transform of the imaged volume, and signal acquired at each time point corresponds to Fourier transform at a spatial frequency $\mathbf{k} = [k_x, k_y, k_z]$. Desired values of $\mathbf{k}$ can be acquired by adjusting $\mathbf{G}$. This defines the sampling k-space trajectory.

Manipulation of transverse magnetization via applied RF pulse and gradient fields can facilitate in imaging desired properties of the tissues of interest, thus enabling the multi-contrast acquisition of tissues. For instance, one of the most widely used pulse-sequences is spin-echo pulse sequence, where the acquired signal can be expressed as [40]:

$$S_o = k\rho(\mathbf{r})[1 - e^{-TR/T_1(\mathbf{r})}]e^{-TE/T_2(\mathbf{r})} \tag{2.5}$$

where $\rho(\mathbf{r})$ is the proton density, k is a proportionality constant, TR is the repetition time of the pulse sequence, and TE is the echo time. Here, TR and TE values can be adjusted to give more weight to T_1 or T_2 values of the tissues. For example low value of TE and moderate value of TR can produce T_1-weighted contrast. Similarly moderate value of TE and long value of TR can produce T_2-weighted contrast. Sample T_1- and T_2-weighted images are shown in Fig. 2.1.

2.2 Accelerated MRI

Despite exceptional soft tissue contrast in MRI, prolonged scan duration may be a limiting factor. In those scenarios accelerating MR acquisitions without

degrading the image quality can be a viable option.

2.2.1 Reconstruction

One popular framework for acceleration of MR acquisition is undersampling in k-space followed by a tailored reconstruction method (Fig. 2.2). The recovery problem can be formulated as:

$$F_u x = x_u^k \tag{2.6}$$

where F_u denotes the partial Fourier transform operator at the sampled k-space locations, x is the image to be reconstructed, and x_u^k are the acquired k-space data. Note that Eq. 2.6 is ill-posed, and thus requires some prior information regarding x. This information is incorporated in the form of a regularization term $R(x)$:

$$x_{rec} = \arg\min_x \quad \lambda||F_u x - x_u^k||_2 + R(x) \tag{2.7}$$

Given the inherent compressible nature of MR images, a popular technique for recovery of undersampled MR acquisitions is compressive sensing (CS) [1, 2]. In CS, the inverse image recovery problem in Eq. 2.7 is solved by equating $R(x)$ to sparsity in a known transform domain. Despite improvements in scanning efficiency, CS relies on pre-defined sparsifying transform domains, which can be sub-optimal. Furthermore, it requires careful hyper-parameter tuning for optimal performance.

More recently, deep neural networks (DNNs) have been proposed for MR image reconstruction which do not rely on sparsity priors in pre-defined transform domains [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]. Instead, DNNs learn these priors offline in a data-driven manner using large datasets constituting pairs of fully sampled and undersampled images. DNNs are typically trained to minimize the following loss function:

$$\mathcal{L}_{rec}(\theta) = E_{x_{u-t},x_t}||N(x_{u-t};\theta) - x_t||_p \tag{2.8}$$

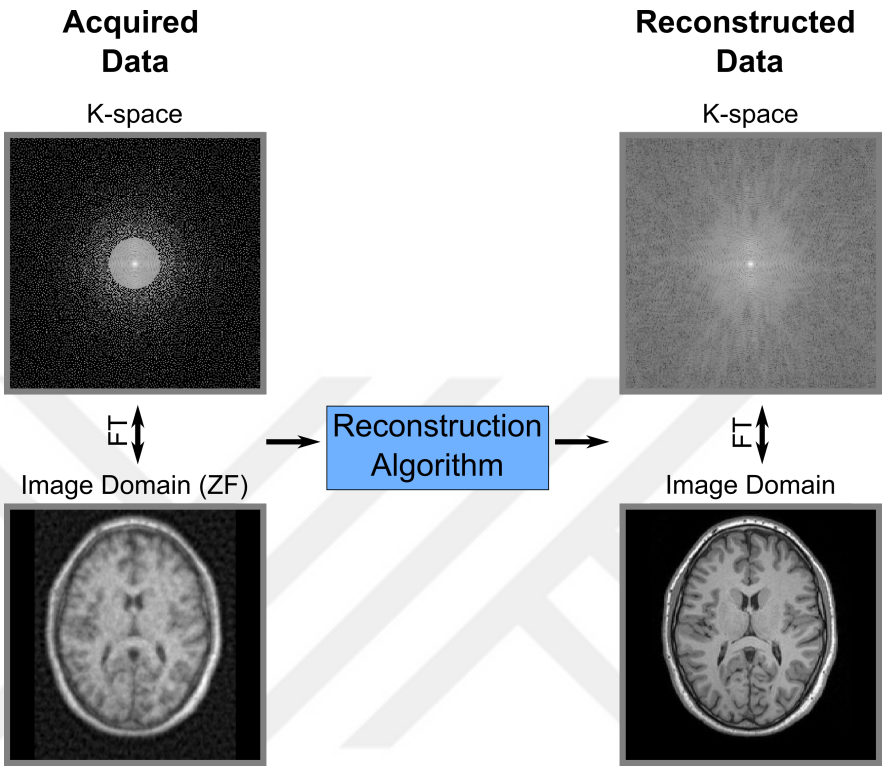

Figure 2.2: In MR image reconstruction, acquisitions are undersampled in k-space followed by a tailored reconstruction algorithm.

where x_{u-t} and x_t represent undersampled and fully-sampled training images, $N(x_{u-t};\theta)$ is the reconstructed output of the DNN having parameters θ, and $||.||_p$ denotes ℓ_p-norm (typically 1 or 2). During online reconstruction, Eq. 2.7 can be formulated as:

$$x_{rec} = \arg\min_{x} \quad \lambda||F_u x - x_u^k||_2 + ||N(x_u;\theta^*) - x||_2 \tag{2.9}$$

where $N(x_u;\theta^*)$ is the image reconstructed by the trained network with parameters θ^*. The consistency with the acquired data is weighed against consistency with the DNN-based population-driven prior.

2.2.2 Synthesis

Another alternative framework for accelerated MRI is synthesis. In synthesis, an image x of a source contrast (X) is acquired, and it is used to synthesize

image y of a target contrast (Y). Synthesis mainly performs intensity-based mappings between the source and target contrasts learned via readily available paired images of source and target contrasts, x_t and y_t respectively. Traditional methods rely on learning sparse linear combinations between source and target contrast patches [21, 22]. The combinations is obtained by solving the following optimization problem:

$$\alpha(j) = \underset{\alpha(j)}{\arg\min} \quad ||x_t(j) - \Phi_x.\alpha(j)||_2 + ||\alpha(j)||_1 \tag{2.10}$$

where $\alpha(j)$ denotes the learned combination coefficients for the jth patch, $x_t(j)$ denotes the jth patch in the source contrast, and Φ_x denotes the dictionary formed using patches from x_t. The same linear combination is applied to patches in y_t to estimate y.

$$\widehat{y}(j) = \Phi_y.\alpha(j) \tag{2.11}$$

where Φ_y denotes the dictionary formed using patches from y_t, and $\widehat{y}(j)$ is the jth patch of the final synthesized image. Such methods are unable to capture highly complex non-linear relations between source and target contrasts. More recent methods such as DNNs cast it as a non-linear regression problem [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38]. DNNs learn a mapping from X to Y using x_t and y_t.

$$\mathcal{L}_{syn}(\theta) = E_{x_t,y_t}||N(x_t;\theta) - y_t||_p \tag{2.12}$$

where x_t and y_t represent training images of the source and target contrasts, and $N(x_t;\theta)$ is the image synthesized via DNN having parameters θ. The learned mapping is then applied to estimate y from x.

$$\widehat{y} = N(x;\theta^*) \tag{2.13}$$

where $\widehat{y}$ is the image synthesized by the trained network $N(x;\theta^*)$ with parameters θ^*.

Chapter 3

A TransferLearning Approach for Accelerated MRI Using Deep Neural Networks

The contents of this chapter reflect the work reported in the following publications:

- S. U. H. Dar, M. Özbey, A. B. Çatlı, and T. Çukur, "A transfer-learning approach for accelerated mri using deep neural networks," *arXiv preprint*, 2017
- S. U. H. Dar and T. Çukur, "Transfer learning for reconstruction of accelerated MRI acquisitions via neural networks," in *26th Annual Meeting of ISMRM*, p. 0569, 2018
- S. U. H. Dar, M. Özbey, A. B. Çatlı, and T. Çukur, "A transfer-learning approach for accelerated MRI using deep neural networks," *Magnetic Resonance in Medicine*, vol. 84, no. 2, pp. 663–685, 2020

3.1 Introduction

The unparalleled soft-tissue contrast in MRI has rendered it a preferred modality in many diagnostic applications, but long scan durations limit its clinical use. Acquisitions can be accelerated by undersampling in k-space, and a tailored reconstruction can be used to recover unacquired data. Because MR images are inherently compressible, a popular framework for accelerated MRI has been compressive sensing (CS) [43, 1]. CS has offered improvements in scan efficiency in many applications [1, 44, 45, 46, 47, 48] including structural [1], angiographic [44], functional [45], diffusion [46], and parametric imaging [47]. Yet the CS framework is not without limitation. First, CS involves nonlinear optimization algorithms that scale poorly with growing data size and hamper clinical workflow. Second, CS commonly assumes that MRI data are sparse in fixed transform domains, such as finite differences or wavelet transforms. Recent studies highlight the need for learning the transform domains specific to each dataset to optimize performance [49]. Lastly, CS requires careful parameter tuning (e.g., for regularization) for optimal performance. While several approaches were proposed for data-driven parameter tuning [50, 51], these methods can induce further computational burden.

Neural network (NN) architectures that reconstruct images from undersampled data have recently been proposed to address the above mentioned limitations. Improved image quality over traditional CS has readily been demonstrated for several applications including angiographic [52], cardiac [53, 4, 11], brain [15, 16, 54, 55, 56, 57, 58, 12, 8, 59, 13, 14, 3, 11, 60, 61, 62, 63, 59, 64, 65], abdominal [66, 17, 67], and musculoskeletal imaging [68, 5, 69, 70, 71]. The common approach is to train a network off-line using a relatively large set of fully-sampled MRI data, and then use it for on-line reconstruction of undersampled data. Reconstructions can be achieved in several hundred milliseconds, significantly reducing computational burden [5, 69]. The NN framework also alleviates the need for ad hoc selection of transform domains. For example, a recent study used a cascade of convolutional neural networks (CNNs) to recover images directly from zero-filled Fourier reconstructions of undersampled data [53, 8, 69]. The trained

CNN layers reflect suitable transforms for image reconstruction. The NN framework introduces more tunable hyperparameters (e.g., number of layers, units, activation functions) than would be required in CS. However, previous studies demonstrate that hyperparameters optimized during the training phase generally perform well in the testing phase [69]. Taken together, these advantages render the NN framework a promising avenue for accelerated MRI.

A common strategy to enhance network performance is to boost model complexity by increasing the number of layers and units in the architecture. A large set of training data must then be used to reliably learn the numerous model parameters [72]. Previous studies either used an extensive database of MR images comprising several tens to hundreds of subjects [4, 13, 5], or data augmentation procedures to artificially expand the size of training data [53, 4]. For instance, an early study performed training on T_1-weighted brain images from nearly 500 subjects in the human connectome project (HCP) database, and testing on T_2-weighted images [13]. Yet, it remains unclear how well a network trained on images acquired with a specific type of tissue contrast generalize to images acquired with different contrasts. Furthermore, for optimal reconstruction performance the network must be trained on images acquired with the same scan protocol that it later will be tested on. However, large databases such as those provided by the HCP may not be readily available in many applications, potentially rendering NN-based reconstructions suboptimal.

In this study, we propose a transfer-learning approach to address the problem of data scarcity in network training for accelerated MRI (Fig. 3.1). In transfer-learning, network training is performed in some domain where large datasets are available, and knowledge captured by the trained network is then transferred to a different domain where data are scarce [73, 74]. Domain transfer was previously used to suppress coherent aliasing artifacts in projection reconstruction acquisitions [16], to perform non-Cartesian to Cartesian interpolation in k-space [13], and to assess the robustness of network reconstructions to variations in SNR and undersampling patterns [70]. In contrast, we employ transfer-learning to enhance NN-based reconstructions of randomly undersampled acquisitions in the testing domain. A deep CNN architecture with multiple subnetworks is taken as a model

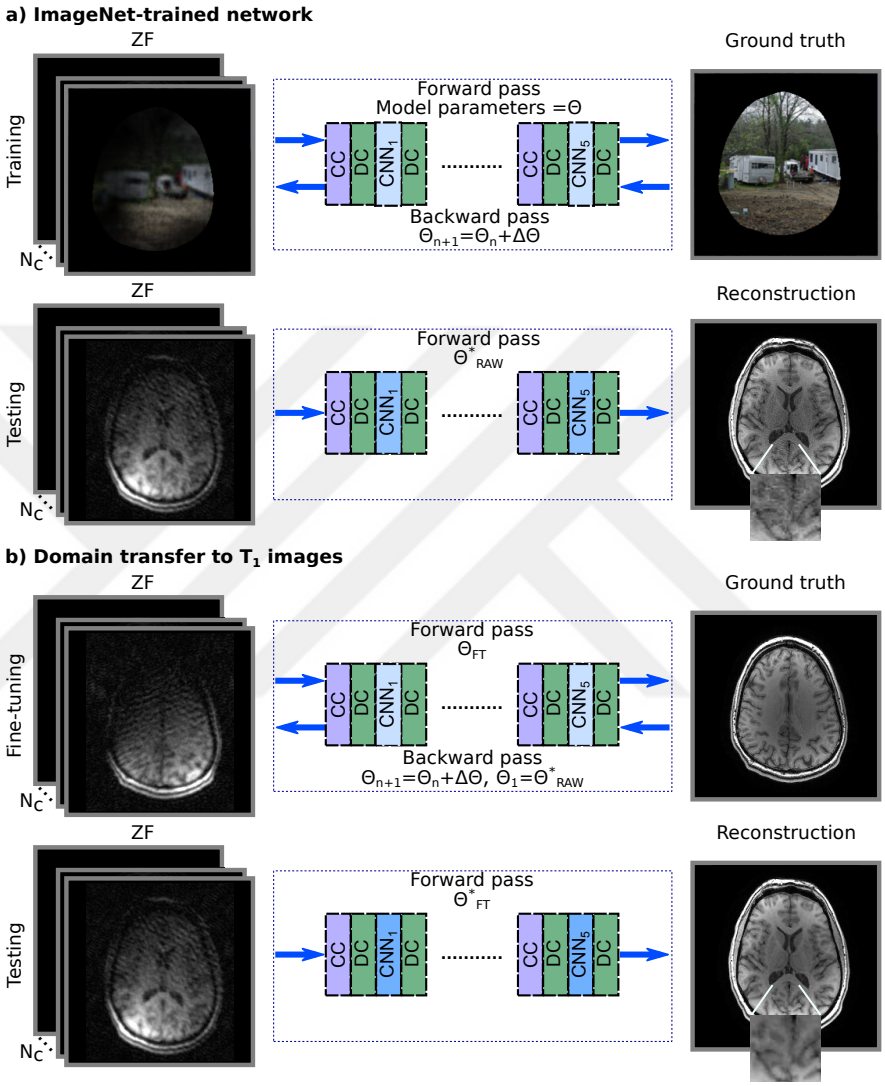

Figure 3.1: Proposed transfer-learning approach for NN-based reconstructions of multi-coil (N_c coils) undersampled acquisitions. A deep architecture with multiple subnetworks is used. The subnetworks consist of calibration consistency CC and convolutional neural network CNN blocks, each followed by a data consistency block DC. (a) Each CNN block is trained sequentially to reconstruct synthetic multi-coil natural images from ImageNet, given zero-filled Fourier reconstructions of their undersampled versions. Due to differences in the characteristics of natural and MR images, the ImageNet-trained network will yield suboptimal performance when directly tested on MR images. (b) For domain transfer, the ImageNet-trained network is fine-tuned end-to-end in the testing domain using few tens of images. This approach enables successful domain transfer between natural and MR images.

network [53]. For reconstruction of multi-coil data, calibration consistency, data consistency and CNN blocks are incorporated to synthesize missing samples. In the training domain using several thousand images, the network is pretrained to reconstruct reference images from zero-filled reconstructions of undersampled data. The trained network is then fine-tuned end-to-end in the testing domain using few tens of images.

To demonstrate the proposed approach, comprehensive evaluations were performed across a broad range of acceleration factors (R=4-10) on T_1- and T_2-weighted brain images, considering both single-coil data from a public database, and multi-coil data acquired on a 3T scanner. Separate network models were learnt for domain transfer between natural and MR images (ImageNet and T_1- or T_2-weighted). Domain-transferred networks were quantitatively compared against networks trained in the testing domain, and against conventional CS reconstructions in the single-coil setting [43, 1] and iTerative Self-consistent Parallel Imaging Reconstruction (SPIRiT) in the multi-coil setting [75]. We find that domain-transferred networks fine-tuned with tens of images achieve nearly identical performance to networks trained directly in the testing domain using thousands of images, and that networks outperform conventional image reconstruction methods.

3.2 Methods

In accelerated MRI, an undersampled acquisition is followed by a reconstruction to recover missing k-space samples. This recovery can be formulated as a linear inverse problem:

$$F_u x = y_u \tag{3.1}$$

where x denotes the image to be reconstructed, F_u is the partial Fourier transform operator at the sampled k-space locations, and y_u denotes acquired k-space data. Since Eq. 3.1 is underdetermined, additional prior information is typically

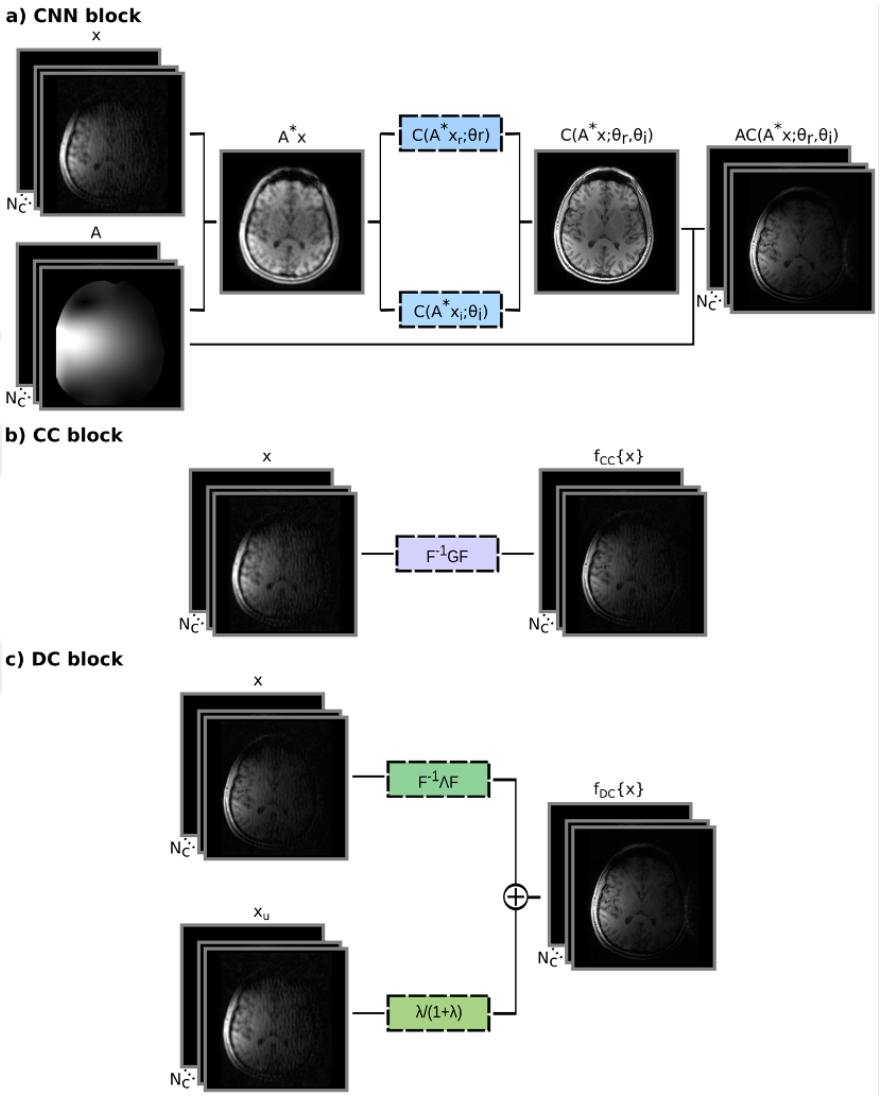

Figure 3.2: Demonstration of (a) convolutional neural network (CNN), (b) calibration consistency (CC) and (c) data consistency (DC) blocks given a multi-coil image x as an input. (a) The CNN block first combines undersampled multi-coil images using coil-sensitivity maps A, estimated via ESPIRiT. Real and imaginary parts of the coil-combined image are then reconstructed using two separate networks. The outputs of the real and imaginary networks are joined to form a complex image, which is then back projected onto individual coils again using the coil sensitivity maps. (b) The CC block transforms the input image into Fourier domain, applies the interpolation operator on multi-coil k-space data, and converts the image back into image domain. (c) The DC block performs a weighted combination of samples recovered by the previous block (CNN or CC) and the originally-acquired samples.

incorporated in the form of a regularization term:

$$x_{rec} = \arg\min_{x} \quad \lambda||F_u x - y||_2 + R(x) \tag{3.2}$$

Here, the first term enforces consistency between acquired and reconstructed data, whereas $R(x)$ enforces prior information to improve reconstruction performance. In CS, $R(x)$ typically corresponds to L1-norm of the image in a known transform domain (e.g., wavelet transform or finite differences transform).

The solution of Eq. 3.2 involves non-linear optimization algorithms that are often computationally complex. This reduces clinical feasibility as reconstruction time becomes prohibitive with increasing size of data. Furthermore, assuming ad hoc selection of fixed transform domains leads to suboptimal reconstructions in many applications [49]. Lastly, it is often challenging to find a set of reconstruction parameters that work optimally across subjects [76].

3.2.1 Multi-Coil Data

For reconstruction of multi-coil data, a hybrid of parallel imaging [75, 77, 78, 79] and compressed sensing [80, 81, 82, 83, 84] approach is commonly used. In the common SPIRiT method, k-space samples are synthesized as a weighted linear combination of acquired samples across neighboring k-space locations and coils [75]. The synthesis operation can be formulated as:

$$\widehat{y_m} = \sum_{j=1}^{N_C} g_{mj} \circledast y_j \tag{3.3}$$

where $\circledast$ is the convolution operator, g_{mj} denotes weights of the interpolation kernel that takes as input data for the j^{th} coil (y_j) and outputs data for the m^{th} coil ($\widehat{y_m}$), and N_C denotes the number of coils. For each coil, the interpolation kernel is estimated from calibration data y^c, a fully sampled central k-space region. For the m^{th} coil, the estimation is performed via Tikhonov regularized regression as follows:

$$g_m = (Y^*Y + \beta I)Y^* y_m^c \tag{3.4}$$

where g_m is obtained by aggregating g_{mj} across coils, y_m^c are calibration data from the m^{th} coil, Y is obtained by aggregating calibration data y_j^c in form of a matrix, and β is the Tikhonov regularization parameter.

Given the entire k-space data y, Eq. 3.3 can be expressed in matrix form with the use of an interpolation operator G as follows:

$$\hat{y} = Gy \tag{3.5}$$

where $\hat{y}$ is the recovered k-space data, and G is the operator that performs interpolation in matrix form [75].

In SPIRiT [75], the recovery problem in Eq. 3.2 can be reformulated as:

$$x_{rec} = \arg\min_x \quad \lambda||F_u x - y||_2 + (G - I)Fx + R(x) \tag{3.6}$$

where x denotes multi-coil images to be reconstructed, y_u denotes acquired multi-coil k-space data, F is the forward Fourier transform operator and G denotes the interpolation operator that synthesizes unacquired samples in terms of acquired samples across neighboring k-space and coils. To enforce sparsity, $R(x)$ can be selected as the L1-norm of wavelet coefficients. One efficient way to solve Eq. 3.6 is via the projection onto convex sets (POCS) algorithm [85]. POCS alternates among a calibration-consistency (CC) projection that applies G, a sparsity projection that enforces sparsity in the transform domain, and a data-consistency (DC) projection.

3.2.2 MRI Reconstruction via Neural Networks (NN)

3.2.2.1 Single-coil data

In the NN framework, a network architecture is used for reconstruction instead of explicit transform-domain constraints. Network training is performed via a supervised learning procedure, with the aim to find the set of network parameters that yield accurate reconstructions of undersampled acquisitions. This procedure is performed on a large set of training data (with N_{train} samples), where

fully-sampled reference acquisitions are retrospectively undersampled. Network training typically amounts to minimizing the following loss function [15]:

$$\min_{\theta} = \sum_{n=1}^{N_{train}} \frac{1}{N_{train}} ||C(x_{un};\theta) - x_{refn}||_2 \tag{3.7}$$

where x_{un} represents the Fourier reconstruction of nth undersampled acquisition, x_{refn} represents the respective Fourier reconstruction of the fully-sampled acquisition, $C(x_{un};\theta)$ denotes the output of the network given the input image x_{un} and the network parameters θ. To reduce sensitivity to outliers, here we minimized a hybrid loss that includes both mean-squared error and mean-absolute error terms. To minimize over-fitting, we further added an L2-regularization term on the network parameters. Therefore, neural network training was performed with the following loss function:

$$\begin{aligned} \min_{\theta} = & \sum_{n=1}^{N_{train}} \frac{1}{N_{train}} ||C(x_{un};\theta) - x_{refn}||_2 + \\ & \sum_{n=1}^{N_{train}} \frac{1}{N_{train}} ||C(x_{un};\theta) - x_{refn}||_1 + \gamma_\theta ||\theta||_2 \end{aligned} \tag{3.8}$$

where γ_θ is the regularization parameter for network parameters.

A network trained on a sufficiently large set of training examples can then be used to reconstruct an undersampled acquisition from an independent test dataset. This reconstruction can be achieved by reformulating the problem in Eq. 3.2 [15]:

$$x_{rec} = \arg\min_{x} \quad \lambda ||F_u x - y||_2 + ||C(x_u;\theta^*) - x||_2 \tag{3.9}$$

where $C(x_u;\theta^*)$ is the output of the trained network with optimized parameters θ^*. Note that the problem in Eq. 3.9 has the following closed-form solution [15]:

$$y_{ir}(k) = \begin{cases} \dfrac{FC(x_u;\theta^*)(k) + \lambda y_u(k)}{1+\lambda} & \text{if } k \in \Omega \\ \\ FC(x_u;\theta^*)(k) & \text{otherwise} \end{cases}$$

$$x_{rec} = F^{-1} y_{rec} \tag{3.10}$$

where k denotes k-space location, Ω represents the set of acquired k-space locations, F and F^{-1} are the forward and backward Fourier transform operators, and x_{rec} is the reconstructed image. The solution outlined in Eq. 3.10 performs two separate projections during reconstruction. The first projection calculates the output of the trained neural network $C(x_u; \theta^*)$ given the input image x_u, the Fourier reconstruction of undersampled data. The second projection enforces data consistency. The parameter λ in Eq. 3.10 controls the relative weighting between data samples that are originally acquired and those that are recovered by the network. Here we used $\lambda{=}\infty$ to enforce data consistency strictly. Given an input x_{in} in the image domain, the data consistency projection outlined in Eq. 3.10 can be compactly expressed as [53]:

$$f_{DC}(x_{in}) = F^{-1}\Lambda \mathrm{F} x_{in} + \frac{\lambda}{1+\lambda} x_u \tag{3.11}$$

where Λ is a diagonal matrix:

$$\Lambda_{kk} = \begin{cases} \dfrac{1}{1+\lambda} & \text{if } k \in \Omega \\ 1 & \text{otherwise} \end{cases} \tag{3.12}$$

Conventional optimization algorithms for CS run iteratively to progressively minimize the loss function. A similar approach can also be adopted for NN-based reconstructions [53, 8, 69]. Here, we cascaded several CNN blocks in series with DC projections interleaved between consecutive CNN blocks [53]. In this architecture, the input x_{ip} to the p^{th} CNN block was formed as:

$$x_{ip} = \begin{cases} x_{un}, & \text{if p=1} \\ f_{DC}(C_{p-1}(f_{DC}(C_{p-2}(f_{DC} \cdots C_1(x_{un}; \theta_1^*)); \cdots \theta_{p-1}^*)), & \text{if p>1} \end{cases} \tag{3.13}$$

where θ_p^* denotes the parameters of the p^{th} CNN block. Starting with the initial network with $p = 1$, each CNN block was trained sequentially by solving the following optimization problem:

$$\begin{aligned} \min_{\theta_p} = \sum_{n=1}^{N_{train}} \frac{1}{N_{train}} ||C(x_{ip}; \theta_p) - x_{refn}||_2 + \\ \sum_{n=1}^{N_{train}} \frac{1}{N_{train}} ||C(x_{ip}; \theta_p) - x_{refn}||_1 + \gamma_\theta ||\theta_p||_2 \end{aligned} \tag{3.14}$$

While training the p^{th} CNN block, the parameters of preceding networks and thus the input x_{ip} are assumed to be fixed.

3.2.2.2 Multi-Coil Data

Similar to SPIRiT, for multi-coil reconstructions, here we reformulate Eq. 3.6 as:

$$x_{rec} = \arg\min_{x} \quad \lambda||F_u x - y||_2 + (G - I)Fx + ||C(A^* x_u; \theta^*) - A^* x||_2 \tag{3.15}$$

where x denotes the multi-coil images to be reconstructed, A denotes coil-sensitivity profiles using ESPIRiT [86] and A^* denotes its adjoints, and G denotes the interpolation operator in SPIRiT as in Eq. 3.6. The network C has been trained to recover fully-sampled coil-combined images given undersampled coil-combined images as outlined in Eq. 3.8. The trained network regularizes the reconstruction in Eq. 3.15 given undersampled coil-combined images $A^* x_u$. The optimization problem in Eq. 3.15 is solved by alternating projections for calibration-consistency (CC), data-consistency (DC) and convolutional neural network (CNN) blocks (see Fig. 3.2 for details). CNN blocks are cascaded in series with data consistency and calibration consistency projections. Given an input x_{in} in the image domain, the calibration-consistency projection can be compactly expressed as:

$$f_{CC}(x_{in}) = F^{-1} G F x_{in} \tag{3.16}$$

where F and F^{-1} are the forward and backward Fourier transform operators. Note that the input and output of the CC blocks are in the image domain. In this multi-coil implementation, the input x_{ip} to the p^{th} CNN block was formed as:

$$x_{ip} = \begin{cases} x_{un}, & \text{if p=1} \\ f_{DC}(f_{CC}(f_{DC}(AC_{p-1}(A^* f_{DC}(f_{CC} \cdots \\ AC_1(A^* f_{DC}(f_{CC}(x_{un}); \theta_1^*)); \cdots \theta_{p-1}^*)), & \text{if p>1} \end{cases} \tag{3.17}$$

Note that CNN blocks receive coil-combined images, and CC and DC blocks receive multi-coil images as input. A^* converts multi-coil images into a coil-combined image, and A back projects the coil-combined image onto individual coils. CC and CNN blocks are both followed by a DC block.

3.2.3 Datasets

3.2.3.1 Single-coil magnitude images

For demonstrations on single-coil data, two distinct types of datasets were used: MR brain images and natural images. The details are listed below.

MR brain Images

Training deep neural networks for MR image reconstruction typically requires large datasets containing thousands of images that may be difficult to acquire. Yet, in this study we wanted to systematically examine the interaction between the number of training and fine-tuning samples for domain-transferred neural networks. To comprehensively examine this issue, we opted for the publicly available MIDAS dataset with multi-contrast MR images from nearly 100 subjects. T_1-weighted images: We assembled a total of 6500 T_1-weighted images (58 subjects) from the MIDAS database [87]. These images were divided into 4580 training images (42 subjects), 720 fine-tuning images (6 subjects) and 1200 testing images (10 subjects). In the training phase, for CNN block training 4000 images (34 subjects) were used for training while 240 images (2 subjects) were reserved for validation, and for the end-to-end network training 100 images (4 subjects) were used for training while 240 images (2 subjects) were reserved for validation. In the fine-tuning phase, 480 images (4 subjects) were used for fine-tuning and 240 images (2 subjects) were reserved for validation. There was no overlap between subjects included in the training, validation and testing sets. T_1-weighted images analyzed here were collected on a 3T scanner via the following parameters: a 3D gradient-echo sequence, TR=14ms, TE=7.7ms, flip angle=25^o, matrix size=256x176, 1 mm isotropic resolution. T_2-weighted images: We assembled a total of 6100 T_2-weighted images (64 subjects) from the MIDAS database [87]. These images were divided into 4500 training images (48 subjects), 600 fine-tuning images (6 subjects) and 1000 testing images (10 subjects), with no subject overlap between training, validation and testing sets. In the training

phase, for CNN block training 4000 images (40 subjects) were used for training while 200 images (2 subjects) were reserved for validation, and for the end-to-end network training 100 images (4 subjects) were used for training while 200 images (2 subjects) were reserved for validation. In the fine-tuning phase, 400 images (4 subjects) were used for fine-tuning and 200 images (2 subjects) were used for validation. T_2-weighted images analyzed here were collected on a 3T scanner via the following parameters: a 2D spin-echo sequence, TR=7730ms, TE=80ms, flip angle=90^o, matrix size=256x192, 1 mm isotropic resolution. For the fine-tuning phase, images from 4 subjects were reserved. Cross-section images from the reserved subjects were aggregated, and 100 images were randomly selected from within the aggregate set. Therefore, the selected images during the fine-tuning phase contained images from multiple different subjects. Note that the MIDAS dataset contains DICOM images with only magnitude information. Therefore, all analyses were performed for magnitude-only reconstructions.

Natural Images

To perform domain transfer from natural images to single-coil magnitude MR images, we assembled 5100 natural images from the validation set used during the ImageNet Large Scale Visual Recognition Challenge 2011 (ILSVRC2011) [88]. 4000 images were used for training, 100 images were used for end-to-end training, and 1000 images were used for validation. All images were either cropped or zero-padded to yield consistent dimensions of 256x256. Color RGB images were first converted to LAB color space, and the L-channel was extracted to obtain grayscale images.

3.2.3.2 Multi-coil complex images

MR brain images

The proposed approach was also demonstrated on multi-coil complex k-space data. Images from 10 subjects were acquired. Within each subject, 60 central cross-sections containing sizeable amount of brain tissue were selected. Images were then divided into 360 training images (6 subjects), 60 validation images (1 subject) and 180 testing images (3 subjects), with no subject overlap. Images were collected on a 3T Siemens Magnetom scanner (maximum gradient strength of 45mT/m and slew rate of 200 T/m/s) using a 32-channel receive-only head coil. 1) T_1-weighted images: The images were collected via the following parameters: a 3D MP-RAGE sequence [89, 90, 91], TR=2000ms, TE=5.53ms, flip angle=20^0, matrix size=256x192x80, 1 mm x 1 mm x 2 mm resolution. 2) T_2-weighted images: The images were collected via the following parameters: a 3D Spin-Echo sequence, TR=1000ms, TE=118ms, flip angle=90^0, matrix size=256x192x80, 1 mm x 1 mm x 2 mm resolution. Imaging protocols were approved by the local ethics committee at Bilkent University and all participants provided written informed consent. To reduce computational complexity, geometric-decomposition coil compression (GCC) was performed to reduce number of coils from 32 to 8 [92].

Natural images

The multi-coil data mentioned the previous subsection consisted of complex T_1- and T_2-weighted images acquired on a 3T scanner. However, the ImageNet dataset consisted of magnitude images. Therefore, to perform domain transfer from natural images to multi-coil MR images, complex natural images were simulated from 2420 magnitude images in ImageNet by adding sinusoidal phase at random spatial frequencies along each axis varying from -π to +π. The amplitude of the sinusoids was normalized between 0 and 1. Fully-sampled multi-coil T_1-weighted acquisitions from 2 training subjects were selected to extract

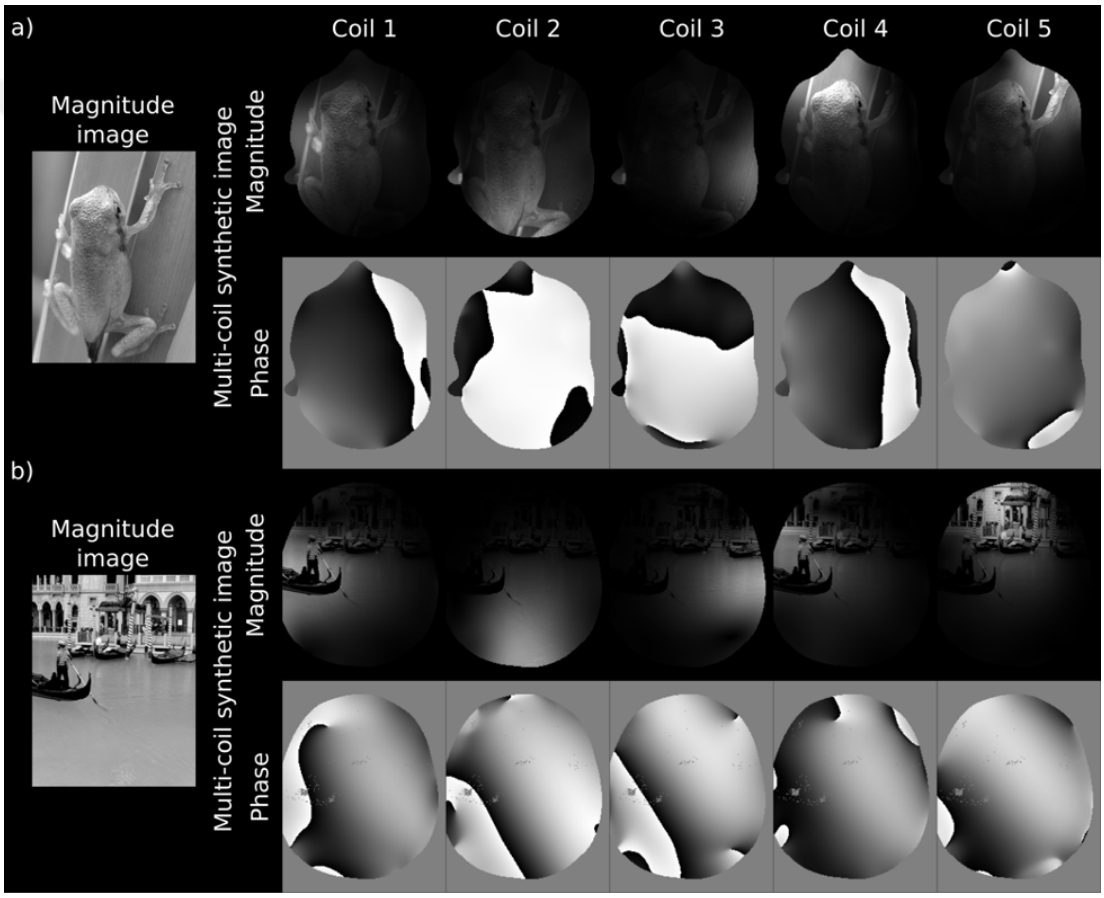

Figure 3.3: Representative synthetic complex multi-coil natural images. Complex multi-coil natural images were simulated from magnitude images in ImageNet (see Methods for details). Magnitude and phase of two simulated multi-coil natural images (a and b) are shown along with their reference magnitude images.

coil-sensitivity maps using ESPIRiT [86]. Each multi-coil complex natural image was then simulated by utilizing coil-sensitivity maps of a randomly selected cross-section from the 2 reserved subjects (see Fig. 3.3 for sample multi-coil complex natural images). Please note that this phase simulation procedure was also demonstrated to enable successful domain transfer in other recent studies on image reconstruction [13, 70]. From the simulated 2420 images, 2000 images were used for initial CNN block training, 360 images were used for end-to-end training, and 60 images were used for validation.

3.2.3.3 Single-coil complex images

Single-coil reconstructions on the MIDAS dataset were performed on magnitude images that were Fourier transformed and undersampled in k-space. To demonstrate the proposed approach on single-coil complex images, we conducted additional experiments using the multi-coil complex MRI data. To do this, multi-coil images were combined via coil-sensitivity maps estimated using ESPIRiT. For domain transfer from natural images to single-coil complex MR images, complex natural images were synthesized from 2420 ImageNet images by adding sinusoidal phase at random spatial frequencies along each axis varying from π to $+\pi$. Note that for domain transfer experiments in the multi-coil case, natural images were multiplied with coil sensitivity maps estimated from actual MRI data to synthesize multi-coil images. This multiplication intrinsically restricts the spatial extent of objects in natural images. When performing domain transfer in the single-coil complex case, we wanted to match the simulation procedures as closely as possible. Therefore, the synthesized images were spatially restricted by utilizing brain masks extracted from coil sensitivity maps of a randomly selected cross-section from two subjects reserved for this purpose. From the simulated 2420 images, 2000 images were used for initial CNN block training, 360 images were used for end-to-end training, and 60 images were used for validation. Data augmentation is a common method to increase data size for network training. Yet, artificially created samples are inherently correlated with the original samples. Since a central aim of the current study was to examine the interaction between the number

of training and fine-tuning samples, no data augmentation was employed to minimize bias due to sample correlation. Undersampling patterns: Images in each dataset were undersampled via variable-density Poisson-disc sampling [75]. All datasets were undersampled for varying acceleration factors (R=4, 6, 8, 10). Fully sampled images were first Fourier transformed and then retrospectively undersampled. To ensure reliability against mask selection, 100 unique undersampling masks were generated and used during the training phase. A different set of 100 undersampling masks were used during the testing phase.

3.2.4 Network training and fine-tuning

We adopted a cascade of neural networks as inspired by [53]. Five subnetworks were cascaded in series. For single-coil magnitude data, the CNN block within each subnetwork contained an input layer, four convolutional layers and an output layer. The input layer consisted of two channels for real imaginary parts of undersampled images. Each convolution operation in the convolutional layers was passed through a rectified linear unit (ReLU) activation. The hidden layers consisted of 64 channels. The output layer consisted of only a single channel for a magnitude reconstruction. For multi-coil complex data, undersampled multi-coil data were combined prior to CNN blocks using coil-sensitivity maps estimated via ESPIRiT. Real and imaginary parts of coil-combined images were then reconstructed using two separate networks, and each network consisted of a single input and output channel. The network outputs were joined to form a coil-combined complex image. Note that the DC block operates on individual-coil data. Thus, prior to the DC block, the coil-combined complex image was back projected onto individual coils again using coil-sensitivity maps.

3.2.4.1 CNN block training

CNN blocks were trained on N_{train} images in the source domain via the back-propagation algorithm [93]. In the forward passes, a batch of 50 samples in

the single-coil case and 10 samples in the multi-coil case were passed through the network to calculate the respective loss function. In the backward passes, network parameters were updated according to the gradients of this function with respect to the parameters. The gradient of the loss function with respect to parameters of the m^{th} hidden layer (θ_m) can be calculated using chain rule:

$$\frac{\partial L}{\partial \theta_m} = \frac{\partial L}{\partial o_l}\frac{\partial o_l}{\partial a_l}\frac{\partial a_l}{\partial \theta_l}\frac{\partial o_{l-1}}{\partial a_{l-1}} \cdots \frac{\partial o_m}{\partial a_m}\frac{\partial a_m}{\partial \theta_m} \tag{3.18}$$

where l is the output layer of the network, a_l is the output of the l^{th} layer, and o_l is the output of the l^{th} layer passed through the activation function. The parameters of the m^{th} layer are only updated if the loss-function gradient flows through all subsequent layers (i.e., gradients are non-zero). Each subnetwork was trained individually for 20 epochs. In the CNN block training, the network parameters were optimized using the ADAM optimizer with a learning rate of $\eta = 10^{-4}$, decay rate for first moment of gradient estimates of $\beta_1 = 0.9$ and decay rate for the second moment of gradient estimate of $\beta_2 = 0.999$ [94]. Connection weights were L2-regularized with a regularization parameter of $\gamma_\Phi = 10^{-6}$.

3.2.4.2 End-to-end network training

Networks formed by sequential training of the CNN blocks were then trained end-to-end on $N_{end-to-end}$ images in the source domain. For single-coil magnitude data, this end-to-end training was performed on only 100 images from the source domain (i.e., $N_{end-to-end}$=100). For single-coil and multi-coil complex data, a relatively smaller set of images were used for initial training (360 images), so end-to-end training was performed on 360 images from the source domain (i.e., $N_{end-to-end}$=360). In the forward passes, a batch of 20 samples in the single-coil case and 4 samples in the multi-coil case were passed through the network to calculate the respective loss function. To perform end-to-end training, the gradients must be calculated through the CNN, DC and CC blocks. The gradient flow through the convolutional network layers that contain basic arithmetic operations and ReLU activation functions are well known [95]. The gradient flow through

DC in Eq. 3.11 with respect to its input x_{in} is given as:

$$\frac{\partial f_{DC}}{\partial x_{in}} = F^{-1}\Lambda F \tag{3.19}$$

due to the linearity of the Fourier operator (F). Similarly, the gradient flow through CC in Eq. 3.16 with respect to its input x_{in} is given as:

$$\frac{\partial f_{CC}}{\partial x_{in}} = GF \tag{3.20}$$

Based on Eqs. 3.19 and 3.20 the gradient of the loss function with respect to output of the j^{th} CNN block is given as:

$$\frac{\partial L}{\partial C_j} = \frac{\partial L}{\partial C_l}\frac{\partial C_{l-1}}{\partial f_{DC,(l-1)2}}\frac{\partial f_{DC,(l-1)2}}{\partial f_{CC,l-1}}\frac{\partial f_{CC,l-1}}{\partial f_{DC,(l-1)1}}\frac{\partial f_{DC,(l-1)1}}{\partial C_{l-1}}\cdots\frac{\partial f_{DC,(j+1)1}}{\partial C_j} \tag{3.21}$$

where l corresponds to the last subnetwork, $f_{DC,(l-1)2}$ corresponds to the DC layer posterior to the $(l-1)^{th}$ CC block, and $f_{DC,(l-1)1}$ corresponds to the DC block posterior to the $(l-1)^{th}$ subnetwork. Once we have the gradient of the loss function with respect to output of the j^{th} CNN block, the gradients of the m^{th} hidden layer (θ_m) within the j^{th} CNN block can be calculated using chain rule.

$$\frac{\partial L}{\partial \theta_m} = \frac{\partial L}{\partial C_j}\frac{\partial C_j}{\partial \theta_l}\frac{\partial \theta_{l-1}}{\partial \theta_{l-2}}....\frac{\partial \theta_{m+1}}{\partial \theta_m} \tag{3.22}$$

where l corresponds to the last layer. If we define the gradient $\frac{\partial L}{\partial \theta_m}$ at the k^{th} iteration as g_m^k, then estimates of the first and second moments of the gradients at the k^{th} iteration can be expressed as:

$$m_m^k = m_m^{k-1}\beta_1 + (1-\beta_1)g_m^k \tag{3.23}$$

$$v_m^k = v_m^{k-1}\beta_2 + (1-\beta_2){g^2}_m^k \tag{3.24}$$

where m_m^k is the estimate of the first moment of the gradient at the k^{th} iteration, β_1 is the decay rate for m_m^k, v_m^k is the estimate of the second moment of the gradient at the k^{th} iteration, and β_2 is the decay rate for v_m^k. The update for the parameters of the m^{th} hidden layer (θ_m) in the k^{th} iteration can then be expressed as:

$$\theta_m^k = \theta_m^{k-1} - \eta\frac{m_m^k}{\sqrt{v_m^k}+\epsilon} \tag{3.25}$$

where η is the learning rate and ϵ is a small constant that avoids division by zero (set to 10^{-8}). During the end-to-end training phase, the ADAM optimizer was used with identical parameters to those used in the subnetwork training, apart from a lower learning rate of 10^{-5} and a total of 100 epochs.

3.2.4.3 Network fine-tuning

A network trained in one domain might lead to sub-optimal performance in a different target domain. For this purpose, end-to-end fine-tuning was performed on a small number of images (Ntune) from the target domain. Gradient calculation and parameter updates were identical to end-to-end network training as described in subsection 2.2.4.2. During the fine-tuning phase, the ADAM optimizer was used with identical parameters to those used in subnetwork training, apart from a lower learning rate of 10^{-5} and a total of 100 epochs.

3.2.5 Network validation

In both training and fine-tuning phases, the number of epochs and learning rate were selected based on reconstruction error (mean absolute error + mean square error) on the validation set. Training and fine-tuning phases exercised early stopping based on network performance on the validation set. During the course of model training, prediction errors will initially decrease on both training and validation sets. Yet, continued training will reduce training error at the expense of elevated validation error. This transition serves as a hallmark symptom of overfitting. To catch the onset of overfitting, we stopped network training based on a convergence criterion. Convergence was taken as the number of epochs where the percentage change in validation error across consecutive epochs fell below 0.1% of the initial validation error. We found that for CNN block training, all CNN blocks converged within 20 epochs, and for end-to-end training all networks converged within 100 epochs (see Fig. 3.4). Learning rate was selected to facilitate convergence while preventing undesirable oscillations in the validation error. We observed the resulting learning rates to be 10^{-4} in the subnetwork training phase, 10^{-5} in the end-to-end training phase and 10^{-4} in the fine-tuning phase. In the fine-tuning phase, since fine-tuning is performed on a few samples, proper selection of the learning rate is more critical. An excessive learning rate can cause the networks to overfit to the fine-tuning samples. This overfitting can be observed in the form of undesirable oscillations and increase in validation error (see

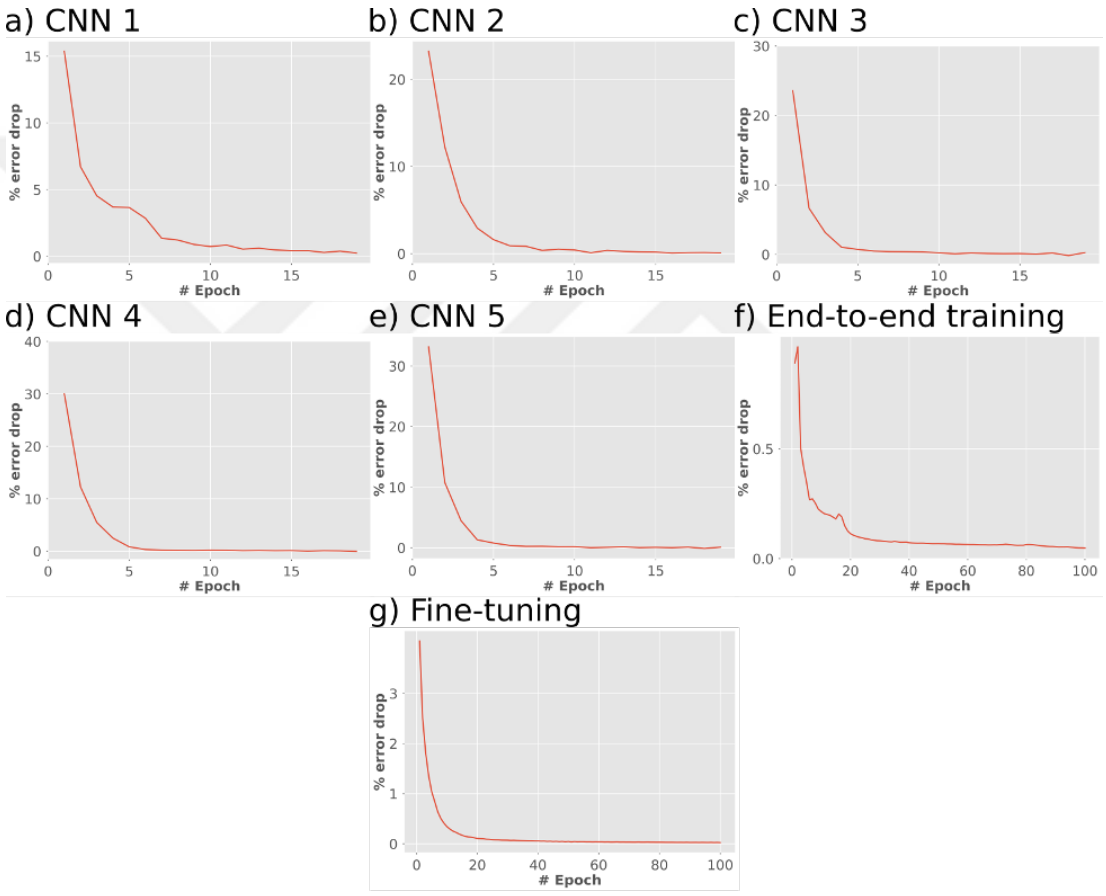

Figure 3.4: Percentage change in validation error as a function of number of epochs for T_2 to T_1 domain transfer at acceleration factor R=4. Results are shown for sequential training of individual CNN blocks (a-e), end-to-end training of the complete network (f) and fine-tuning of the complete network (g). Initial CNN block training was performed on 2000 T_2-weighted images, end-to-end training was performed on 100 T_2-weighted images, and fine-tuning was performed on 20 T_1-weighted images.

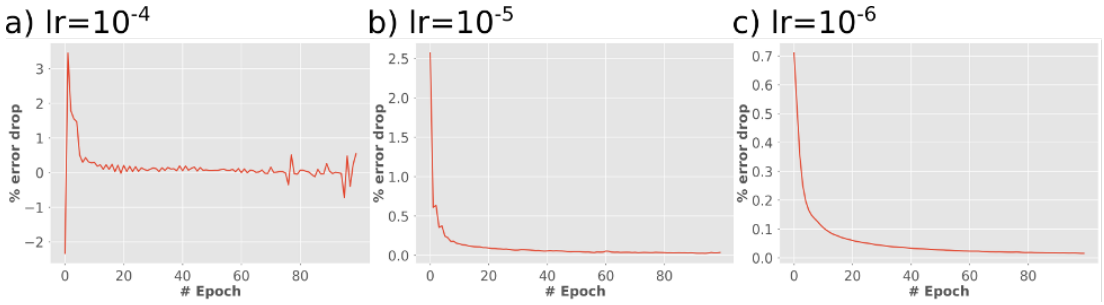

Figure 3.5: Percentage change in validation error as a function of number of epochs for ImageNet to T_1 domain transfer at acceleration factor R=4. Results are shown for learning rates (lr) equal to (a) 10^{-4}, (b) 10^{-5} and (c) 10^{-6}. Initial CNN block training was performed on 2000 ImageNet images, end-to-end training was performed on 100 ImageNet images, and fine-tuning was performed on 20 T_1-weighted images. Learning rate equal to 10^{-5} facilitates convergence while preventing undesirable oscillations in the validation error. A learning rate of 10^{-5} ensures both stable fine-tuning and faster convergence. In contrast, a higher learning rate of 10^{-4} leads to oscillatory behavior in validation error, potentially suggesting overfitting to fine-tuning samples. While fine-tuning is relatively stable for a lower learning rate of 10^{-6}, network convergence is noticeably slower.

Fig. 3.5). In the fine-tuning phase, validation data were again used to select the number of epochs and learning rate, and additionally to determine the number of fine-tuning samples required for successful domain transfer. Peak signal-to-noise ratio (PSNR) values obtained on the validation images were used to assess domain transfer performance. The PSNR convergence point was used to select the number of fine-tuning samples. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing number of fine-tuning samples fell below 0.05% of PSNR for the network trained in the target domain. Both training and fine-tuning phases consisted of separate validation datasets. In the training phase, validation data were exclusively selected from the source domain. For example, the validation set for the ImageNet-trained network contained ImageNet images, whereas the validation set for the T_1-trained network contained T_1-weighted images. In contrast, the validation set in the fine-tuning phase contained data exclusively from the target domain. For example, when T_1 was the target domain, the validation set for both domain-transferred and T_1-trained networks contained an identical set of T_1-weighted images.

3.2.6 Performance analyses

3.2.6.1 Single-coil magnitude data

We first evaluated the performance of networks under implicit domain transfer (i.e., without fine tuning in the target domain). We reasoned that a network trained and tested in the same domain should outperform networks trained and tested on different domains. To investigate this issue, we reconstructed undersampled T_1-weighted acquisitions using the ImageNet-trained and T_2-trained networks for varying acceleration factors (R=4, 6, 8, 10). The reconstructions obtained via these two networks were compared with reference reconstructions obtained from the network trained directly on T_1-weighted images. To ensure that our results were not biased by the selection of a specific MR contrast as the test set, we also reconstructed undersampled T_2-weighted acquisitions using the ImageNet-trained and T_1-trained networks. The reconstructions obtained via these two networks were compared with reference reconstructions obtained from the network trained directly on T_2-weighted images. Next, we evaluated the performance of network under explicit domain transfer (i.e., with fine tuning in the target domain). Networks were fine-tuned end-to-end in the testing domain. When T_1-weighted images were the testing domain, ImageNet-trained and T_2-trained networks were fine-tuned using a small set of T_1-weighted images (Ntune) with size ranging in [0 100]. When T_2-weighted images were the testing domain, ImageNet-trained and T_1-trained networks were fine-tuned using a small set of T_2-weighted images (Ntune) with size ranging in [0 100]. In both cases, the performance of fine-tuned networks was compared with the networks trained and further fine-tuned end-to-end directly in the testing domain on N_{tune} images. We also compared the performance of the fine-tuned networks with limited networks that were obtained via end-to-end training only on N_{tune} images. Reconstruction performance of a fine-tuned network likely depends on the number of both training and fine-tuning images. To examine potential interaction between the number of training and fine-tuning samples, separate networks were trained using training sets of varying size (N_{train}) in [500 4000]. Each network was then fine-tuned using sets of varying size (N_{tune}) in [0 100]. Performance was evaluated to

determine the number of fine-tuning samples that are required to achieve near-optimal performance for each separate size of training set. Optimal performance was taken as the peak signal-to-noise ratio (PSNR) of a network trained directly in the testing domain. Please note that for all aforementioned analyses, networks were also end-to-end trained using a set of 100 images in the source domain (i.e., $N_{end-to-end}$=100). NN-based reconstructions were also compared to those obtained by conventional CS (SparseMRI) [1]. Single-coil CS reconstructions were implemented via a nonlinear conjugate gradient method. Daubechies-4 wavelets were selected as the sparsifying transform. Parameter selection was performed to maximize PSNR on the validation images from the fine-tuning set. Consequently, an L1-regularization parameter of 10^{-3}, and 80 iterations for T_1-weighted acquisitions and 120 iterations for T_2-weighted acquisitions were observed to yield near-optimal performance broadly across R.

3.2.6.2 Multi-Coil Complex Data

We also demonstrated the proposed approach on multi-coil MR images. For this purpose, a network was trained, where initial CNN block training was performed on 2000 (N_{train}) multi-coil complex natural images, and end-to-end training was performed on 360 ($N_{end-to-end}$) additional multi-coil complex natural images (see Section 3.2.3.2 for details). The network was then fine-tuned using a set of multi-coil images (N_{tune}) from the target domain (T_1- or T_2-weighted) with varying size in [0 100]. Here, cross-sections from the training set in the target domain were aggregated, and 100 images were then randomly selected. Reconstruction performance was compared with networks trained using 360 multi-coil MR images from the target domain (6 subjects) and L1-SPIRiT (48). A POCS implementation of SPIRiT was used. For each R, parameter selection was performed to maximize PSNR on validation images drawn from the multi-coil MR image dataset. For T_1-weighted images, an interpolation kernel width of 7, a Tikhonov regularization parameter of 10^{-2} for calibration, an L1-regularization parameter of 10^{-3} were observed to yield near-optimal performance across R. Meanwhile, the optimal number of iterations varied based on acceleration factor. For R= [4, 6, 8, 10], the

following number of iterations= [30, 45, 65, 80] were selected. For T_2-weighted images, an interpolation kernel width of 7, a Tikhonov regularization parameter of 10^{-2} for calibration, an L1-regularization parameter of 10^{-4} were observed to yield near-optimal performance across R. Meanwhile, the optimal number of iterations varied based on acceleration factor. For R= [4, 6, 8, 10], the following number of iterations= [45, 70, 80, 80] were selected. The interpolation kernels optimized for SPIRiT were used in the calibration-consistency blocks of the networks that contained 5 consecutive CC projections. We also inspected the degree of change in model weights following fine-tuning. To inspect the changes, we computed percentage change in coefficients of convolution kernels in CNN layers for R=4-10. Measurements were averaged across neurons within each layer and across R.

3.2.6.3 Single-Coil Complex Data

We also demonstrated the proposed approach on single-coil complex images. For this purpose, initial CNN block training was performed on 2000 (N_{train}) synthetic single-coil complex natural images, and end-to-end training was performed on 360 ($N_{end-to-end}$) additional synthetic single-coil complex natural images (see Section 3.2.3.3 for details). The network was then fine-tuned using a set of single-coil complex images (N_{tune}) from the target domain (T_1- or T_2-weighted) with varying size in [0 100]. Here, cross-sections from the training set in the target domain were aggregated, and 100 images were then randomly selected. Reconstruction performance was compared with networks trained using 360 single-coil MR complex images from the target domain (6 subjects). To quantitatively compared alternative methods, we measured the structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR) between the reconstructed and fully-sampled reference images. For multi-coil data, the reference image was taken as the coil-combined image obtained via weighted linear combination using coil sensitivity maps from ESPIRiT. The training and testing of NN architectures were performed in the TensorFlow framework [96]

using 2 NVIDIA Titan X Pascal GPUs (12 GB VRAM). Single-coil CS reconstructions were performed via libraries in the SparseMRI V0.2 toolbox available at `https://people.eecs.berkeley.edu/~mlustig/Software.html`. Multi-coil CS reconstructions were performed via libraries in the SPIRiT V0.3 toolbox available at `https://people.eecs.berkeley.edu/~mlustig/Software.html`.

3.3 Results

3.3.1 Single-coil magnitude data

3.3.1.1 T_1-domain transfer

A network trained on the same type of images that it will later be tested on should outperform networks trained and tested on different types of images. However, this performance difference should diminish following successful domain transfer between the training and testing domains. To test this prediction, we first investigated generalization performance for implicit domain transfer (i.e., without fine tuning) in a single-coil setting. The training domain contained natural images from the ImageNet database or T_2-weighted images, and the testing domain contained T_1-weighted images. Fig. 3.6 displays reconstructions of an undersampled T_1-weighted acquisition via the ImageNet-trained, T_2-trained and T_1-trained networks for R=4. As expected, the T_1-trained network yields sharper and more accurate reconstructions compared to the raw ImageNettrained and T_2-trained networks. Next, we examined explicit domain transfer where ImageNet-trained and T_2-trained networks were fine-tuned. In this case, all networks yielded visually similar reconstructions. Furthermore, when compared against conventional compressive sensing (CS), all network models yielded superior performance. Fig. 3.7 displays reconstructions of an undersampled T_1-weighted acquisition via the ImageNet-trained, T_2-trained and T_1-trained networks, and CS for R=4. The ImageNet-trained network produces images of similar visual quality to other networks and it outperforms CS in terms of image sharpness and residual aliasing

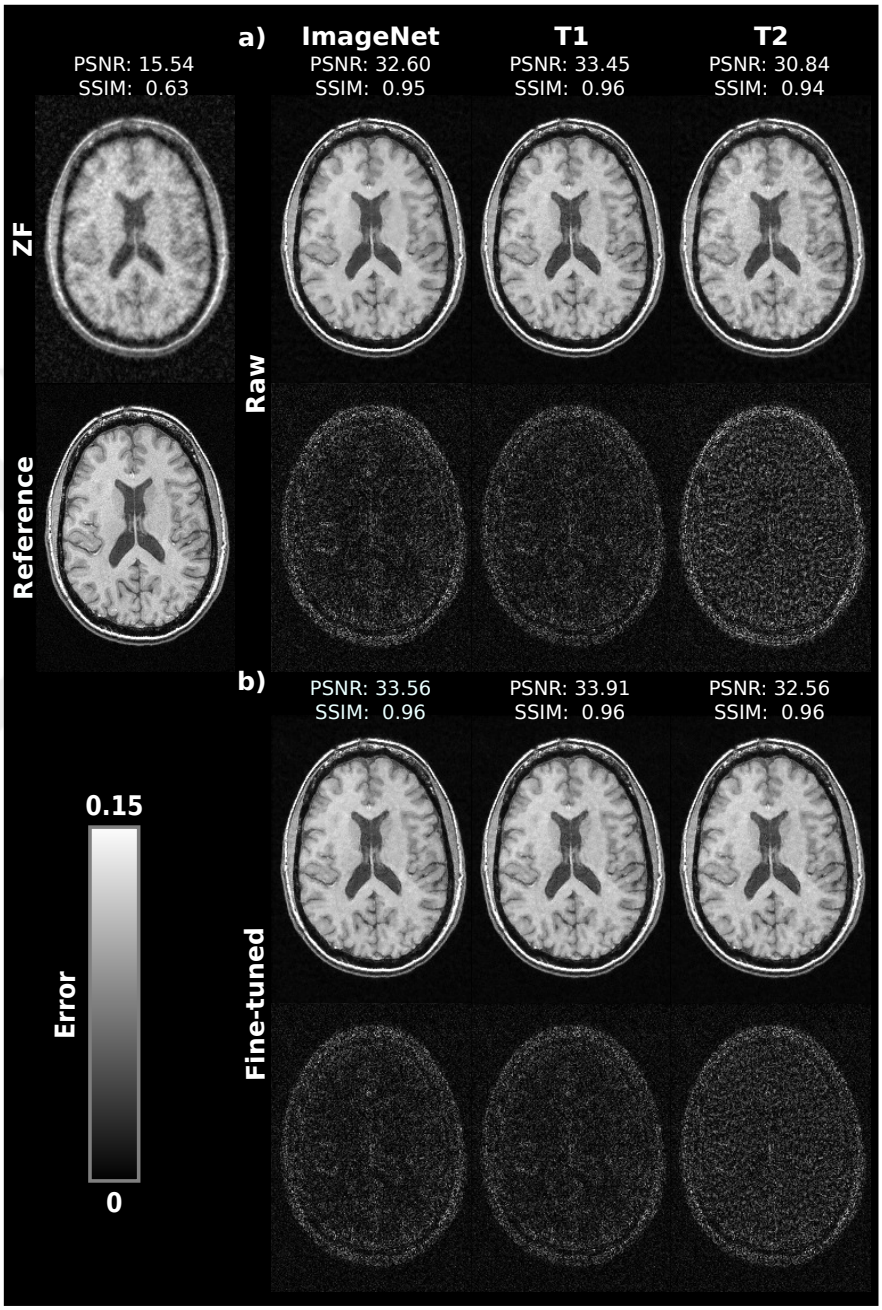

Figure 3.6: Representative reconstructions of a T_1-weighted acquisition at acceleration factor R=4. Reconstructions were performed via the Zero-filled Fourier method (ZF), and ImageNet-trained, T_2-trained, and T_1-trained networks. (a) Reconstructed images and error maps for raw networks (see colorbar). (b) Reconstructed images and error maps for fine-tuned networks. The fully-sampled reference image is also shown. Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_1-weighted images. Following fine-tuning with few tens of samples, ImageNet-trained and T_2-trained networks yield reconstructions of highly similar quality to the T_1-trained network.

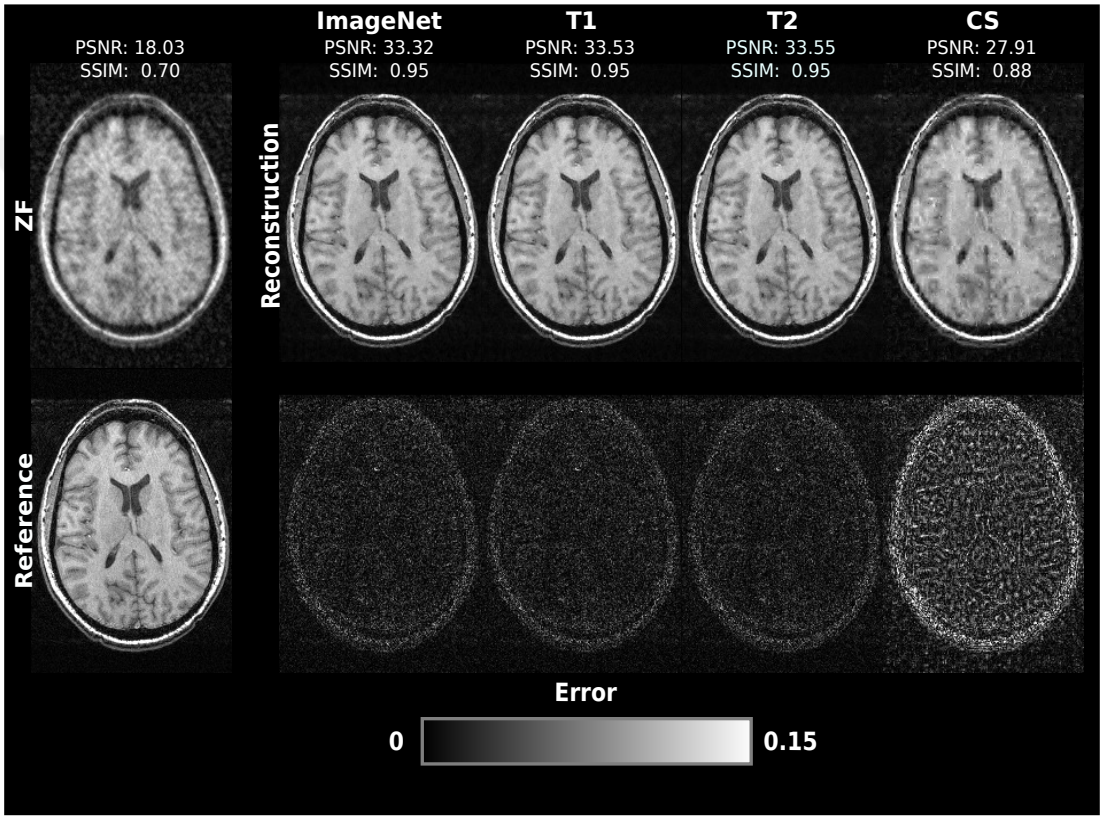

Figure 3.7: Reconstructions of a T_1-weighted acquisition with R=4 via ZF, conventional compressed-sensing (CS), and ImageNet-trained, T_1-trained and T_2-trained networks along with the fully-sampled reference image. Error maps for each reconstruction are shown below (see colorbar). Networks were trained on 2000 images and fine-tuned on 20 images acquired with the test contrast. The domain-transferred networks maintain nearly identical performance to the networks trained directly in the testing domain. Furthermore, the domain-transferred networks reconstructions outperform conventional CS in terms of image sharpness and residual aliasing artifacts.

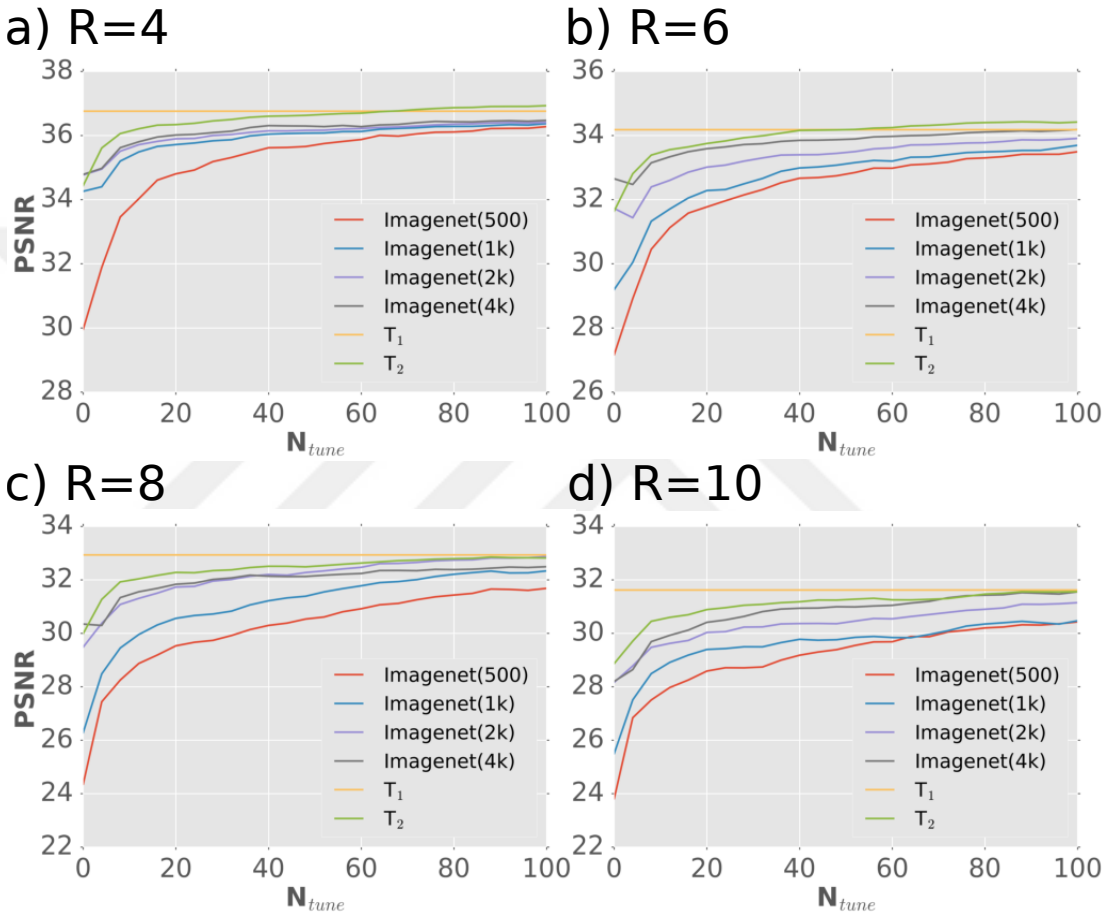

Figure 3.8: Reconstruction performance was evaluated for undersampled T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained on 4k images and fine-tuned on 100 images), ImageNet-trained networks (trained on 500, 1000, 2000, or 4000 images), and T_2-trained network (trained on 4000 images). Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms all domain-transferred networks. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. Domain-transferred networks trained on fewer samples require more fine-tuning samples to yield similar performance consistently across R.

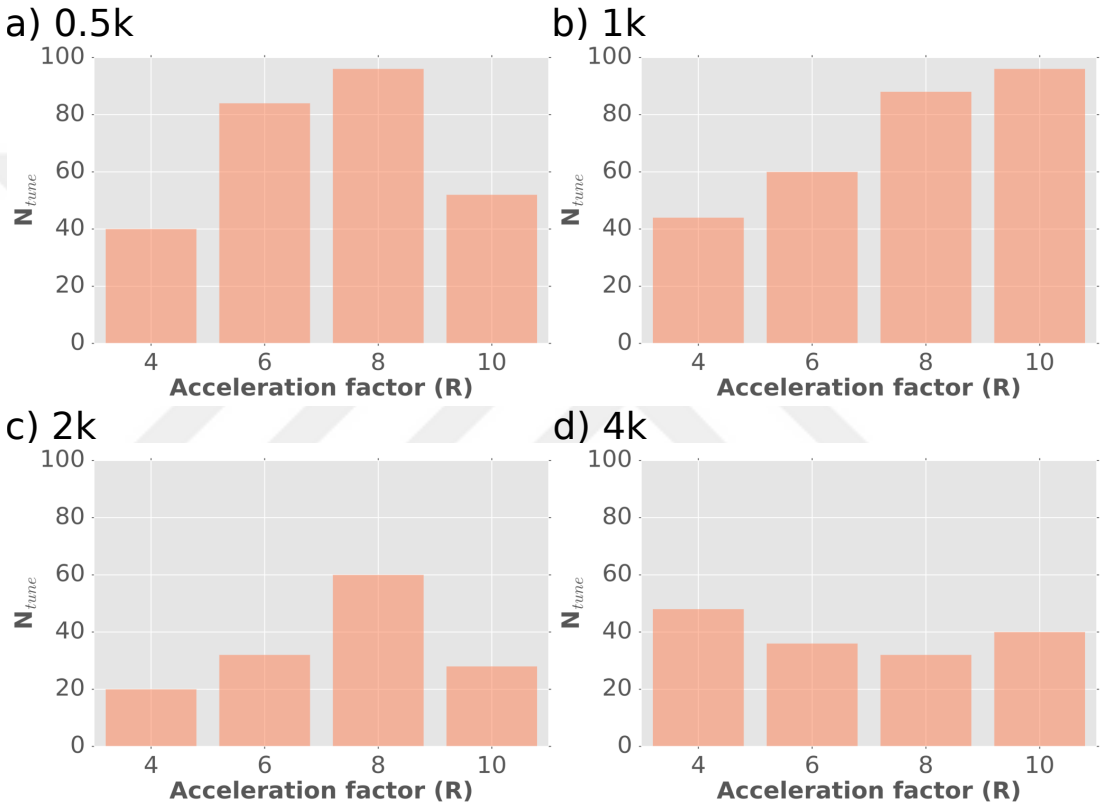

Figure 3.9: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil magnitude images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained networks trained on (a) 500, (b) 1000, (c) 2000, and (d) 4000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network. Domain-transferred networks trained on fewer samples require more fine-tuning samples for the PSNR values to converge. Furthermore, at higher values of R, more fine-tuning samples are required for convergence.

artifacts.

Reconstruction performance of domain-transferred networks may depend on the sizes of both training and fine-tuning sets. To examine interactions between the number of training (N_{train}) and fine-tuning (N_{tune}) samples, we trained networks using training sets in the range [500 4000] and fine-tuning sets in the range [0 100]. Fig. 3.8 shows average PSNR values for a reference T_1-trained network trained on 4000 and fine-tuned on 100 images, and domain-transferred networks for R=4-10. Without fine-tuning, the T_1-trained network outperforms both domain-transferred networks. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. Consistently across R, domain-transferred networks trained on smaller training sets require more fine-tuning samples to yield similar performance.

Fig. 3.9 displays the number of fine-tuning samples required for the PSNR values for ImageNet-trained networks to converge for R=4-10. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing number of fine-tuning samples fell below 0.05% of PSNR for the T_1-trained network. Across R, networks trained on fewer samples require more fine-tuning samples for convergence. However, the required number of fine-tuning samples is greater for higher R. Averaged across R, N_{tune}=68 for N_{train}=500, N_{tune}=72 for N_{train}=1000, N_{tune}=35 for N_{train}=2000, N_{tune}=38 for N_{train}=4000.

To corroborate the visual observations, reconstruction performance was quantitively assessed for both implicit and explicit domain transfer across R=4-10. PSNR and SSIM measurements across the test set are listed in Table 3.1 and Table 3.2. (For reconstruction performance when N_{tune} is fixed to 100, please refer to Table 3.3.) For implicit domain transfer, the T_1-trained networks outperform domain-transferred networks and CS consistently across all R. For explicit domain transfer, the differences between the T_1-trained and domain-transferred networks diminish. Following fine-tuning, the average differences in (PSNR, SSIM) across R between ImageNet and T_1-trained networks diminish from (1.61dB, 1.50%) to (0.35dB, 0.50%), and difference between T_2-trained and T_1-trained networks diminish from (1.96dB, 2.50%) to (0.20dB, 0.25%). Furthermore, the

Table 3.1: Quality of recovered single-coil magnitude T_1-weighted images

		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=4	**Raw**	34.92 ±3.57	0.96 ± 0.02	36.01 ± 3.17	0.97 ± 0.02	34.11 ± 3.64	0.95 ± 0.03
	Tuned	36.06 ± 3.10	0.97 ± 0.01	36.47 ± 3.27	0.97 ± 0.01	36.22 ± 3.13	0.97 ± 0.02
	Limited						
	PSNR	SSIM					
	29.79 ± 3.95	0.93 ± 0.03					
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=6	**Raw**	31.97 ± 3.33	0.94 ± 0.02	33.49 ± 3.29	0.95 ± 0.02	31.27 ± 3.78	0.93 ± 0.04
	Tuned	33.53 ± 3.02	0.95 ± 0.02	33.99 ± 3.29	0.96 ± 0.02	33.71 ± 3.15	0.96 ± 0.02
	Limited						
	PSNR	SSIM					
	27.02 ± 4.35	0.90 ± 0.05					
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=8	**Raw**	29.78 ± 3.75	0.92 ± 0.03	31.64 ± 3.37	0.94 ± 0.03	29.72 ± 3.75	0.91 ± 0.05
	Tuned	32.56 ± 3.15	0.95 ± 0.02	32.32 ± 3.36	0.95 ± 0.03	32.39 ± 3.45	0.95 ± 0.02
	Limited						
	PSNR	SSIM					
	24.02 ± 4.46	0.85 ± 0.07					
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=10	**Raw**	28.52 ± 3.86	0.91 ± 0.04	30.50 ± 3.37	0.93 ± 0.03	28.70 ± 3.70	0.90 ± 0.05
	Tuned	30.37 ± 3.34	0.93 ± 0.03	31.12 ± 3.38	0.94 ± 0.03	30.78 ± 3.14	0.93 ± 0.03
	Limited						
	PSNR	SSIM					
	21.45 ± 4.52	0.79 ± 0.08					

Reconstruction quality for single-coil magnitude T_1-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained, T_1-trained, T_2-trained and limited networks. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_1-weighted images (tuned).

domain-transferred networks outperform CS consistently across R, by an average of 3.70dB PSNR and 6.13% SSIM, and limited networks by an average of 7.63dB PSNR and 8.38% SSIM.

3.3.1.2 T_2-domain transfer

Next, we repeated the analyses for implicit and explicit domain transfer when the testing domain contained T_2-weighted images. Fig. 3.10 displays reconstructions of an undersampled T_2-weighted acquisition via the ImageNet-, T_1- and T_2-trained networks for acceleration factor R=4. Again, the network trained directly

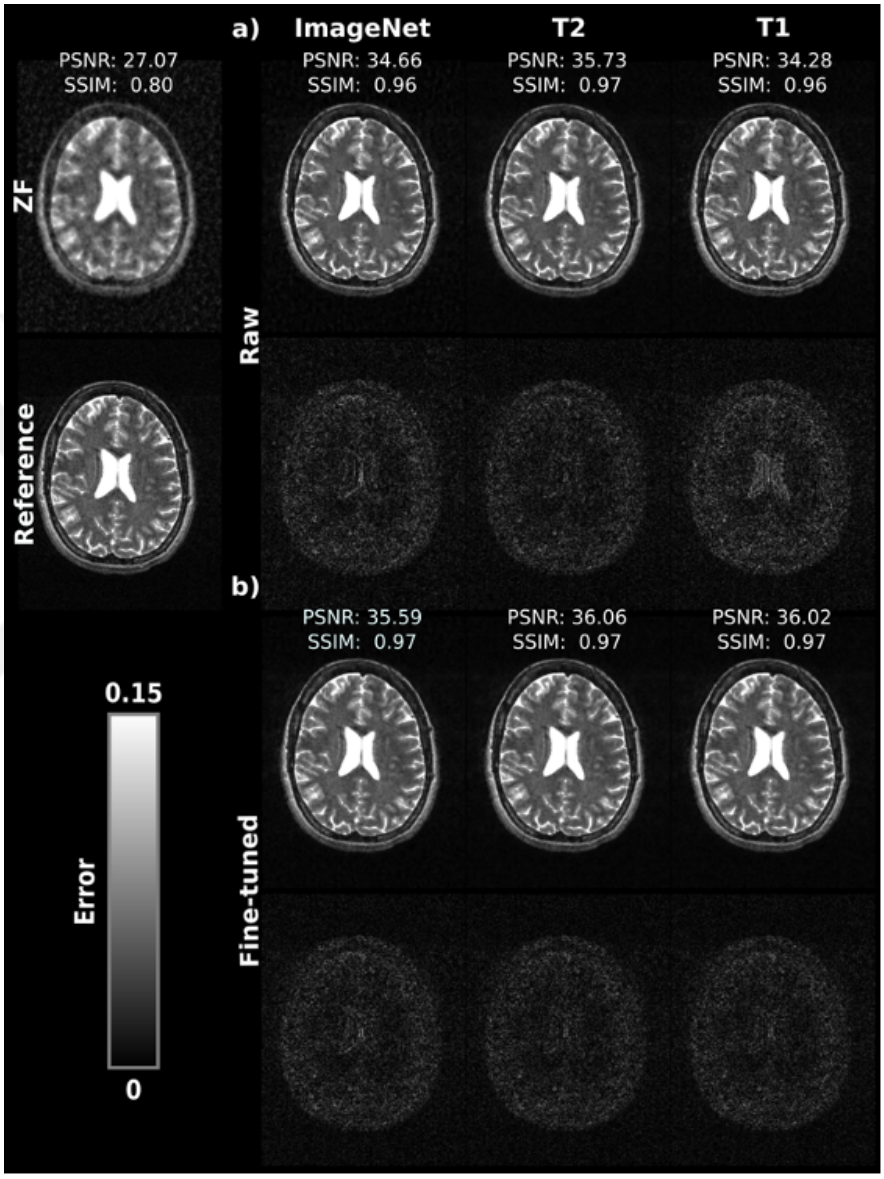

Figure 3.10: Representative reconstructions of a T_2-weighted acquisition at acceleration factor R=4. Reconstructions were performed via the Zero-filled Fourier method (ZF), and ImageNet-trained, T_2-trained, and T_1-trained networks. (a) Reconstructed images and error maps for raw networks (see colorbar). (b) Reconstructed images and error maps for fine-tuned networks. The fully-sampled reference image is also shown. Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_2-weighted images. Following fine-tuning with few tens of samples, ImageNet-trained and T_1-trained networks yield reconstructions of highly similar quality to the T_2-trained network.

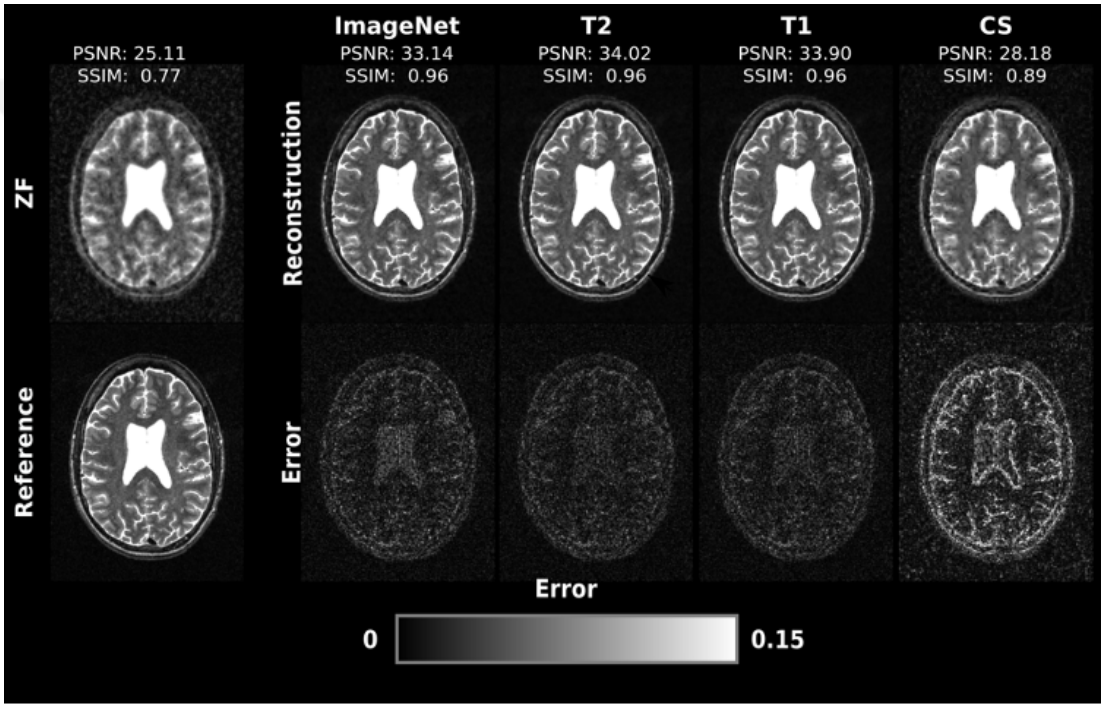

Figure 3.11: Reconstructions of a T_2-weighted acquisition with R=4 via ZF, conventional compressed-sensing (CS), and ImageNet-trained, T_1-trained and T_2-trained networks along with the fully-sampled reference image. Error maps for each reconstruction are shown below (see colorbar). Networks were trained on 2000 images and fine-tuned on 20 images acquired with the test contrast. The domain-transferred networks maintain nearly identical performance to the networks trained directly in the testing domain. Furthermore, the domain-transferred networks reconstructions outperform conventional CS in terms of image sharpness and residual aliasing artifacts.

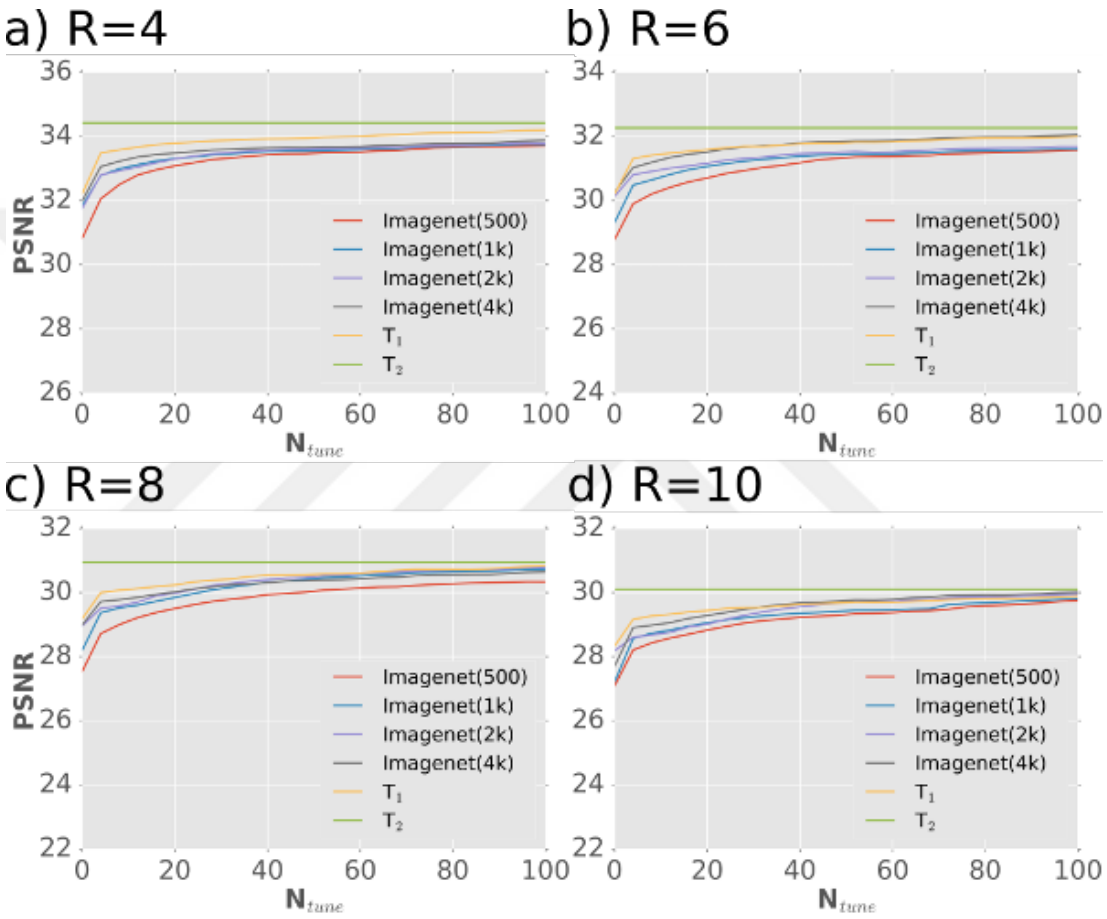

Figure 3.12: Reconstruction performance was evaluated for undersampled T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained on 4k images and fine-tuned on 100 images), ImageNet-trained networks (trained on 500, 1000, 2000, or 4000 images), and T_1-trained network (trained on 4000 images). Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level. Domain-transferred networks trained on fewer samples require more fine-tuning samples to yield similar performance consistently across R.

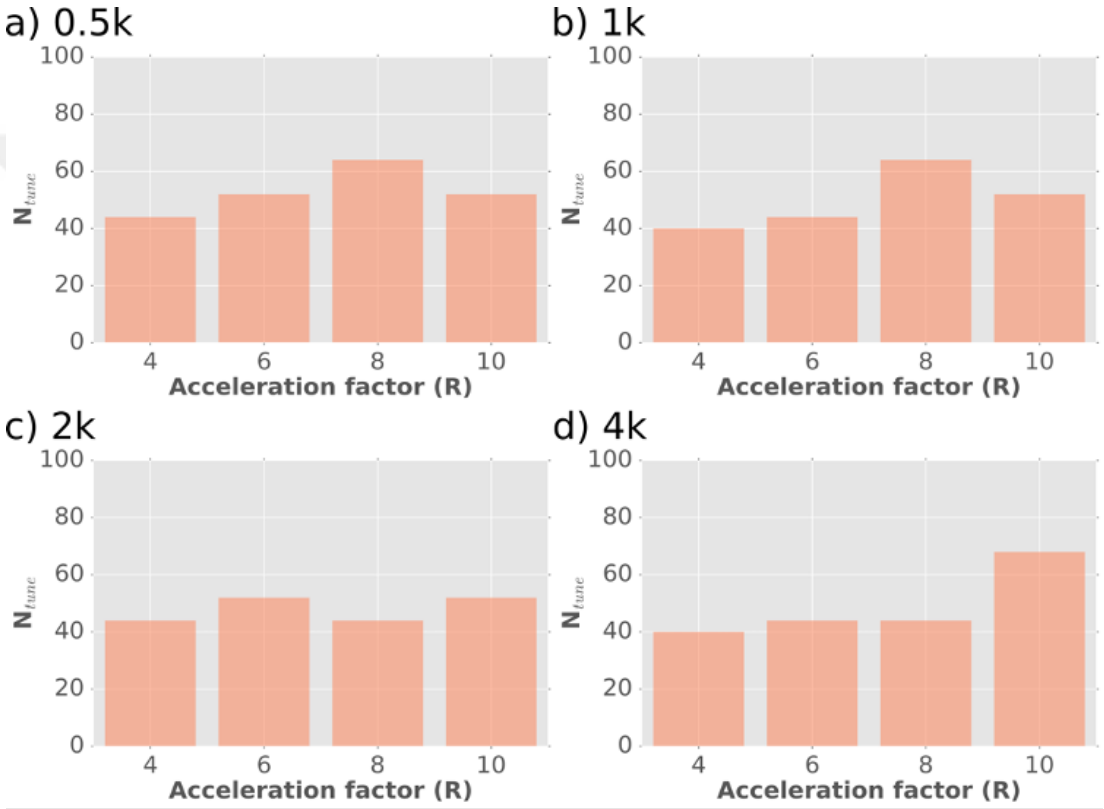

Figure 3.13: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained networks trained on (a) 500, (b) 1000, (c) 2000, and (d) 4000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network. Domain-transferred networks trained on fewer samples require more fine-tuning samples for the PSNR values to converge. Furthermore, at higher values of R, more fine-tuning samples are required for convergence.

in the testing domain (T_2-weighted) outperforms domain-transferred networks. After fine tuning with as few as 20 images, the domain-transferred networks yield visually similar reconstructions to the T_2-trained network. Fig. 3.11 displays reconstructions of an undersampled T_2-weighted acquisition via the ImageNet-trained, T_2-trained and T_1-trained networks, and CS for R=4. The ImageNet-trained network produces images of similar visual quality to other networks and it outperforms CS in terms of image sharpness and residual aliasing artifacts.

We also examined interactions between the number of training and fine-tuning samples when the target domain contained T_2-weighted images. Fig. 3.12 shows average PSNR values for a reference T_2-trained network trained on 4000 and fine-tuned on 100 images, and domain-transferred networks for R=4-10. Compared to the case of T_1-weighted images, interaction between number of training and fine-tuning samples is weaker. Yet, a greater number of fine-tuning samples is still required for reconstructions at higher R. Fig. 3.13 displays the number of fine-tuning samples required for convergence of ImageNet-trained networks. Averaged across R=4-10, N_{tune}=53 for N_{train}=500, N_{tune}=50 for N_{train}=1000, N_{tune}=48 for N_{train}=2000, N_{tune}=49 for N_{train}=4000.

PSNR and SSIM measurements on T_2-weighted reconstructions across the test set are listed in Table 3.4 (For results using N_{tune}=100, please refer to Table 3.5). Following fine-tuning, average (PSNR, SSIM) differences between ImageNet and T_2-trained networks diminish from (1.23dB, 2.00%) to (0.29dB, 0.50%), and difference between T_1-trained and T_2-trained networks diminish from (0.67dB, 1.25%) to (0.15dB, 0.50%). Across R, the domain-transferred networks also outperform CS by 5.20dB PSNR and 8.50% SSIM, and limited networks by 2.67dB PSNR and 1.50% SSIM.

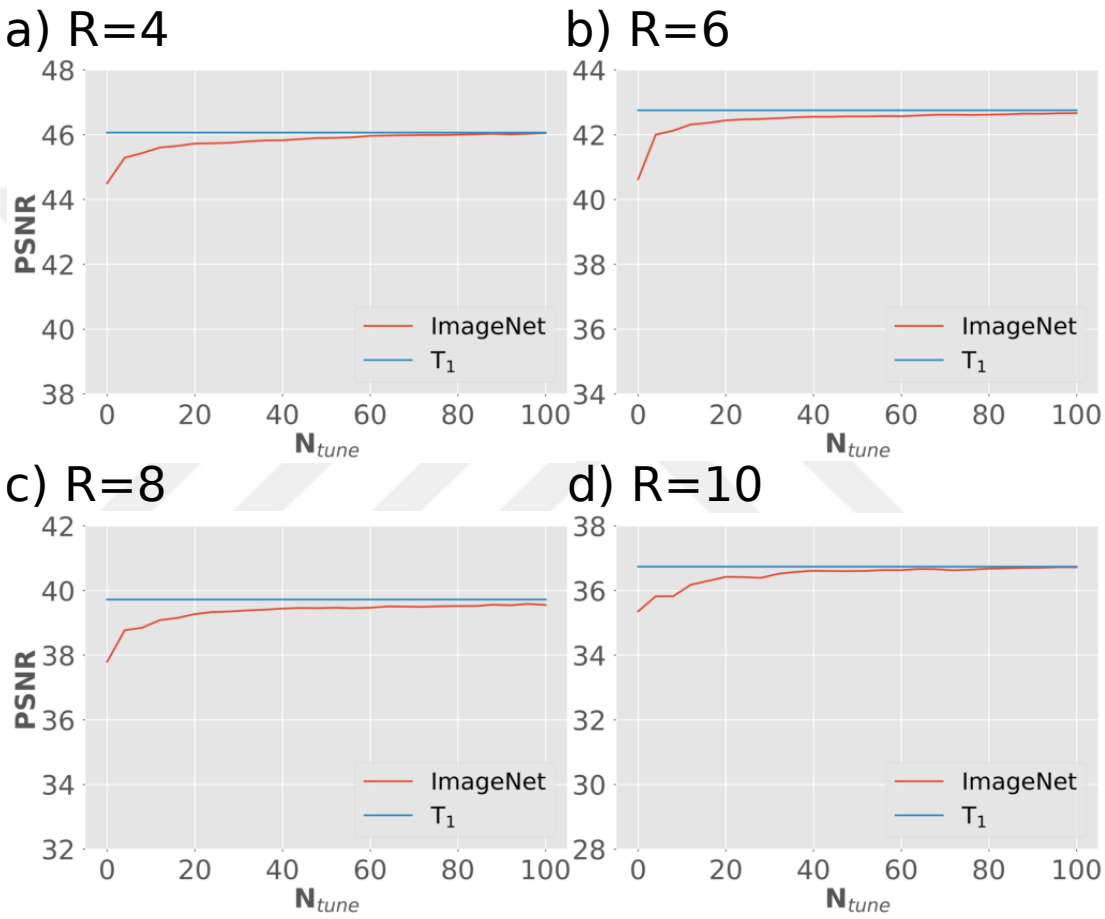

Figure 3.14: Reconstruction performance was evaluated for undersampled multi-coil T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level.

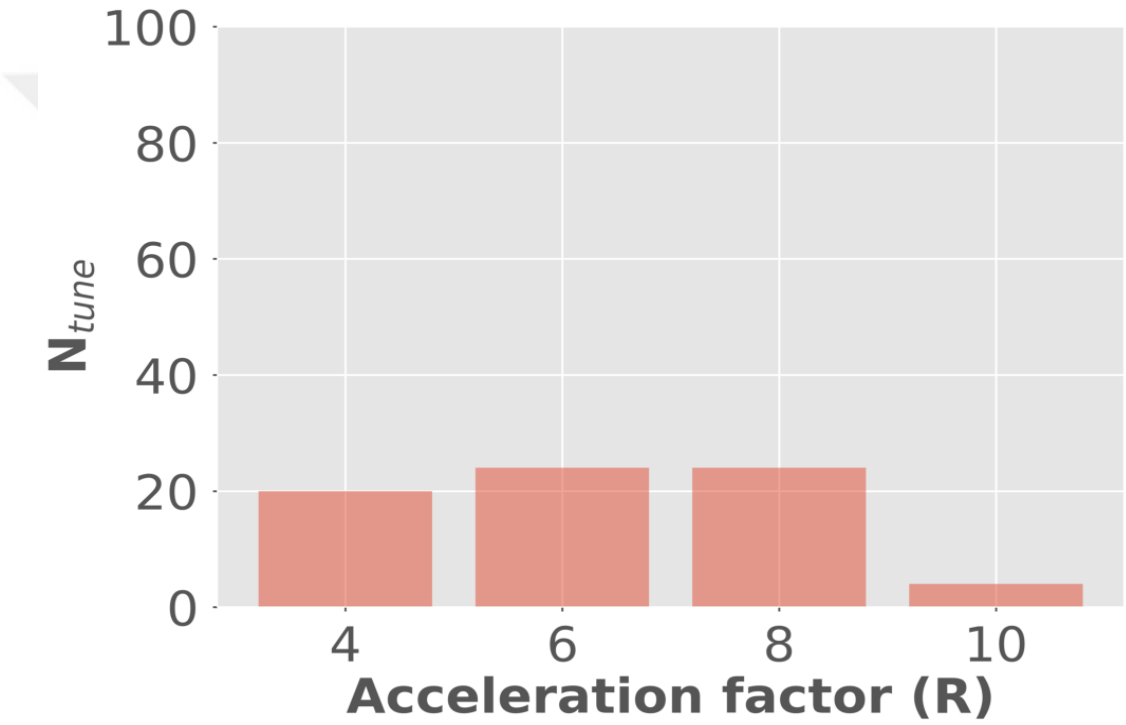

Figure 3.15: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on multi-coil complex images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network (see Figure 6). At higher values of R, more fine-tuning samples are required for convergence.

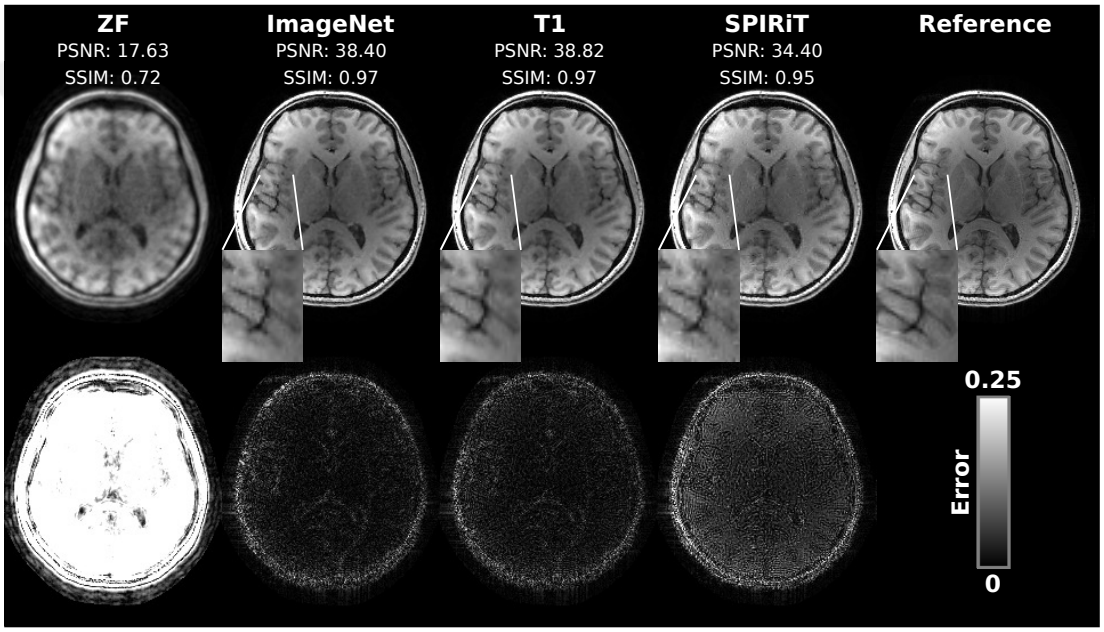

Figure 3.16: Representative reconstructions of a multi-coil T_1-weighted acquisition at acceleration factor R=10. Reconstructions were performed via ZF, ImageNet-trained and T_1-trained networks, and SPIRiT (top row). Corresponding error maps are also shown (see colorbar; bottom row) along with the fully-sampled reference (top row). Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_1-weighted images. The ImageNet-trained network maintains similar performance to the T_1-trained network trained directly on the images from the test domain. Furthermore, the domain-transferred network outperforms conventional SPIRiT in terms of residual aliasing artifacts.

3.3.2 Multi-coil complex data

3.3.2.1 T_1-domain transfer

Next, we demonstrated the proposed approach on multi-coil T_1-weighted images. We compared ImageNet- and T_1-trained networks at R=4-10. Fig. 3.14 displays average PSNR values for the T_1-trained network (trained and fine-tuned on 360 images) and ImageNet-trained network (N_{train}=2000 and $N_{end-to-end}$=360 multi-coil natural images, and $N_{tune} \in$[0,100] T_1-weighted images). As N_{tune} increases, the PSNR differences between T_1- and ImageNet-trained networks start diminishing. Fig. 3.15 displays the number of fine-tuning samples required for the PSNR values for ImageNet-trained networks to converge. Averaged across R=4-10, ImageNet-trained networks require N_{tune}=18 for convergence. We also compared the proposed transfer learning approach with L1-regularized SPIRiT. Fig. 3.16 shows representative reconstructions obtained via the ImageNet-trained network, T_1-trained network and SPIRiT for R=10. The ImageNet-trained network produces images of similar visual quality to the T_1-trained network, and it outperforms SPIRiT in terms of residual aliasing artifacts.

Quantitative assessment of multi-coil reconstructions for the ImageNet-trained network, T_1-trained network and SPIRiT across R=4-10 are listed in Table 3.6. For implicit domain transfer, the T_1-trained network performs better than the ImageNet-trained network. Following fine-tuning, the average differences in (PSNR, SSIM) across R between ImageNet and T_1-trained networks diminish from (2.10dB, 1.43%) to (0.62dB, 0.15%). Furthermore, the ImageNet-trained network outperforms SPIRiT in all cases. On average across R, the ImageNet-trained network improves performance over SPIRiT by 0.93dB PSNR and 0.60

3.3.2.2 T_2-domain transfer

We also demonstrated the proposed approach when the testing domain contained multi-coil T_2-weighted images. Fig. 3.17 displays average PSNR values

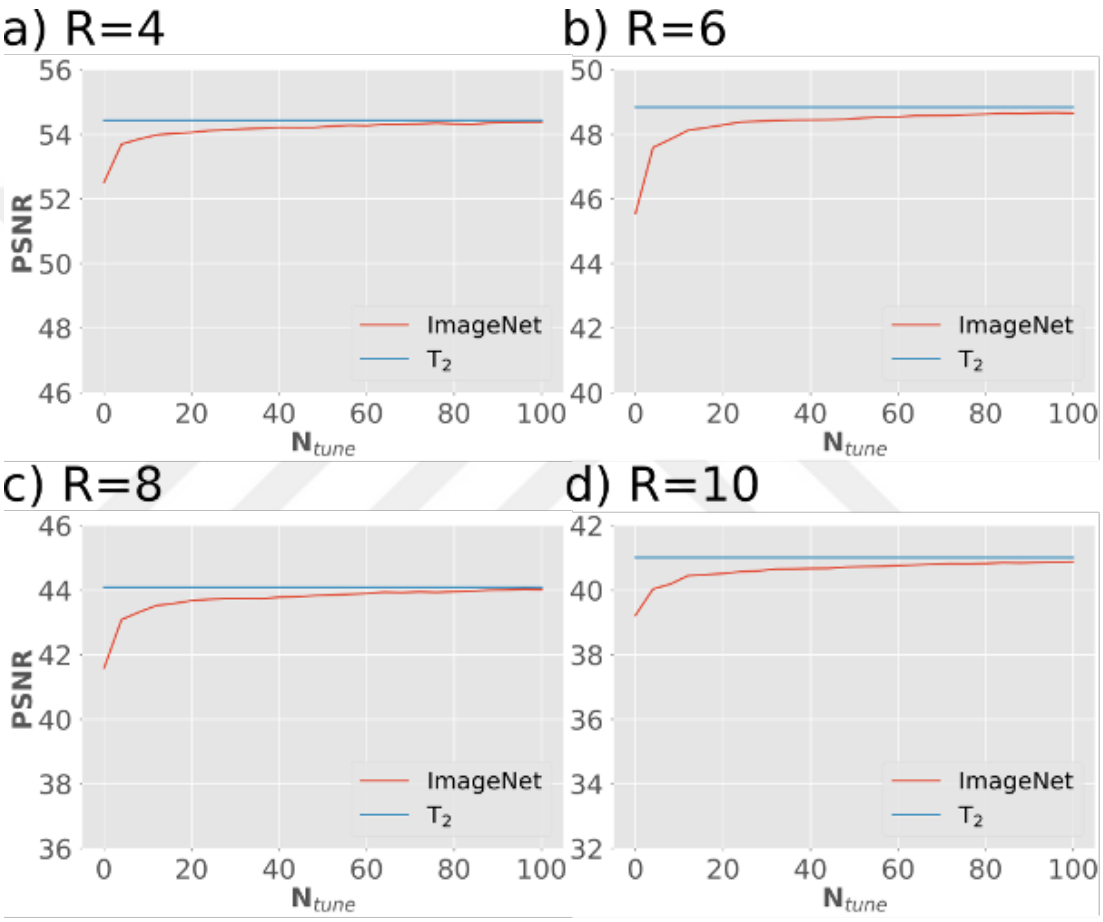

Figure 3.17: Reconstruction performance was evaluated for undersampled multi-coil T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_2-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level.

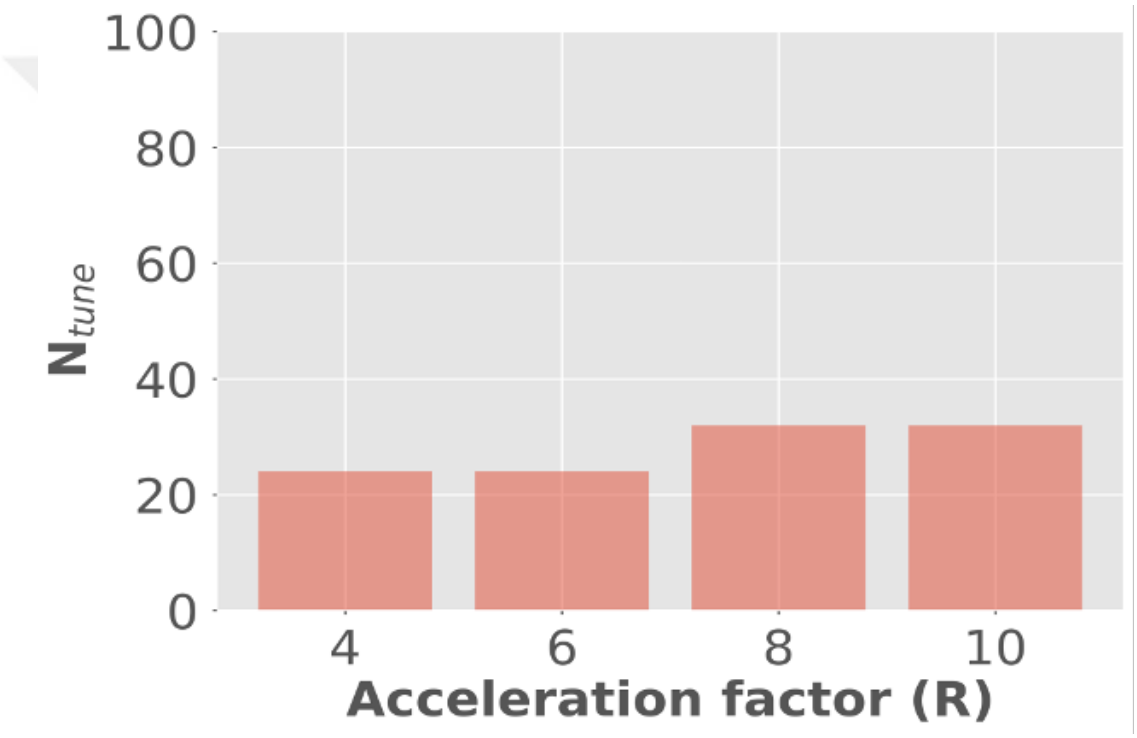

Figure 3.18: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on multi-coil complex images) to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network. At higher values of R, more fine-tuning samples are required for convergence.

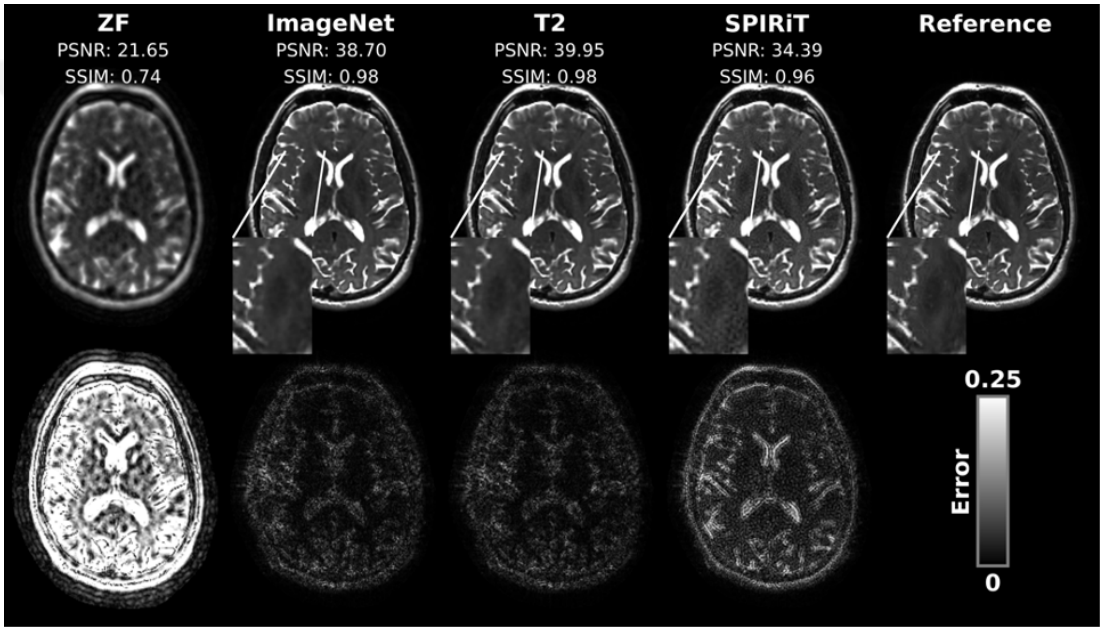

Figure 3.19: Representative reconstructions of a multi-coil T_2-weighted acquisition at acceleration factor R=10. Reconstructions were performed via ZF, ImageNet-trained and T_2-trained networks, and SPIRiT (top row). Corresponding error maps are also shown (see colorbar; bottom row) along with the fully-sampled reference (top row). Network training was performed on a training dataset of 2000 images and fine-tuned on a sample of 20 T_2-weighted images. The ImageNet-trained network maintains similar performance to the T_2-trained network trained directly on the images from the test domain. Furthermore, the domain-transferred network outperforms conventional SPIRiT in terms of residual aliasing artifacts.

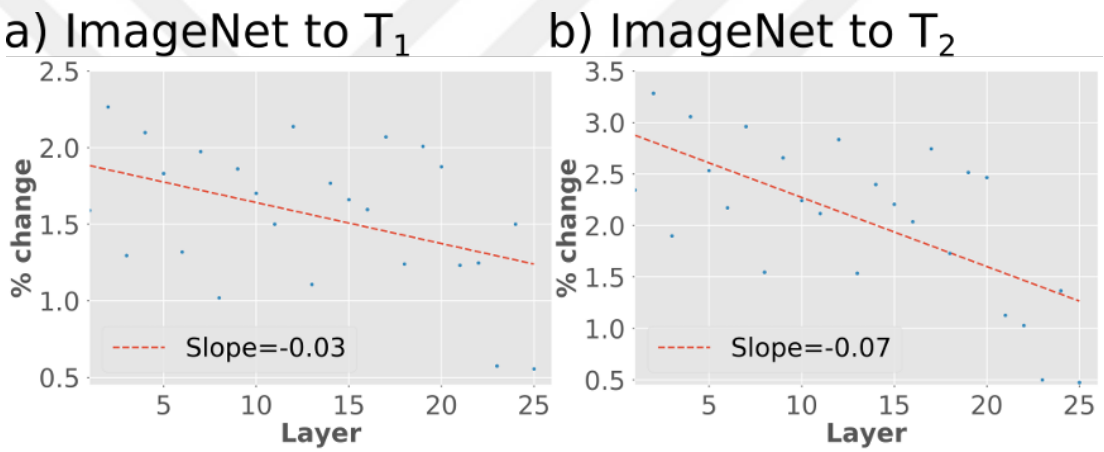

Figure 3.20: Percentage change in network weights as a function of network depth for multi-coil ImageNet to (a) T_1 and (b) T_2 domain transfer averaged across acceleration factors (R=4-10). Red dots correspond to the percentage change, and blue dashed lines correspond to a linear least squares fit to the percentage change. Overall, the percentage change in weights is higher for earlier versus later layers of the network. For ImageNet to T_1 domain transfer, percentage change varies from 2.27% to 0.56%, and for ImageNet to T_2 domain transfer percentage change varies from 3.28% to 0.47%. Note that the layer number ranges from 1 to 25. This is because the CNN architecture used in this study consists of 5 cascades and each cascade consists of 5 layers.

for the T_2-trained network (trained and fine-tuned on 360 images) and ImageNet-trained network (N_{train}=2000 and $N_{end-to-end}$=360 multi-coil natural images, and N_{tune} ∈[0,100] T_2-weighted images). Fig. 3.18 displays the number of fine-tuning samples required for the ImageNet-trained network to converge. Averaged across R, ImageNet-trained networks require N_{tune}=28 for convergence. Fig. 3.19 shows representative reconstructions obtained via the ImageNet-trained network, T_2-trained network and SPIRiT for R=10. The ImageNet-trained network produces images of similar visual quality to the T_2-trained network, while outperforming SPIRiT in terms of residual artifacts. Meanwhile, Table 3.7 lists quantitative assessments of reconstruction quality across R=4-10. Following fine-tuning, the average differences in (PSNR, SSIM) between ImageNet and T_2-trained networks diminish from (2.33dB, 0.70%) to (0.37dB, 0.05%). On average across R, the ImageNet-trained network improves performance over SPIRiT by 1.53dB PSNR and 1.07% SSIM.

We also computed percentage change in coefficients of convolution kernels in CNN layers for R=4-10. Fig. 3.20 demonstrates percentage change in network weights as a function of network depth for multi-coil ImageNet to T_1 and T_2 domain transfer, averaged across R. Overall, the percentage change in weights is higher for earlier versus later layers of the network. For ImageNet to T_1 domain transfer, percentage change varies from 2.27% to 0.56%, and for ImageNet to T_2 domain transfer percentage change varies from 3.28% to 0.47%. The difference in the level of weight change across layers can be attributed to the level of residual artifacts present in inputs to each layer. Since the inputs to earlier layers contain more domain-specific residual artifacts, they might undergo greater change during fine-tuning compared to later layers.

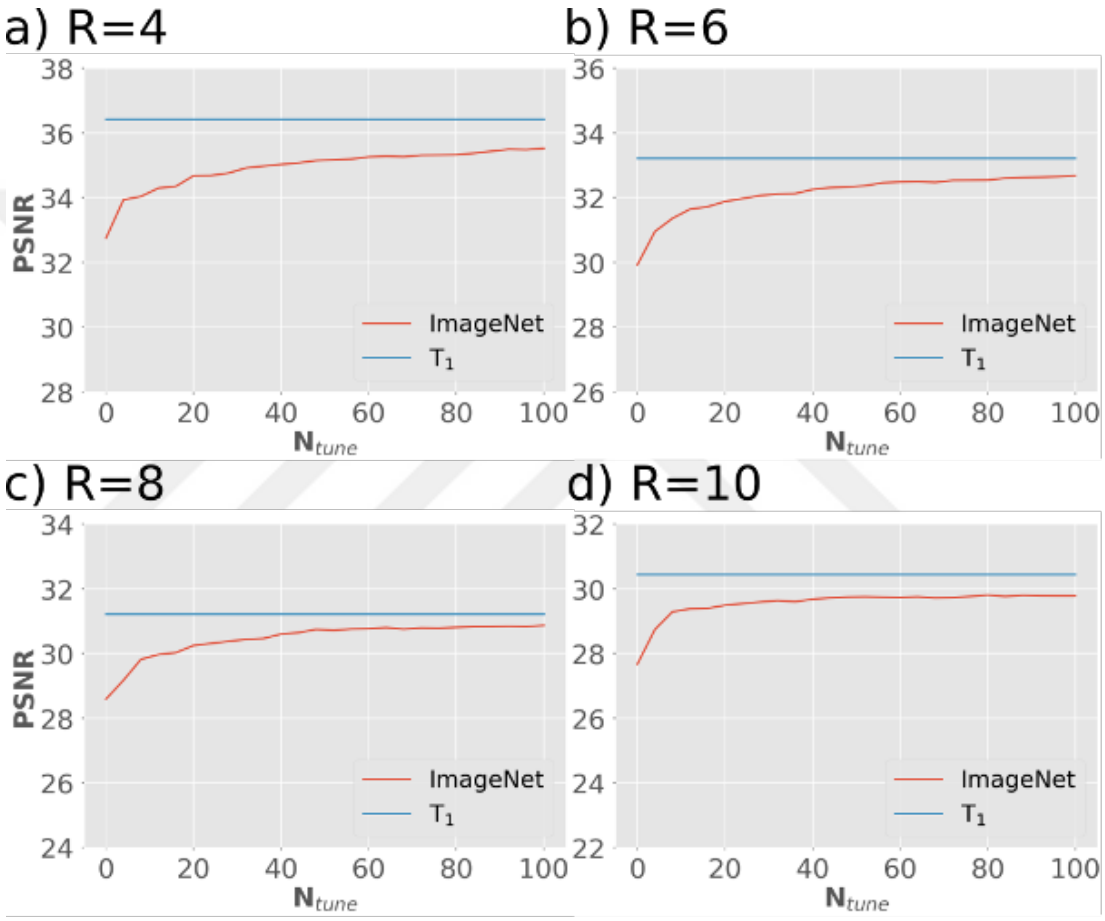

Figure 3.21: Reconstruction performance was evaluated for undersampled single-coil complex T_1-weighted acquisitions. Average PSNR values across T_1-weighted validation images were measured for the T_1-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_1-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level.

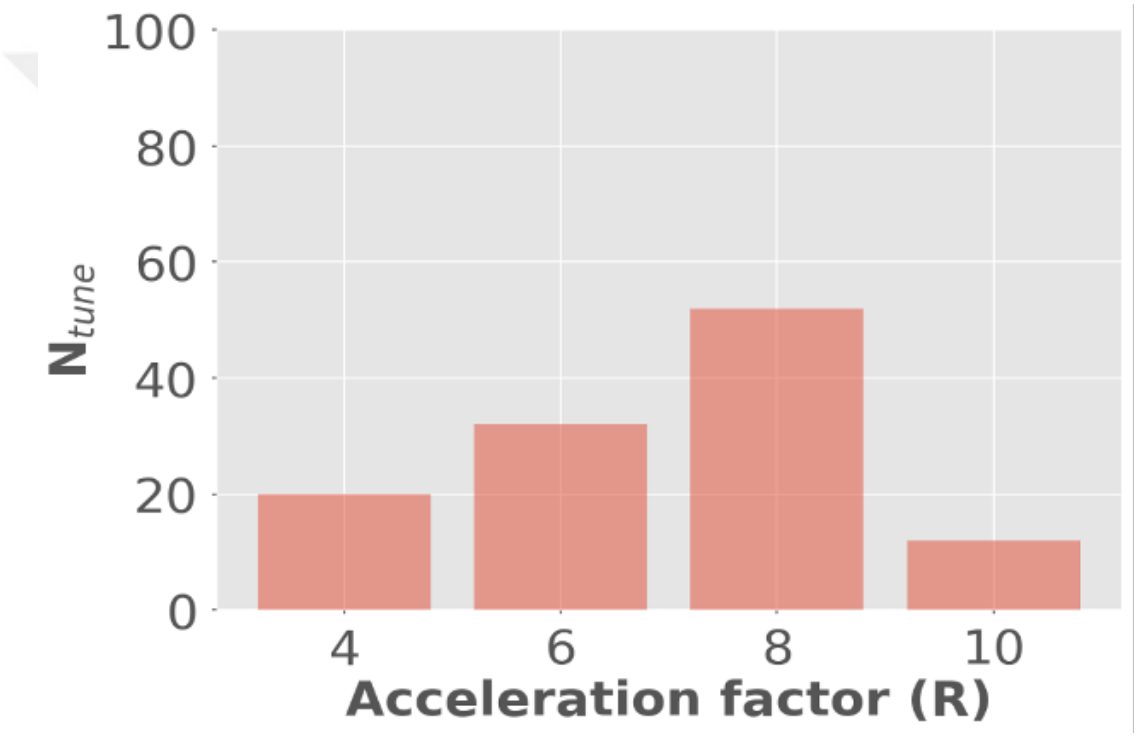

Figure 3.22: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil complex images) to converge. Average PSNR values across T_1-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_1-trained network.

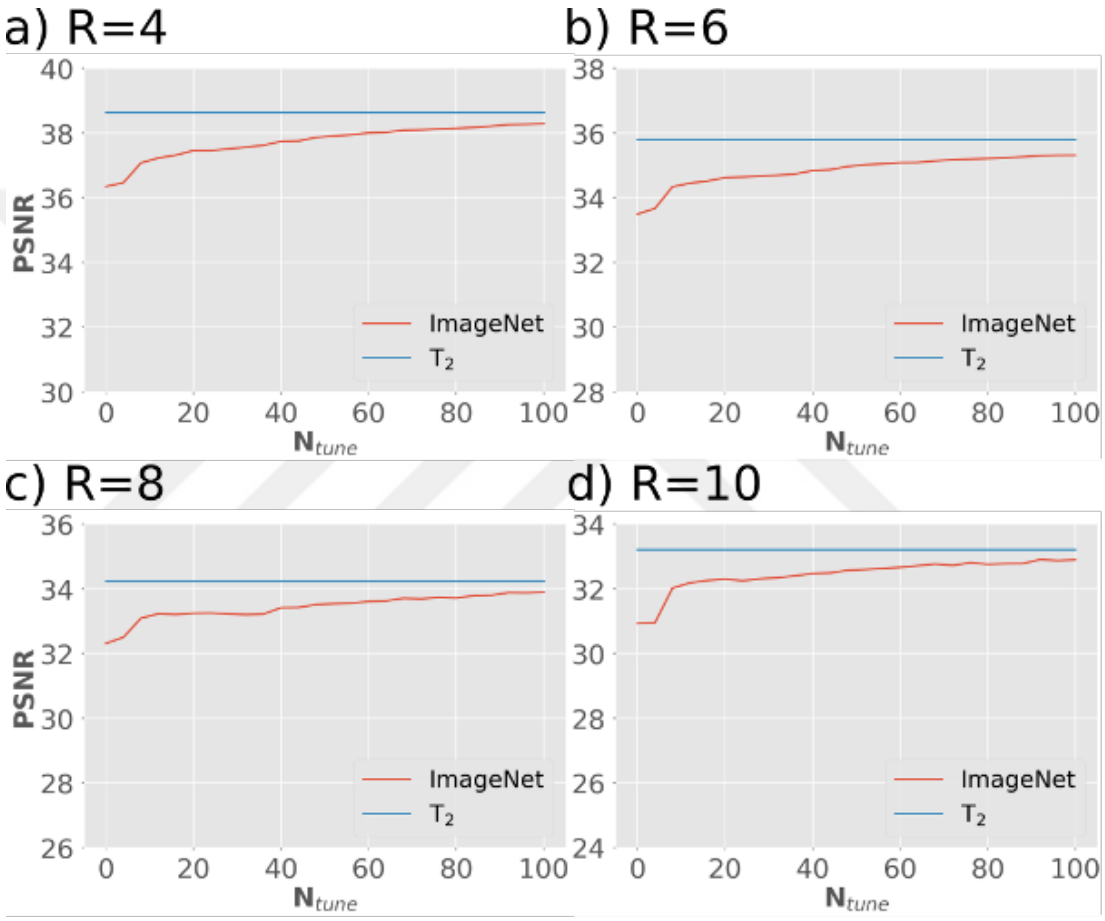

Figure 3.23: Reconstruction performance was evaluated for undersampled single-coil complex T_2-weighted acquisitions. Average PSNR values across T_2-weighted validation images were measured for the T_2-trained network (trained and fine-tuned on 360 images), and ImageNet-trained network trained on 2000 images. Results are plotted as a function of number of fine-tuning samples for acceleration factors (a) R= 4, (b) R = 6, (c) R = 8, and (d) R= 10. Without fine-tuning, the T_2-trained network outperforms the domain-transferred network. As the number of fine-tuning samples increases, the PSNR differences decay gradually to a negligible level.

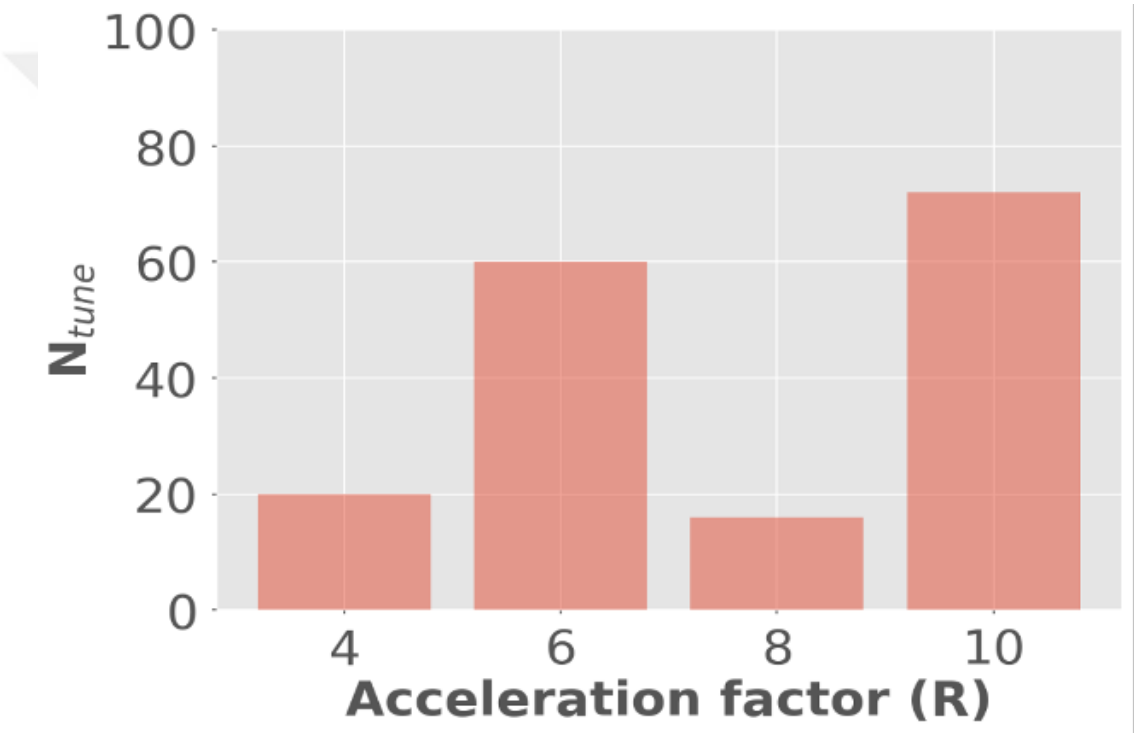

Figure 3.24: Number of fine-tuning samples required for the PSNR values for ImageNet-trained networks (trained on single-coil complex images) to converge. Average PSNR values across T_2-weighted validation images were measured for the ImageNet-trained network trained on 2000 images. Convergence was taken as the number of fine-tuning samples where the percentage change in PSNR by incrementing N_{tune} fell below 0.05% of the average PSNR for the T_2-trained network.

3.3.3 Single-coil complex data

3.3.3.1 T_1-domain transfer

Next, we demonstrated the proposed approach on single-coil complex T_1-weighted images, obtained by combining multi-coil images via coil sensitivity maps estimated using ESPIRiT. Fig. 3.21 displays average PSNR values for the T_1-trained network (trained and fine-tuned on 360 images) and ImageNet-trained network (N_{train}=2000 and $N_{end-to-end}$=360 single-coil natural images, and $N_{tune} \in$[0,100] T_1-weighted images); and Fig. 3.22 displays the number of fine-tuning samples required for the ImageNet-trained network. Averaged across R, ImageNet-trained networks require N_{tune}=29 for convergence. Quantitative assessment for the ImageNet-trained network and T_1-trained network across R=4-10 are listed in Table 3.8. Following fine-tuning, the average differences in (PSNR, SSIM) across R between ImageNet and T_1-trained networks diminish from (4.23dB, 5.95%) to (1.65dB, 1.25%).

3.3.3.2 T_2-domain transfer

Finally, we demonstrated the proposed approach when the testing domain contained single-coil complex T_2-weighted images. Fig. 3.23 displays average PSNR values for the T_2-trained network (trained and fine-tuned on 360 images) and ImageNet-trained network (N_{train}=2000 and $N_{end-to-end}$=360 single-coil natural images, and $N_{tune} \in$[0,100] T_2-weighted images); and Fig. 3.24 displays the number of fine-tuning samples required for the ImageNet-trained network. Averaged across R, ImageNet-trained networks require N_{tune}=42 for convergence. Quantitative assessment for the ImageNet-trained network and T_2-trained network across R=4-10 are listed in Table 3.9. Following fine-tuning, the average differences in (PSNR, SSIM) across R between ImageNet T_2-trained networks diminish from (2.48dB, 2.00%) to (0.79dB, 0.50%).

3.4 Discussion

Neural networks for MRI reconstruction involve many free parameters to be learnt, so an extensive amount of training samples is typically needed [97]. In theory, network performance should be optimized by drawing the training and testing samples from the same domain, acquired under a common MRI protocol. In practice, however, compiling large public datasets can require coordinated efforts among multiple imaging centers, and so such datasets are rare. As an alternative, several recent studies trained neural networks on a collection of multi-contrast images [14]. When needed, data augmentation procedures were used to further expand the training dataset [53, 4]. While these approaches gather more samples for training, it remains unclear how well a network trained on images acquired with a specific type of tissue contrast generalizes to images acquired with different contrasts. Thus, variability in MR contrasts can lead to suboptimal reconstruction performance.

Here, we first questioned the generalizability of neural network models across different contrasts. We find that a network trained on MR images of a given contrast (e.g. T_1-weighted) yields suboptimal reconstructions on images of a different contrast (e.g. T_2-weighted). This confirms that the best strategy is to train and test networks in the same domain. Yet, it may not be always feasible to gather a large collection of images from a desired contrast. To address the problem of data scarcity, we proposed a transfer-learning approach for accelerated MRI. The proposed approach trains neural networks using training samples from a large public dataset of natural images. The network is then fine-tuned end-to-end using only few tens of MR images. Reconstructions obtained via the ImageNet-trained network are of nearly identical quality to reconstructions obtained by networks trained directly in the testing domain using thousands of MR images.

In the current study, we proposed an explicit domain transfer approach, where networks are initially trained using a large number of images in a source domain, and then fine-tuned using fewer samples in the target domain. If the source and target domains were structurally dissimilar, the domain-transferred

networks would not be expected to perform successfully in the target domain. Note that previous studies reported natural and MR images to have similar early-to-intermediate visual features [70], and to have similar energy spectrums in the Fourier domain [1, 98]. Thus, the comparable performance of domain-transferred and target-domain networks here can be attributed to such shared visual features. In T_1 reconstructions, we observed that the ImageNet-trained and T_2-trained networks performed similarly at R= (8, 10), yet the ImageNet-trained network was superior at relatively low R= (4, 6). At high R, high-spatial-frequency samples in the target domain are largely missing, and the networks aim to synthesize missing samples primarily based on low-frequency samples. Thus, the general similarity of energy spectrums between natural and MR images might lead to similar performance for ImageNet- and T_2-trained networks. At low R, however, additional high-frequency information is available in the target domain, and success of implicit domain transfer might rely more critically on the similarity of high-frequency structure between source and target domains. In the datasets reported here, natural and T_1-weighted images have sharper object boundaries, whereas T_2-weighted images have broadened tissue transitions. This difference might have contributed to the superior performance of the ImageNet-trained network at low R. Please also note that, in T_2 reconstructions, T_1-trained networks were superior to ImageNet-trained network at all acceleration factors. These results suggest that T_2-weighted images are structurally closer to T_1-weighted images than to natural images.

Here we demonstrated successful domain transfer from natural images to brain MR images. A future research direction is to examine the success of this approach for domain transfer between MR images of different organs. In these images, the anatomy of interest might occupy different portions of the field of view (FOV) due to inherent shape and size differences among organs. These differences might in turn limit reconstruction performance of domain-transferred networks. That said, our domain-transfer experiments from natural to MR images indicate that such performance loss is not significant. In the single-coil magnitude case, we demonstrated successful domain transfer with ImageNet-trained networks that were trained on natural images spanning across the entire FOV.

This result implies that the proposed transfer learning approach should generalize well between images of different organs. Here the propsoed approach was mainly demonstrated for structural MR imaging. Another possible avenue of the future research can be to apply domain transfer for other applications such as functional [99, 100, 101, 102, 103, 104, 105, 106, 107] and diffusion MRI [108, 109].

An important concern for complex models trained on relatively restricted datasets is overfitting. Several precautions were employed here to minimize potential bias due to overfitting. First, a limited learning rate was used to fine-tune domain-transferred networks in the target domain to limit the range of parameter updates. Second, early stopping based on validation errors were used in both training and fine-tuning phases. Lastly, a cross-validation procedure was used with a three-way split of training/validation and test data, where test data are exclusively reserved for assessment of model performance. Note that the high performance of domain-transferred networks reported here imply that networks are not unduly biased by overfitting.

Another important concern is characteristic failures of domain-transferred networks such as hallucination of features from the source domain. Hallucination poses an important limitation particularly for generative network architectures designed to draw new samples of data from a learned distribution[110, 111]. Unlike generative models, the deterministic CNN architectures are considered less prone to hallucination, and no significant hallucination was observed in the reconstructions reported here. However, reconstruction artifacts were visible for networks under implicit domain transfer, and these artifacts were alleviated following fine-tuning in the target domain.

Several recent studies have considered domain transfer to enhance performance in NN-based MRI reconstruction [15, 16, 13, 70, 112]. A group of studies have aimed to perform implicit domain transfer across MRI contrasts without fine-tuning. One proposed method was to train networks on MR images in a given contrast, and then to directly use the trained networks on images of different contrasts [13]. While this method yields successful reconstructions, our results

suggest that network performance can be further boosted with additional fine-tuning in the testing domain. Another method to enhance generalizability was to compound datasets containing a mixture of distinct MRI contrasts during network training [15]. This approach enforces the network to better adapt to variations in tissue contrast. Yet, in the absence of contrast-specific fine-tuning, networks may deliver suboptimal performance for some individual contrasts. A recent study proposed implicit domain transfer from natural images to MR images [112]. Our study differs from [112] in several ways. A. We propose explicit domain transfer via end-to-end fine tuning in the target domain, which is shown to significantly enhance success of domain-transferred networks. B. We introduce pretraining and domain-transfer approaches capable of reconstructing multi-coil MRI data. C. Unlike [112] where the number of training samples was 118k in the natural-image domain versus 0.3k-1k in the MRI domain, here we maintain comparable sizes of training data across different domains with the ratio of natural to MR images ranging in [1 5.5]. D. Unlike [112] that reports domain-transferred networks to occasionally outperform networks trained in the target domain, networks obtained via implicit domain transfer perform sub-optimally in all cases examined here. This apparent contradiction might be attributed to the differences in the relative size of training data between the natural image and MRI domains.

A second group of studies have attempted explicit domain transfer across training and testing domains via fine-tuning. A recent proposed method trained a deep residual network to remove streaking artifacts from CT images, and the trained network was then used to suppress aliasing artifacts in projection-reconstruction MRI [16]. This method leverages the notion that the characteristic structure of artifacts due to polar sampling should be similar in CT and MRI. Here, we considered random sampling patterns on a Cartesian grid, and therefore, the domain transfer method proposed in [16] is not directly applicable to our reconstructions that possess incoherent artifacts. Similar idea can be extended to randomm sampling patterns on a non-Cartesian grid nonethless by passing the image through a gridding module prior to feedin it to the network [113, 114]. Another recent, independent effort examined the reliability of reconstructions from a variational

network to deviations in undersampling patterns and SNR between the training and testing domains [70]. Mismatch in patterns or SNR between the two domains caused suboptimal performance even for modest acceleration factors. They also assessed the generalization capability by performing implicit domain transfer between PD-weighted knee images with and without fat suppression. A network trained on PD-weighted knee images without fat suppression was observed to yield relatively poor reconstructions of images with fat suppression and vice versa. Consistent with these observations, we also find that, without fine-tuning, networks trained on MR images of a given contrast (e.g. T_1-weighted) do not generalize well to images of a different contrast (e.g. T_2-weighted). That said, a distinct contribution of our work was to address the issue of data scarcity by training a network in a domain with ample data, and transferring the network to a domain with fewer samples.

An alternative approach proposed to train neural networks for MRI reconstruction with small datasets is Robust artificialneuralnetworks for kspace interpolation, RAKI [11]. This previous method aims to train a neural network for each individual subject that learns to synthesize missing k-space samples from acquired data. Unlike traditional k-space parallel imaging methods [75, 115], a nonlinear interpolation kernel was estimated from central calibration data. Such nonlinear interpolation was shown to boost reconstruction performance beyond linear methods. However, RAKI might yield suboptimal performance when the optimal interpolation kernel shows considerable variation across k-space. Our proposed architecture for multi-coil reconstructions leverages a linear interpolation kernel, so the output of calibration-consistency blocks in our network can manifest similar reconstruction errors. Yet, the remaining CNN blocks are trained to recover fully-sampled reference images given images with residual artifacts at the output of CC blocks.

Here, we demonstrated domain transfer based on a cascade architecture with multiple CNNs interleaved with data- and calibration-consistency blocks. The proposed approach might facilitate the use of neural networks for MRI reconstruction in applications where data are relatively scarce. It might also benefit other types of architectures that have been proposed for accelerated MRI [8, 14, 69],

in particular architectures that require extensive datasets for adequate training [4, 5]. Here, the calibration-consistency projections were based on the SPIRiT method. These projections can also be replaced with other k-space methods for parallel imaging such as GRAPPA or RAKI. Note that the current study examined the generalization capability of networks trained on natural images to T_1-weighted and T_2-weighted images of the brain. ImageNet-trained networks could also be beneficial for reconstruction of MR images acquired with more specialized contrasts such as angiograms, and images acquired in other body parts.

Table 3.2: Quality of recovered single-coil magnitude T_1-weighted images

		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=4	Raw	34.92 3.57	0.96 0.02	36.01 3.17	0.97 0.02	34.11 3.64	0.95 0.03
	Tuned	36.06 3.10	0.97 0.01	36.47 3.27	0.97 0.01	36.22 3.13	0.97 0.02
	CS			**Limited**			
	PSNR		SSIM	PSNR	SSIM		
	31.77 3.51		0.92 0.03	29.79 3.95	0.93 0.03		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=6	Raw	31.97 3.33	0.94 0.02	33.49 3.29	0.95 0.02	31.27 3.78	0.93 0.04
	Tuned	33.53 3.02	0.95 0.02	33.99 3.29	0.96 0.02	33.71 3.17	0.96 0.02
	CS			**Limited**			
	PSNR		SSIM	PSNR	SSIM		
	29.71 3.52		0.89 0.04	27.02 4.35	0.90 0.05		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=8	Raw	29.78 3.75	0.92 0.03	31.64 3.37	0.94 0.03	29.72 3.75	0.91 0.05
	Tuned	32.56 3.15	0.95 0.02	32.32 3.36	0.95 0.03	32.39 3.45	0.95 0.02
	CS			**Limited**			
	PSNR		SSIM	PSNR	SSIM		
	28.56 3.53		0.88 0.05	24.02 4.46	0.85 0.07		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=10	Raw	28.52 3.86	0.91 0.04	30.50 3.37	0.93 0.03	28.70 3.70	0.90 0.05
	Tuned	30.37 3.34	0.93 0.03	31.12 ± 3.38	0.94 ± 0.03	30.78 ± 3.14	0.93 ± 0.03
	CS			**Limited**			
	PSNR		SSIM	PSNR	SSIM		
	27.98 ± 3.49		0.87 ± 0.05	21.45 ± 4.52	0.79 ± 0.08		

Reconstruction quality for single-coil magnitude T_1-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained, T_1-trained, T_2-trained and limited networks. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_1-weighted images (tuned).

Table 3.3: Quality of recovered single-coil magnitude T_1-weighted images

		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=4	Raw	34.92 ±3.57	0.96 ± 0.02	36.01 ± 3.17	0.97 ± 0.02	34.11 ± 3.64	0.95 ± 0.03
	Tuned	36.62 ± 3.28	0.97± 0.01	37.08 ± 3.28	0.97 ± 0.01	36.93 ± 3.30	0.97 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	31.77 ± 3.51	0.92 ± 0.03		35.32± 2.96	0.97 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=6	Raw	31.97 ± 3.33	0.94 ± 0.02	33.49 ± 3.29	0.95 ± 0.02	31.27 ± 3.78	0.93 ± 0.04
	Tuned	34.03 ± 3.27	0.96± 0.02	34.32 ± 3.40	0.96 ± 0.02	34.36 ± 3.38	0.96 ± 0.02
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	29.71 ± 3.52	0.89 ± 0.04		32.35 ± 3.34	0.95 ± 0.02		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=8	Raw	29.78 ± 3.75	0.92 ± 0.03	31.64± 3.37	0.94 ± 0.03	29.72 ± 3.75	0.91 ± 0.05
	Tuned	32.98 ± 3.20	0.95 ± 0.02	32.50 ± 3.47	0.95 ± 0.03	32.82 ± 3.39	0.95 ± 0.02
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	28.56 ± 3.53	0.88 ± 0.05		29.54 ± 3.70	0.92 ± 0.03		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=10	Raw	28.52 ± 3.86	0.91 ± 0.04	30.50 ± 3.37	0.93 ± 0.03	28.70 ± 3.70	0.90 ± 0.05
	Tuned	31.26 ± 3.41	0.93 ± 0.02	31.72 ± 3.40	0.94 ± 0.02	31.65 ± 3.29	0.94 ± 0.02
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	27.98 ± 3.49	0.87 ± 0.05		29.00 ± 3.66	0.91 ± 0.04		

Reconstruction quality for single-coil magnitude T_1-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained, T_1-trained, T_2-trained and limited networks, as well as conventional CS. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with 100 T_1-weighted images (tuned).

Table 3.4: Quality of recovered single-coil magnitude T_1-weighted images

		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=4	Raw	33.43 ± 1.52	0.95 ± 0.01	34.29 ± 1.54	0.96 ± 0.01	34.99 ± 1.40	0.97 ± 0.01
	Tuned	35.37 ± 1.38	0.97 ± 0.01	35.79 ± 1.38	0.97 ± 0.01	35.69 ± 1.38	0.97 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	29.79 ± 1.51	0.90 ± 0.02		33.89 ± 1.51	0.96 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=6	Raw	31.41 ± 1.38	0.93 ± 0.01	31.94± 1.47	0.94 ± 0.01	32.43 ± 1.37	0.95 ± 0.01
	Tuned	32.96 ± 1.36	0.95 ± 0.01	33.19 ± 1.38	0.95 ± 0.01	33.41 ± 1.38	0.96 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	27.71 ± 1.54	0.87 ± 0.02		32.11 ± 1.35	0.94 ± 0.031		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=8	Raw	29.99 ± 1.35	0.92 ± 0.02	30.45 ± 1.45	0.93 ± 0.02	31.11 ± 1.32	0.94 ± 0.02
	Tuned	31.75 ± 1.34	0.94 ± 0.01	31.58 ± 1.38	0.94 ± 0.01	31.92 ± 1.34	0.94 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	26.74 ± 1.55	0.85 ± 0.03		30.67 ± 1.33	0.93 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=10	Raw	29.04 ± 1.36	0.91 ± 0.02	29.43 ± 1.40	0.91 ± 0.02	30.25 ± 1.28	0.93 ± 0.02
	Tuned	30.84 ± 1.30	0.93 ± 0.02	30.92 ± 1.33	0.93 ± 0.02	31.05 ± 1.34	0.94 ± 0.02
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	26.16 ± 1.54	0.83 ± 0.03		28.53 ± 1.35	0.90 ± 0.02		

Reconstruction quality for single-coil magnitude T_2-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained, T_1-trained, T_2-trained and limited networks, as well as conventional CS. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_2-weighted images (tuned).

Table 3.5: Quality of recovered single-coil magnitude T_2-weighted images

		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=4	Raw	33.43 ± 1.52	0.95 ± 0.01	34.29 ± 1.54	0.96 ± 0.01	34.99 ± 1.40	0.97 ± 0.01
	Tuned	35.68 ± 1.37	0.97 ± 0.01	36.22 ± 1.36	0.97 ± 0.01	36.09 ± 1.37	0.97 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	29.79 ± 1.51	0.90 ± 0.02		35.25 ± 1.40	0.97 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=6	Raw	31.41 ± 1.38	0.93 ± 0.01	31.94± 1.47	0.94 ± 0.01	32.43 ± 1.37	0.95 ± 0.01
	Tuned	33.22 ± 1.34	0.95 ± 0.01	33.53 ± 1.37	0.96 ± 0.01	33.70 ± 1.36	0.96 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	27.71 ± 1.54	0.87 ± 0.02		32.35 ± 1.42	0.95 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=8	Raw	29.99 ± 1.35	0.92 ± 0.02	30.45 ± 1.45	0.93 ± 0.02	31.11 ± 1.32	0.94 ± 0.02
	Tuned	32.23 ± 1.35	0.95 ± 0.01	32.00 ± 1.37	0.94 ± 0.01	32.31 ± 1.35	0.95 ± 0.01
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	26.74 ± 1.55	0.85 ± 0.03		30.78 ± 1.32	0.93 ± 0.01		
		ImageNet-trained		**T_1-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=10	Raw	29.04 ± 1.36	0.91 ± 0.02	29.43 ± 1.40	0.91 ± 0.02	30.25 ± 1.28	0.93 ± 0.02
	Tuned	31.19 ± 1.29	0.94 ± 0.01	31.22 ± 1.34	0.94 ± 0.02	31.31 ± 1.32	0.94 ± 0.02
	CS			**Limited**			
	PSNR	SSIM		PSNR	SSIM		
	26.16 ± 1.54	0.83 ± 0.03		30.79 ± 1.32	0.93 ± 0.02		

Reconstruction quality for single-coil magnitude T_2-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained, T_1-trained, T_2-trained and limited networks, as well as conventional CS. PSNR and SSIM values are reported as meanstandard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with 100 T_2-weighted images (tuned).

Table 3.6: Quality of recovered multi-coil complex T_1-weighted images

		ImageNet-trained		**T_1-trained**		**SPIRiT**	
R=4		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	43.48±1.89	0.984±.006	45.36±1.75	0.989±.004	44.60±1.75	0.987±.004
	Tuned	44.82±1.77	0.989±.004				
R=6		**ImageNet-trained**		**T_1-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	39.77±1.84	0.968±.010	42.06±1.85	0.981±.006	40.62±1.73	0.975±.007
	Tuned	41.72±1.87	0.980±.007				
R=8		**ImageNet-trained**		**T_1-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	36.64±1.66	0.951±.014	39.14±1.75	0.971±.009	37.11±1.71	0.961±.012
	Tuned	38.50±1.77	0.969±.010				
R=10		**ImageNet-trained**		**T_1-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	34.46±1.59	0.941±.016	36.19±1.85	0.960±.012	34.23±1.72	0.948±.017
	Tuned	35.24±1.82	0.957±.013				

Reconstruction quality for multi-coil complex T_1-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained and T_1-trained networks as well as SPIRiT. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_1-weighted images (tuned).

Table 3.7: Quality of recovered multi-coil complex T_2-weighted images

		ImageNet-trained		**T_2-trained**		**SPIRiT**	
R=4		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	51.06 ± 3.20	0.995±.004	53.08±3.00	0.997±.002	52.61±3.15	0.997±.003
	Tuned	52.77 ± 2.92	0.997±.002				
R=6		**ImageNet-trained**		**T_2-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	44.07±2.98	0.984±.011	47.46±2.55	0.992±.006	45.88±3.04	0.987±.021
	Tuned	47.00±2.48	0.992±.006				
R=8		**ImageNet-trained**		**T_2-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	40.31±2.16	0.974±.017	42.71±1.98	0.984±.010	40.06±3.43	0.966±.075
	Tuned	42.31±1.86	0.983±.011				
R=10		**ImageNet-trained**		**T_2-trained**		**SPIRiT**	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	Raw	37.91±1.52	0.967±.019	39.41±1.49	0.975±.015	36.54±2.87	0.953±.077
	Tuned	39.10±1.44	0.974±.015				

Reconstruction quality for multi-coil complex T_2-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained and T_2-trained networks as well as SPIRiT. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_2-weighted images (tuned).

Table 3.8: Quality of recovered single-coil complex T_1-weighted images

		ImageNet-trained		**T_1-trained**	
R=4		PSNR	SSIM	PSNR	SSIM
	Raw	31.64±2.38	0.91±0.04	36.31±1.83	0.95±0.02
	Tuned	34.32±1.92	0.94±0.02		
R=6		**ImageNet-trained**		**T_1-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	28.66±2.11	0.87±0.05	32.91±1.82	0.93±.02
	Tuned	31.35±1.84	0.92±0.02		
R=8		**ImageNet-trained**		**T_1-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	26.98±2.06	0.85±0.05	31.07±2.00	0.92±0.03
	Tuned	29.95±2.03	0.91±0.03		
R=10		**ImageNet-trained**		**T_1-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	26.07±2.34	0.84±0.05	29.98±2.06	0.91±0.03
	Tuned	28.07±2.32	0.89±0.04		

Reconstruction quality for single-coil complex T_1-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained and T_1-trained networks. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_1-weighted images (tuned).

Table 3.9: Quality of recovered single-coil complex T_2-weighted images

		ImageNet-trained		**T_2-trained**	
R=4		PSNR	SSIM	PSNR	SSIM
	Raw	35.75±1.69	0.95±0.03	38.41±1.73	0.97±0.02
	Tuned	37.27±1.65	0.96±0.02		
R=6		**ImageNet-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	32.48±1.76	0.93±0.03	35.55±1.49	0.95±0.02
	Tuned	34.75±1.50	0.95±0.02		
R=8		**ImageNet-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	31.97±1.62	0.92±0.03	33.84±1.42	0.94±0.03
	Tuned	32.95±1.62	0.93±0.03		
R=10		**ImageNet-trained**		**T_2-trained**	
		PSNR	SSIM	PSNR	SSIM
	Raw	30.57±1.66	0.91±0.03	32.87±1.40	0.93±0.03
	Tuned	32.55±1.42	0.93±0.03		

Reconstruction quality for single-coil complex T_2-weighted images undersampled at R= 4, 6, 8, 10. Reconstructions were performed via ImageNet-trained and T_2-trained networks. PSNR and SSIM values are reported as mean±standard deviation across test images. Results are shown for raw networks trained on 2000 training images (raw), and fine-tuned networks tuned with few tens of T_2-weighted images (tuned).

Chapter 4

Image Synthesis in Multi-Contrast MRI With Conditional Generative Adversarial Networks

The contents of this chapter reflect the work reported in the following publication:

- S. U. H. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast MRI with conditional generative adversarial networks," *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2375–2388, 2019

4.1 Introduction

Magnetic resonance imaging (MRI) is pervasively used in clinical applications due to the diversity of contrasts it can capture in soft tissues. Tailored MRI

pulse sequences enable the generation of distinct contrasts while imaging the same anatomy. For instance, T_1-weighted brain images clearly delineate gray and white matter tissues, whereas T_2-weighted images delineate fluid from cortical tissue. In turn, multi-contrast images acquired in the same subject increase the diagnostic information available in clinical and research studies. However, it may not be possible to collect a full array of contrasts given considerations related to the cost of prolonged exams and uncooperative patients, particularly in pediatric and elderly populations [116]. In such cases, acquisition of contrasts with relatively shorter scan times might be preferred. Even then a subset of the acquired contrasts can be corrupted by excessive noise or artifacts that prohibit subsequent diagnostic use [117]. Moreover, cohort studies often show significant heterogeneity in terms of imaging protocol and the specific contrasts that they acquire [118]. Thus, the ability to synthesize missing or corrupted contrasts from other successfully acquired contrasts has potential value for enhancing multi-contrast MRI by increasing availability of diagnostically-relevant images, and improving analysis tasks such as registration and segmentation [119]. Cross-domain synthesis of medical images has recently been gaining popularity in medical imaging. Given a subjects image x in X (source domain), the aim is to accurately estimate the respective image of the same subject y in Y (target domain). Two main synthesis approaches are registration-based [120, 121, 122] and intensity-transformation-based methods [123, 124, 21, 125, 126, 127, 128, 23, 129, 130, 131, 24, 26, 25, 22, 132, 133]. Registration-based methods start by generating an atlas based on a co-registered set of images, x_1 and y_1, respectively acquired in X and Y [5]. These methods further make the assumption that within-domain images from separate subjects are related to each other through a geometric warp. For synthesizing y_2 from x_2, the warp that transforms x_1 to x_2 is estimated, and this warp is then applied on y_1. Since they only rely on geometric transformations, registration-based methods that rely on a single atlas can suffer from across-subject differences in underlying morphology [132]. For example, inconsistent pathology across a test subject and the atlas can cause failure. Multi-atlas registration in conjunction with intensity fusion can alleviate this limitation, and has been

successfully used in synthesizing CT from MR images [121, 122]. Nevertheless, within-domain registration accuracy might still be limited even in normal subjects [132]. An alternative is to use intensity-based methods that do not rely on a strict geometric relationship among different subjects anatomies [123, 124, 21, 125, 126, 127, 128, 23, 129, 130, 131, 24, 26, 25, 22, 132, 133]. One powerful approach for multi-contrast MRI is based on the compressed sensing framework, where each patch in the source image x_2 is expressed as a sparse linear combination of patches in the atlas image x_1 [21, 22]. The learned sparse combinations are then applied to estimate patches in y_2 from patches in y_1. To improve matching of patches across domains, generative models were also proposed that use multi-scale patches and tissue segmentation labels [129, 131]. Instead of focusing on linear models, recent studies aimed to learn more general nonlinear mappings that express individual voxels in y_1 in terms of patches in x_1, and then predict y_2 from x_2 based on these mappings. Nonlinear mappings are learned on training data via techniques such as nonlinear regression [123, 124, 132] or location-sensitive neural networks [24]. An important example is Replica that performs random forest regression on multiresolution image patches [132]. Replica demonstrates great promise in multi-contrast MR image synthesis. However, dictionary construction at different spatial scales is independent, and the predictions from separate random forest trees are averaged during synthesis. These may lead to loss of detailed structural information and suboptimal synthesis performance. Recently an end-to-end framework for MRI image synthesis has been proposed, Multimodal, based on deep neural networks [25]. Multimodal trains a neural network that receives as input images in multiple source contrasts and predicts the image in the target contrast. This method performs multiresolution dictionary construction and image synthesis in a unified framework, and it was demonstrated to yield higher synthesis quality compared to non-network-based approaches even when only a subset of the source contrasts is available. That said, Multimodal assumes the availability of spatially-registered multi-contrast images. In addition, Multimodal uses mean absolute error loss functions that can perform poorly in capturing errors towards higher spatial frequencies [134, 135, 136].

Here we propose a novel approach for image synthesis in multi-contrast MRI

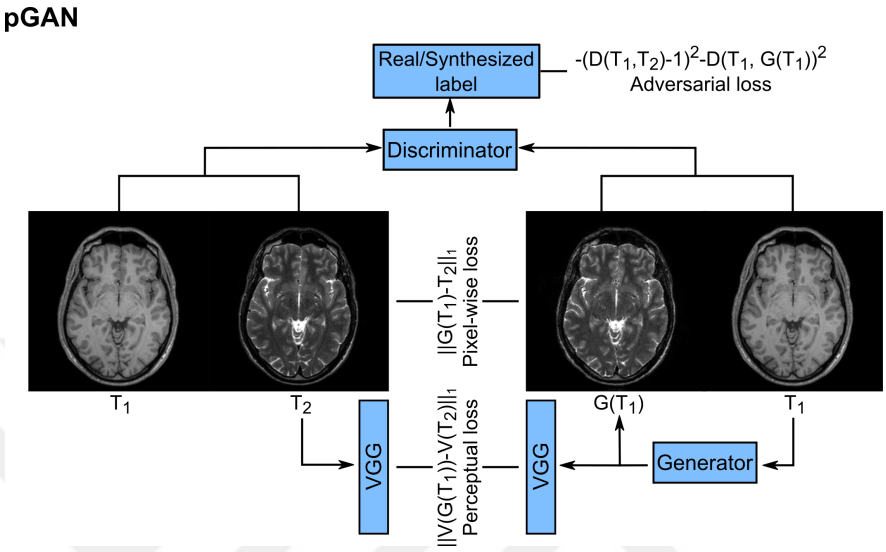

Figure 4.1: The pGAN method is based on a conditional adversarial network with a generator G, a pre-trained VGG16 network V, and a discriminator D. Given an input image in a source contrast (e.g., T_1-weighted), G learns to generate the image of the same anatomy in a target contrast (e.g., T_2-weighted). Meanwhile, D learns to discriminate between synthetic (e.g., T_1-$G(T_1)$) and real (e.g., T_1-T_2) pairs of multi-contrast images. Both subnetworks are trained simultaneously, where G aims to minimize a pixel-wise, a perceptual and an adversarial loss function, and D tries to maximize the adversarial loss function.

based on generative adversarial network (GAN) architectures. Adversarial loss functions have recently been demonstrated for various medical imaging applications with reliable capture of high-frequency texture information [137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 5, 3, 4, 152, 153, 154]. In the domain of cross-modality image synthesis, important applications include CT to PET synthesis [138, 149], MR to CT synthesis [135, 142, 147, 151, 154], CT to MR synthesis [145], and retinal vessel map to image synthesis [144, 150]. Inspired by this success, here we introduce conditional GAN models for synthesizing images of distinct contrasts from a single modality, with demonstrations on multi-contrast brain MRI in normal subjects and glioma patients. For improved accuracy, the proposed method also leverages correlated information across neighboring cross-sections within a volume. Two implementations are provided for use when multi-contrast images are spatially registered (pGAN) and when they are unregistered (cGAN). For the first scenario, we train pGAN with pixel-wise loss

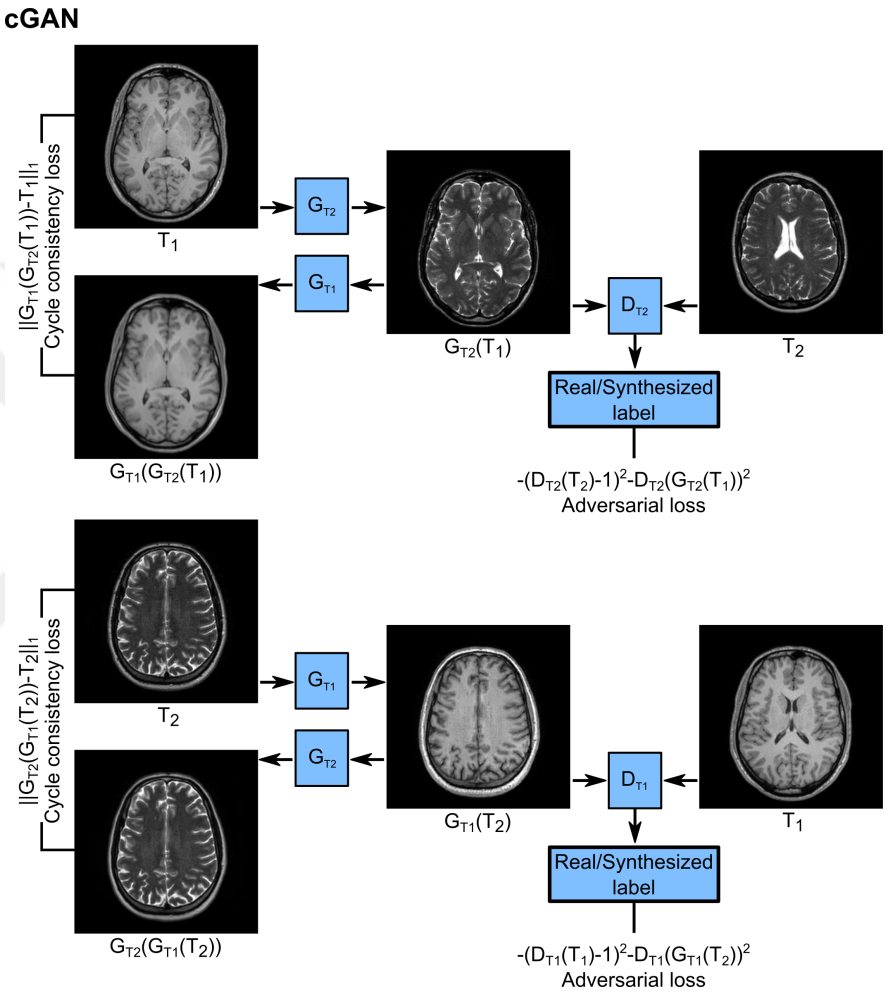

Figure 4.2: The cGAN method is based on a conditional adversarial network with two generators (G_{T_1}, G_{T_2}) and two discriminators (D_{T_1}, D_{T_2}). Given a T_1-weighted image, G_{T_2} learns to generate the respective T_2-weighted image of the same anatomy that is indiscriminable from real T_2-weighted images of other anatomies, whereas D_{T_2} learns to discriminate between synthetic and real T_2-weighted images. Similarly, G_{T_1} learns to generate realistic a T_1-weighted image of an anatomy given the respective T_2-weighted image, whereas D_{T_1} learns to discriminate between synthetic and real T_1-weighted images. Since the discriminators do not compare target images of the same anatomy, a pixel-wise loss cannot be used. Instead, a cycle-consistency loss is utilized to ensure that the trained generators enable reliable recovery of the source image from the generated target image.

and perceptual loss between the synthesized and true images (Fig. 4.1) [134, 111]. For the second scenario, we train cGAN after replacing the pixel-wise loss with a cycle loss that enforces the ability to reconstruct back the source image from the synthesized target image (Fig. 4.2) [155]. Extensive evaluations are presented on multi-contrast MRI images (T_1- and T_2-weighted) from healthy normals and glioma patients. The proposed approach yields visually and quantitatively enhanced accuracy in multi-contrast MRI synthesis compared to state-of-the-art methods (Replica and Multimodal) [25, 132].

4.2 Methods

4.2.1 Image Synthesis via Adversarial Networks

Generative adversarial networks are neural-network architectures that consist of two sub-networks; G, a generator and D, a discriminator. G learns a mapping from a latent variable z (typically random noise) to an image y in a target domain, and D learns to discriminate the generated image $G(z)$ from the real image y [110]. During training of a GAN, both G and D are learned simultaneously, with G aiming to generate images that are indistinguishable from the real images, and D aiming to tell apart generated and real images. To do this, the following adversarial loss function (L_{GAN}) can be used:

$$L_{GAN}(G, D) = E_y[\log D(y)] + E_z[\log(1 - D(G(z)))] \tag{4.1}$$

where E denotes expected value. G tries to minimize and D tries to maximize the adversarial loss that improves modeling high-spatial-frequency information [135]. Both G and D are trained simultaneously. Upon convergence, G is capable of producing realistic counterfeit images that D cannot recognize [110]. To further stabilize the training process, the negative log-likelihood cost for adversarial loss in Eq. 4.1 can be replaced by a squared loss [156]:

$$L_{GAN}(G, D) = -E_y[(D(y) - 1)^2] - E_z[D(G(z))^2] \tag{4.2}$$

Recent studies in computer vision have demonstrated that GANs are very effective in image-to-image translation tasks [111, 155]. Image-to-image translation concerns transformations between different representations of the same underlying visual scene [111]. These transformations can be used to convert an image between separate domains, e.g., generating semantic segmentation maps from images, colored images from sketches, or maps from aerial photos [111, 157, 158]. Traditional GANs learn to generate samples of images from noise. However, in image-to-image translation, the synthesized image has statistical dependence on the source image. To better capture this dependency, conditional GANs can be employed that receive the source image as an additional input [159]. The resulting network can then be trained based on the following adversarial loss function:

$$L_{condGAN}(G, D) = -E_{x,y}[(D(x, y) - 1)^2] - E_{x,z}[D(x, G(x, z))^2] \tag{4.3}$$

where x denotes the source image. An analogous problem to image-to-image translation tasks in computer vision exists in MR imaging where the same anatomy is acquired under multiple different tissue contrasts (e.g., T_1- and T_2-weighted images). Inspired by the recent success of adversarial networks, here we employed conditional GANs to synthesize MR images of a target contrast given as input an alternate contrast. For a comprehensive solution, we considered two distinct scenarios for multi-contrast MR image synthesis. First, we assumed that the images of the source and target contrasts are perfectly registered. For this scenario, we propose pGAN that incorporates a pixel-wise loss into the objective function as inspired by the pix2pix architecture [111]:

$$L_{L1}(G) = E_{x,y,z}[||y - G(x, z)||_1] \tag{4.4}$$

where L_{L1} is the pixel-wise L1 loss function. Since the generator G was observed to ignore the latent variable in pGAN, the latent variable was removed from the model. Recent studies suggest that incorporation of a perceptual loss during network training can yield visually more realistic results in computer vision tasks. Unlike loss functions based on pixel-wise differences, perceptual loss relies on differences in higher feature representations that are often extracted from networks pre-trained for more generic tasks [134]. A commonly used network is VGG-net trained on the ImageNet [88] dataset for object classification. Here, following

[134], we extracted feature maps right before the second max-pooling operation of VGG16 pre-trained on ImageNet. The resulting loss function can be written as:

$$L_{perc}(G) = E_{x,y}[||V(y) - V(G(x))||_1] \tag{4.5}$$

where V is the set of feature maps extracted from VGG16. To synthesize each cross-section y from x we also leveraged correlated information across neighboring cross-sections by conditioning the networks not only on x but also on the neighboring cross-sections of x. By incorporating the neighboring cross-sections Eqs. 4.3, 4.4 , and 4.5 become:

$$L_{condGAN-k}(D, G) = -E_{x_k,y}[(D(x_k, y) - 1)^2] - E_{x_k,z}[D(x_k, G(x_k))^2] \tag{4.6}$$

$$L_{L1_k}(G) = E_{x_k,y}[||y - G(x_k)||_1] \tag{4.7}$$

$$L_{perc_k}(G) = E_{x_k,y}[||V(y) - V(G(x_k))||_1] \tag{4.8}$$

where $x_k = [x_{-\lfloor \frac{k}{2} \rfloor}, \ldots, x_{-2}, x_{-1}, x, x_{-1}, x_{-2}, \ldots, \ x_{+\lfloor \frac{k}{2} \rfloor}]$ is a vector consisting of k consecutive cross-sections ranging from $-\lfloor \frac{k}{2} \rfloor$ to $\lfloor \frac{k}{2} \rfloor$, with the cross section x in the middle, and $L_{condGAN-k}$ and L_{L1-k} are the corresponding adversarial and pixel-wise loss functions. This yields the following aggregate loss function:

$$L_{pGAN}(D, G) = L_{condGAN-k}(D, G) + \lambda L_{L1_k}(G) + \lambda_{perc} L_{perc_k}(G) \tag{4.9}$$

where L_{pGAN} is the complete loss function, λ controls the relative weighing of the pixel-wise loss and λ_{perc} controls the relative weighing of the perceptual loss. In the second scenario, we did not assume any explicit registration between the images of the source and target contrasts. In this case, the pixel-wise and perceptual losses cannot be leveraged since images of different contrasts are not necessarily spatially aligned. To limit the number of potential solutions for the synthesized image, here we proposed cGAN that incorporates a cycle-consistency loss as inspired by the cycleGAN architecture [155]. The cGAN method consists of two generators (G_x,G_y) and two discriminators (D_x,D_y). G_y tries to generate $G_y(x)$ that looks similar to y and D_y tries to distinguish $G_y(x)$ from the images y. On

the other hand, G_x tries to generate $G_x(y)$ that looks similar to x and D_x tries to distinguish $G_x(y)$ from the images x. This architecture incorporates an additional loss to ensure that the input and target images are consistent with each other, called the cycle consistency loss L_{cycle}:

$$L_{cycle}(G_x, G_y) = E_x[||y - G_x(G_y(x))||_1] + E_y[||y - G_y(G_x(y))||_1] \tag{4.10}$$

This loss function enforces that property that after projecting the source images onto the target domain, the source image can be re-synthesized with minimal loss from the projection. Lastly, by incorporating the neighboring cross-sections, the cycle consistency and adversarial loss functions become:

$$L_{cycle-k}(G_x, G_y) = E_{x_k}[||x_k - G_x(G_y(x_k))||_1] + E_{y_k}[||y_k - G_y(G_x(y_k))||_1] \tag{4.11}$$

$$L_{GAN-k}(D_y, G_y) = -E_{y_k}[(D_y(y_k) - 1)^2] - E_{x_k}[D_y(G_y(x_k))^2] \tag{4.12}$$

This yields the following aggregate loss function for training:

$$\begin{aligned} L_{cGAN}(D_x, D_y, G_x, G_y) = L_{GAN-k}(D_x, G_x) + L_{GAN-k}(D_y, G_y) + \\ \lambda_{cycle} L_{cycle-k}(G_x, G_y) \end{aligned} \tag{4.13}$$

where L_{cGAN} is the complete loss function, and λ_{cycle} controls the relative weighing of the cycle consistency loss. While training both pGAN and cGAN, we made a minor modification in the adversarial loss function. As implemented in [155], the generator was trained to minimize $E_{x_k}[(D(x_k, G(x_k)) - 1)^2]$ instead of $-E_{x_k}[(D(x_k, G(x_k)))^2]$.

4.2.2 MRI Datasets

For registered images, we trained both pGAN and cGAN models. For unregistered images, we only trained cGAN models. The experiments were performed on three separate datasets: the MIDAS dataset [87], the IXI dataset (`http://brain-development.org/ixi-dataset/`) and the BRATS dataset (`https://sites.google.com/site/braintumorsegmentation/home/brats2015`). MIDAS and IXI datasets contained data from healthy subjects, whereas the BRATS

dataset contained data from patients with structural abnormality (i.e., brain tumor). For each dataset, subjects were sequentially selected in the order that they were shared on the public databases. Subjects with images containing severe motion-artifacts across the volume were excluded from selection. The selected set of subjects were then sequentially split into training, validation and testing sets. Protocol information for each dataset is described below.

MIDAS dataset: T_1- and T_2-weighted images from 66 subjects were analyzed, where 48 subjects were used for training, 5 were used for validation and 13 were used for testing. From each subject, approximately 75 axial cross sections that contained brain tissue and that were free of major artifacts were manually selected. T_1-weighted images: 3D gradient-echo FLASH sequence, TR=14ms, TE=7.7ms, flip angle=25^o, matrix size=256×176, 1 mm isotropic resolution, axial orientation. T_2-weighted images: 2D spin-echo sequence, TR=7730ms, TE=80ms, flip angle=90^o, matrix size=256×192, 1 mm isotropic resolution, axial orientation.

IXI dataset: T_1- and T_2-weighted images from 40 subjects were analyzed, where 25 subjects were used for training, 5 were used for validation and 10 were used for testing. When T_1-weighted images were registered onto T_2-weighted images, nearly 90 axial cross sections per subject that contained brain tissue and that were free of major artifacts were selected. When T_2-weighted images were registered onto T_1-weighted images, nearly 110 cross sections were selected. In this case due to poor registration quality we had to remove a test subject. T_1-weighted images: TR=9.813ms, TE=4.603ms, flip angle=8^o, volume size = 256×256×150, voxel dimensions = 0.94mm×0.94mm×1.2mm, sagittal orientation. T_2-weighted images: TR=8178ms, TE=100ms, flip angle=90^o, volume size = 256×256×150, voxel dimensions = 0.94×0.94×1.2 mm^3, axial orientation.

BRATS dataset: T_1- and T_2-weighted images from 41 low-grade glioma patients with visible lesions were analyzed, where 24 subjects were used for training, 2 were used for validation and 15 were used for testing. From each subject, approximately 100 axial cross sections that contained brain tissue and that were free of major artifacts were manually selected. Different scanning protocols were

employed on separate sites. Note that each dataset comprises a different number of cross-sections per subject, and we only retained cross-sections that contained brain tissue and that were free of major artifacts. As such, we varied the number of subjects across datasets to balance the total number of images used, resulting in approximately 4000-5000 images per dataset. Control analyses were performed to rule out biases due to the specific selection or number of subjects. To do this, we performed model comparisons using an identical number of subjects (40) within each dataset. This selection included nonoverlapping training, validation and testing sets, such that 25 subjects were used for training, 5 for validation and 10 for testing. In IXI, we sequentially selected a completely independent set of subjects from those reported in the main analyses. This selection was then sequentially split into training/validation/testing sets via a 4-fold cross-validation procedure. Since the number of subjects available was smaller in MIDAS and BRATS, we performed 4-fold cross-validation by randomly sampling nonoverlapping training, validation and testing sets in each fold. No overlap was allowed among testing sets across separate folds, or among the training, testing and validation sets within each fold.

Data normalization: To prevent suboptimal model training and bias in quantitative assessments, datasets were normalized to ensure comparable ranges of voxel intensities across subjects. The multi-contrast MRI images in the IXI and MIDAS datasets were acquired using a single scan protocol. Therefore, for each contrast, voxel intensity was normalized within each subject to a scale of [0 1] via division by the maximum intensity within the brain volume. The protocol variability in the BRATS dataset was observed to cause large deviations in image intensity and contrast across subjects. Thus, for normalization, the mean intensity across the brain volume was normalized to 1 within individual subjects. To attain an intensity scale in [0 1], three standard deviations above the mean intensity of voxels pooled across subjects was then mapped to 1.

4.2.3 Image Registration

For the first scenario, multi-contrast images from a given subject were assumed to be registered. Note that the images contained in the MIDAS and IXI datasets are unregistered. Thus, the T_1- and T_2-weighted images in these datasets were registered prior to network training. In the MIDAS dataset, the voxel dimensions for T_1- and T_2-weighted images were identical, so a rigid transformation based on a mutual information cost function was observed to yield high quality registration. In the IXI dataset, however, voxel dimensions for T_1- and T_2-weighted images were quite distinct. For improved registration accuracy, we therefore used an affine transformation with higher degrees of freedom based on a mutual information cost in this case. No registration was needed for the BRATS dataset that was already registered. No registration was performed for the second scenario. All registrations were implemented in FSL [160, 161].

4.2.4 Network Training

Since we consider two different scenarios for multi-contrast MR image synthesis, network training procedures were distinct. In the first scenario, we assumed perfect alignment between the source and target images, and we then used pGAN to learn the mapping from the source to the target contrast. In a first variant of pGAN (k=1), the input image was a single cross-section of the source contrast, and the target was the respective cross-section of the desired contrast. Note that neighboring cross sections in MR images are expected to show significant correlation. Thus, we reasoned that additional information from adjacent cross-sections in the source contrast should improve synthesis. To do this, a second variant of pGAN was implemented where multiple consecutive cross-sections (k=3, 5, 7) of the source contrast were given as input, with the target corresponding to desired contrast at the central cross-section. For the pGAN network, we adopted the generator architecture from [134], and the discriminator architecture from [155]. Tuning hyperparameters in deep neural networks, especially in complex models such as GANs, can be computationally intensive [162, 163]. Thus, it

is quite common in deep learning research to perform one-fold cross-validation [139, 144] or even directly adopt hyperparameter selection from published work [133, 137, 138, 147, 154, 29]. For computational efficiency, here we selected the optimum weightings of loss functions and number of epochs by performing one-fold cross-validation. We partitioned the datasets into training, validation and test sets, each set containing images from distinct subjects. Multiple models were trained for varying number of epochs (in the range [100 200]) and relative weighting of the loss functions (λ in the set 10,100,150, and λ_{perc} in the set 10,100,150). Parameters were selected based on the validation set, and performance was then assessed on the test set. Among the datasets here, IXI contains the highest-quality images with visibly lower noise and artifact levels compared to MIDAS and visibly sharper images compared to BRATS. To prevent overfitting to noise, artifacts or blurry images, we therefore performed cross-validation of GAN models on IXI, and used the selected parameters in the remaining datasets. Weightings of both pixel-wise and perceptual loss were selected as 100 and the number of epochs was set to 100 (the benefits of perceptual loss on synthesis performance are demonstrated in MIDAS and IXI; Table 4.1). Remaining hyperparameters were adopted from [155], where the Adam optimizer was used with a minibatch size of 1 [94]. In the first 50 epochs, the learning rates for the generator and discriminator were 0.0002. In the last 50 epochs, the learning rate was linearly decayed from 0.0002 to 0. During each iteration the discriminator loss function was halved to slow down the learning process of the discriminator. Decay rates for the first and second moments of gradient estimates were set as β_1= 0.5 and β_2=0.999, respectively. Instance normalization was applied [164]. All weights were initialized using normal distribution with 0 mean and 0.02 std. In the second scenario, we did not assume any alignment between the source and target images, and so we used cGAN to learn the mapping between unregistered source and target images ($\text{cGAN}_{\text{unreg}}$). Similar to pGAN, two variants of cGAN were considered that worked on a single cross-section (k=1) and on multiple consecutive cross-sections. Because training of cGAN brings substantial computational burden compared to pGAN, we only examined k=3 for cGAN. This latter cGAN variant was implemented with multiple consecutive cross-sections of the source contrast. Although cGAN does not assume alignment between the source and

target domains, we wanted to examine the effects of loss functions used in cGAN and pGAN. For comparison purposes, we also trained separate cGAN networks on registered multi-contrast data (cGANreg). The cross-validation procedures, and the architectures of the generator and discriminator were identical to those for pGAN. Multiple models were trained for varying number of epochs (in the range [100 200]), and λ_{cycle} in the set 10,100,150). Model parameters were selected based on performance on the validation set, and model performance was then assessed on the test set. The relative weighting of the cycle consistency loss function was selected as λ_{cycle}=100, and the model was trained for 200 epochs. In the first 100 epochs, the learning rate for both networks were set to 0.0002, and in the remaining 100 epochs, the learning rate was linearly decayed from 0.0002 to 0. During each iteration the discriminator loss function was divided by 2 to slow down the learning process of the discriminator.

4.2.5 Competing Methods

To demonstrate the proposed approach, two state-of-the-art methods for MRI image synthesis were implemented. The first method was Replica that estimates a nonlinear mapping from image patches in the source contrast onto individual voxels in the target contrast [132]. Replica extracts image features at different spatial scales, and then performs a multi-resolution analysis via random forests. The learned nonlinear mapping is then applied on test images. Code posted by the authors of the Replica method was used to train the models, based on the procedures/parameters described in [132]. The second method was Multimodal that uses an end-to-end neural network to estimate the target image given the source image as input. A neural-network implementation implicitly performs multi-resolution feature extraction and synthesis based on these features. Trained networks can then be applied on test images. Code posted by the authors of the Multimodal method was used to train the models, based on procedures/parameters described in [25]. The proposed approach and the competing methods were compared on the same training and test data. Since the proposed models were implemented for unimodal mapping between two separate

contrasts, Replica and Multimodal implementations were also performed with only two contrasts.

4.2.6 Experiments

4.2.6.1 Comparison of GAN-based models

Here we first questioned whether the direction of registration between multi-contrast images affects the quality of synthesis. In particular, we generated multiple registered datasets from T_1- and T_2-weighted images. In the first set, T_2-weighted images were registered onto T_1-weighted images (yielding T_2#). In the second set, T_1-weighted images were registered onto T_2-weighted images (yielding T_1#). In addition to the direction of registration, we also considered the two possible directions of synthesis (T_2 from T_1; T_1 from T_2). For MIDAS and IXI, the above-mentioned considerations led to four distinct cases: a) $T_1 \rightarrow T_2$#, b) T_1# $\rightarrow T_2$, c) $T_2 \rightarrow T_1$#, d) T_2# $\rightarrow T_1$. Here, T_1 and T_2 are unregistered images, T_1# and T_2# are registered images, and $\rightarrow$ corresponds to the direction of synthesis. For each case, pGAN and cGAN were trained based on two variants, one receiving a single cross-section, the other receiving multiple (3, 5 and 7) consecutive cross-sections as input. This resulted in a total of 32 pGAN and 12 cGAN models. Note that the single-cross section cGAN contains generators for both contrasts, and trains a model that can synthesize in both directions. For the multi cross-section cGAN, however, a separate model was trained for synthesis direction. For BRATS, no registration was needed, and this resulted in only two distinct cases for consideration: a) $T_1 \rightarrow T_2$ and d) $T_2 \rightarrow T_1$. A single variant of pGAN (k=3) and cGAN (k=1) was considered.

4.2.6.2 Comparison to state-of-the-art methods

To investigate how well the proposed methods perform with respect to state-of-the-art approaches, we compared the pGAN and cGAN models with Replica

and Multimodal. Models were compared using the same training, and testing sets, and these sets comprised images from different groups of subjects. The synthesized images were compared with the true target images as reference. Both the synthesized and the reference images were normalized to a maximum intensity of 1. To assess the synthesis quality, we measured the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [165] metrics between the synthesized image and the reference.

4.2.6.3 Spectral density analysis

While PSNR and SSIM serve as common measures to evaluate overall quality, they primarily capture characteristics dominated by lower spatial frequencies. To examine synthesis quality across a broader range of frequencies, we used a spectral density similarity (SDS) metric. The rationale for SDS is similar to that for the error spectral plots demonstrated in [166], where error distribution is analyzed across spatial frequencies. To compute SDS, synthesized and reference images were transformed into k-space, and separated into four separate frequency bands: low (0-25%), intermediate (25-50%), high-intermediate (50-75%), and high (75-100% of the maximum spatial frequency in k-space). Within each band, SDS was taken as the Pearsons correlation between vectors of magnitude k-space samples of the synthesized and reference images. To avoid bias from background noise, we masked out background regions to zero before calculating the quality measures.

4.2.6.4 Generalizability

To examine the generalizability of the proposed methods, we trained pGAN, cGAN, Replica and Multimodal on the IXI dataset and tested the trained models on the MIDAS dataset. The following cases were examined: $T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$, $T_2 \rightarrow T_1\#$, and $T_2\# \rightarrow T_1$. During testing, ten sample images were synthesized for a given source image, and the results were averaged to mitigate nuisance variability in individual samples. When T_1-weighted images were registered onto T_2-weighted images, within-cross-section voxel dimensions were isotropic

for both datasets and no extra pre-processing step was needed. However, when T_2-weighted images were registered, voxel dimensions were anisotropic for IXI yet isotropic for MIDAS. To avoid spatial mismatch, voxel dimensions were matched via trilinear interpolation. Because a mismatch of voxel thickness in the cross-sectional dimension can deteriorate synthesis performance, single cross-section models were considered.

4.2.6.5 Reliability against noise

To examine the reliability of synthesis against image noise, we trained pGAN and Multimodal on noisy images. The IXI dataset was selected since it contains high-quality images with relatively low noise levels. Two separate sets of noisy images were then generated by adding Rician noise to the source and target contrast images respectively. The noise level was fixed within subjects and randomly varied across subjects by changing the Rician shape parameter in [0 0.2]. For noise-added target images, background masking was performed prior to training and no perceptual loss was used in pGAN to prevent overfitting to noise. Separate models were trained using noise-added source and original target images, and using original source and noise-added target images. Statistical significance of differences among methods was assessed with nonparametric Wilcoxon signed-rank tests across test subjects. Neural network training and evaluation was performed on NVIDIA Titan X Pascal and Xp GPUs. Implementation of pGAN and cGAN was carried out in Python using the Pytorch framework [167]. Code for replicating the pGAN and cGAN models will be available on `http://github.com/icon-lab/mrirecon`. Replica was based on a MATLAB implementation, and a Keras implementation [168] of Multimodal with the Theano backend [169] was used.

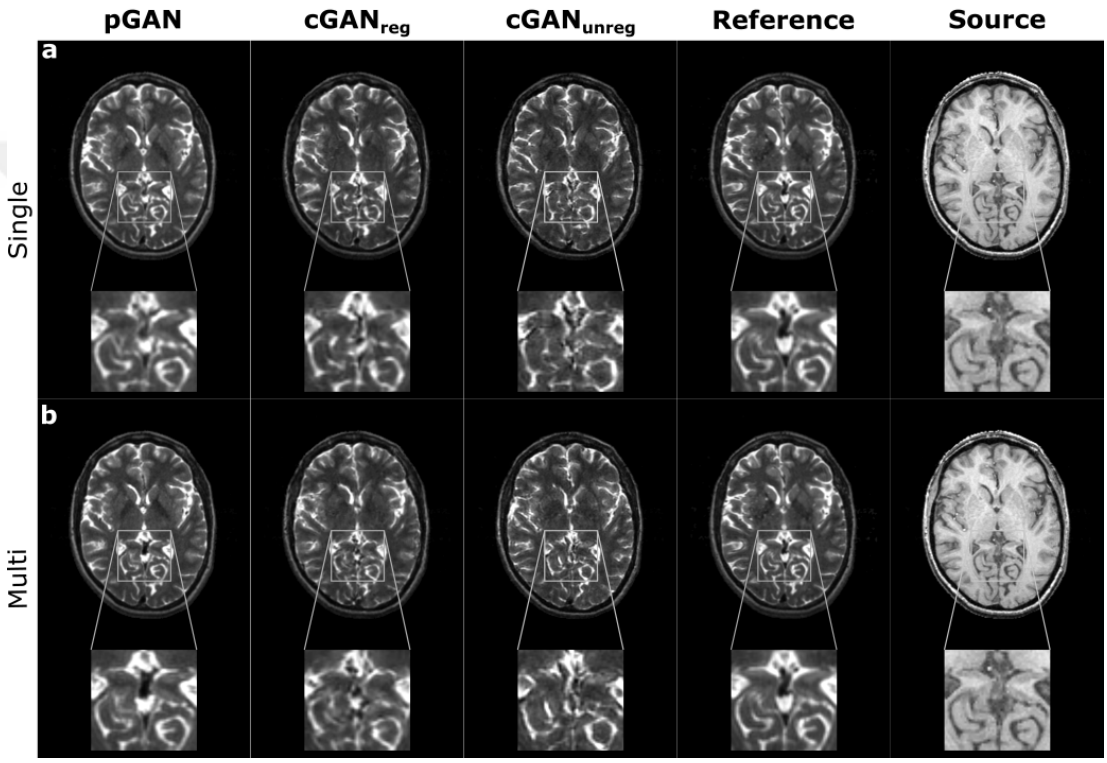

Figure 4.3: The proposed approach was demonstrated for synthesis of T_2-weighted images from T_1-weighted images in the MIDAS dataset. Synthesis was performed with pGAN, cGAN trained on registered images (cGANreg), and cGAN trained on unregistered images ($cGAN_{unreg}$). For pGAN and cGANreg, training was performed using T_2-weighted images registered onto T_1-weighted images ($T_1 \rightarrow T_2\#$). Synthesis results for (a) the single cross-section, and (b) multi cross-section models are shown along with the true target image (reference) and the source image (source). Zoomed-in portions of the images are also displayed. While both pGAN and cGAN yield synthetic images of striking visual similarity to the reference, pGAN is the top performer. Synthesis quality is improved as information across neighboring cross sections is incorporated, particularly for the pGAN method.

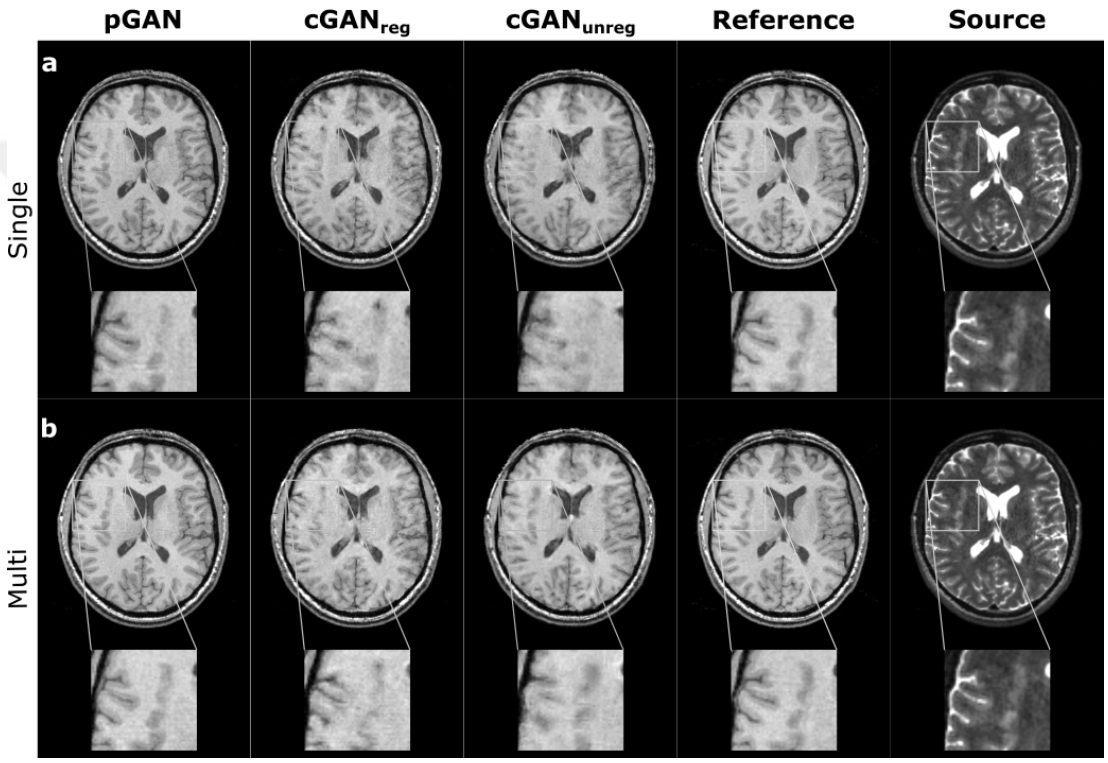

Figure 4.4: The proposed approach was demonstrated for synthesis of T_1-weighted images from T_2-weighted images in the MIDAS dataset. Synthesis was performed with pGAN, cGAN trained on registered images (cGANreg), and cGAN trained on unregistered images (cGAN$_{\text{unreg}}$). For pGAN and cGANreg, training was performed using T_2-weighted images registered onto T_1-weighted images ($T_2\# \rightarrow T_1$). Synthesis results for (a) the single cross-section, and (b) three cross-section models are shown along with the true target image (reference) and the source image (source). Zoomed-in portions of the images are also displayed. While both pGAN and cGAN yield synthetic images of striking visual similarity to the reference, pGAN is the top performer. Synthesis quality is improved as information across neighboring cross sections is incorporated for the pGAN method.

4.3 Results

4.3.1 Comparison of GAN-based models

We first evaluated the proposed models on T_1- and T_2-weighted images from the MIDAS and IXI datasets. We considered two cases for T_2 synthesis (a. $T_1 \rightarrow T_2\#$, b. $T_1\# \rightarrow T_2$, where # denotes the registered image), and two cases for T_1 synthesis (c. $T_2 \rightarrow T_1\#$, d. $T_2\# \rightarrow T_1$). Table 4.2 lists PSNR and SSIM for pGAN, $cGAN_{reg}$ trained on registered data, and $cGAN_{unreg}$ trained on unregistered data in the MIDAS dataset. We find that pGAN outperforms $cGAN_{unreg}$ and $cGAN_{reg}$ in all cases ($p<0.05$). Representative results for $T_1 \rightarrow T_2\#$ are displayed in Fig. 4.3a and $T_2\# \rightarrow T_1$ are displayed in Fig. 4.4a, respectively. pGAN yields higher synthesis quality compared to $cGAN_{reg}$. Although $cGAN_{unreg}$ was trained on unregistered images, it can faithfully capture fine-grained structure in the synthesized contrast. Overall, both pGAN and cGAN yield synthetic images of remarkable visual similarity to the reference. Tables 4.3 and 4.4 (k=1) lists PSNR and SSIM across test images for T_2 and T_1 synthesis with both directions of registration in the IXI dataset. Note that there is substantial mismatch between the voxel dimensions of the source and target contrasts in the IXI dataset, so $cGAN_{unreg}$ must map between the spatial sampling grids of the source and the target. Since this yielded suboptimal performance, measurements for $cGAN_{unreg}$ are not reported. Overall, similar to the MIDAS dataset, we observed that pGAN outperforms the competing methods ($p<0.05$). On average, across the two datasets, pGAN achieves 1.42dB higher PSNR and 1.92% higher SSIM compared to cGAN. These improvements can be attributed to pixel-wise and perceptual losses compared to cycle-consistency loss on paired images.

In MR images, neighboring voxels can show structural correlations, so we reasoned that synthesis quality can be improved by pooling information across cross sections. To examine this issue, we trained multi cross-section pGAN (k=3, 5, 7), $cGAN_{reg}$ and $cGAN_{unreg}$ models (k=3; see Methods) on the MIDAS and IXI datasets. PSNR and SSIM measurements for pGAN are listed in Table 4.3, and those for cGAN are listed in Table 4.4. For pGAN, multi cross-section models

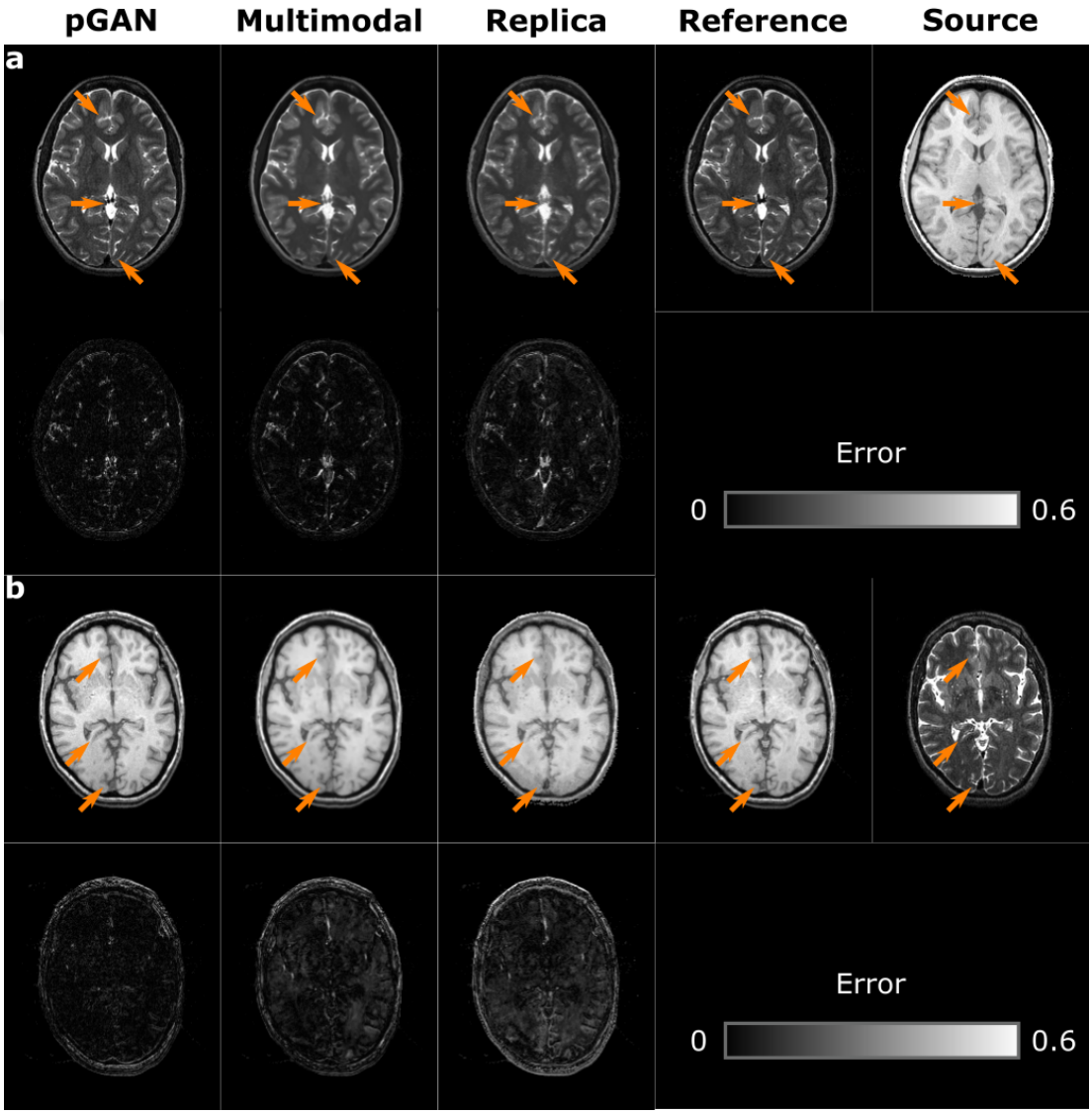

Figure 4.5: The proposed approach was demonstrated for synthesis of T_2-weighted images and T_1-weighted images in the MIDAS dataset. $T_1 \rightarrow T_2\#$ and $T_2\# \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2\#$, and (b) $T_2\# \rightarrow T_1$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions near issue boundaries that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring.

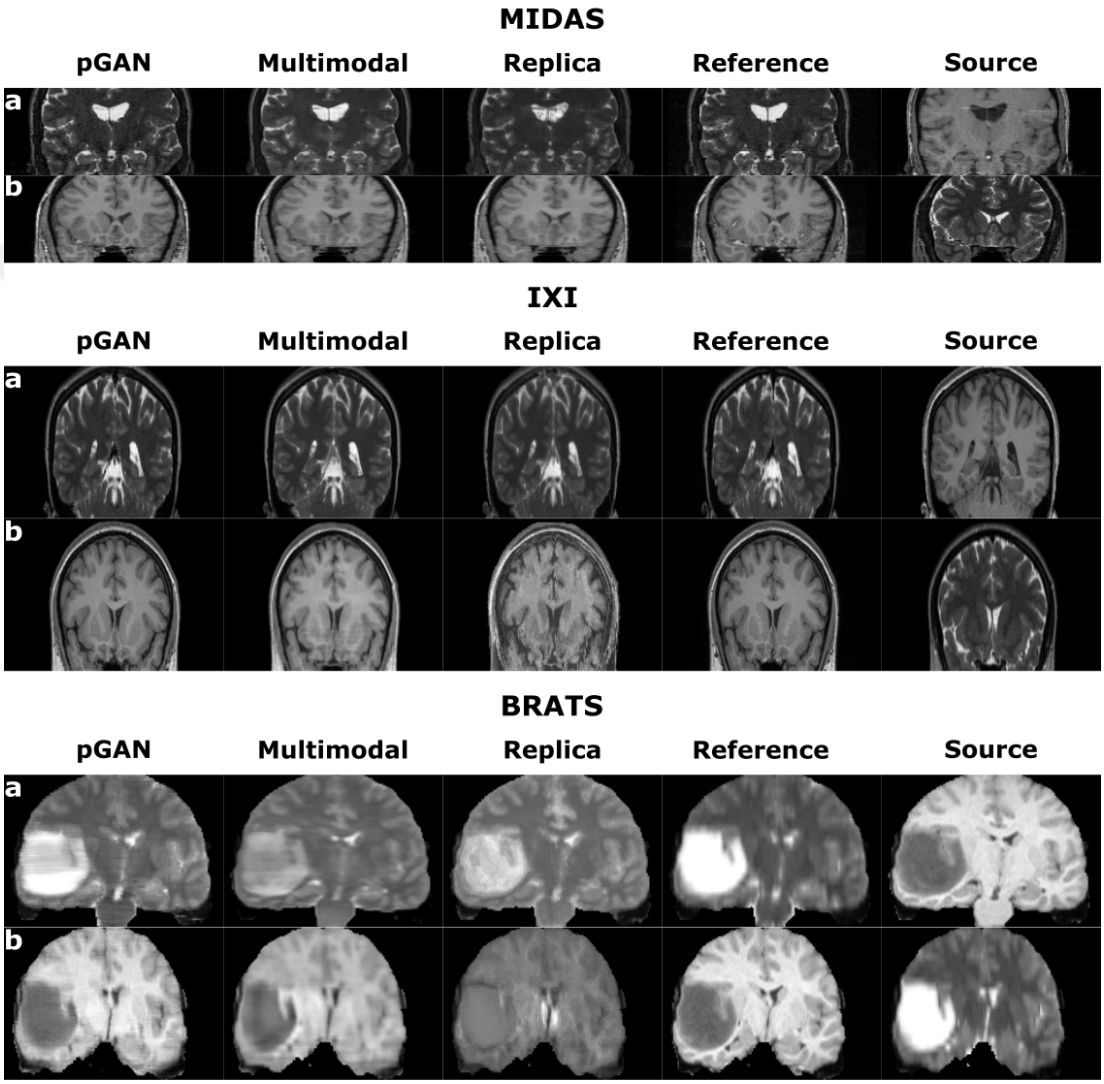

Figure 4.6: The proposed approach was demonstrated for synthesis of T_2-weighted and T_1-weighted images in the all datasets. $T_1 \rightarrow T_2$ and $T_2 \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_2 \rightarrow T_1$ are shown along with the true target image (reference) and the source image (source) for each dataset. The proposed method outperforms competing methods in terms of synthesis quality. The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in synthesized images compared to Multimodal and Replica that show some degree of blurring.

yield enhanced synthesis quality in all cases. Overall, k=3 offers optimal or near-optimal performance while maintaining relatively low model complexity, so k=3 was considered thereafter for pGAN. The results are more variable for cGAN, with the multi-cross section model yielding a modest improvement only in some cases. To minimize model complexity, k=1 was considered for cGAN. Table 4.5 compares PSNR and SSIM of multi cross-section pGAN and cGAN models for T_2 and T_1 synthesis in the MIDAS dataset. Representative results for $T_1 \rightarrow T_2\#$ are shown in Fig. 4.3b and $T_2\# \rightarrow T_1$ are shown in Fig. 4.4b. Among multi cross-section models, pGAN outperforms alternatives in PSNR and SSIM ($p<0.05$), except for SSIM in $T_2\# \rightarrow T_1$. Moreover, compared to the single cross-section pGAN, the multi cross-section pGAN improves PSNR and SSIM values. These measurements are also affirmed by improvements in visual quality for the multi cross-section model in Fig. 4.3 and Fig. 4.4. In contrast, the benefits are less clear for cGAN. Note that, unlike pGAN that works on paired images, the discriminators in cGAN work on unpaired images from the source and target domains. In turn, this can render incorporation of correlated information across cross sections less effective. Tables 4.3 and 4.4 compare PSNR and SSIM of multi cross-section pGAN and cGAN models for T_2 and T_1 synthesis in the IXI dataset. The multi cross-section pGAN outperforms $cGAN_{reg}$ in all cases ($p<0.05$). Moreover, the multi cross-section pGAN outperforms the single cross-section pGAN in all cases ($p<0.05$), except in $T_1 \rightarrow T_2\#$. On average, across the two datasets, multi cross-section pGAN achieves 0.63dB higher PSNR and 0.89% higher SSIM compared to single cross-section pGAN.

4.3.2 Comparison to state-of-the-art methods

Next, we demonstrated the proposed methods against two state-of-the-art techniques for multi-contrast MRI synthesis, Replica and Multimodal. We trained pGAN, $cGAN_{reg}$, Replica, and Multimodal on T_1- and T_2-weighted brain images in the MIDAS and IXI datasets. Note that Replica performs ensemble averaging across random forest trees and Multimodal uses mean-squared error measures

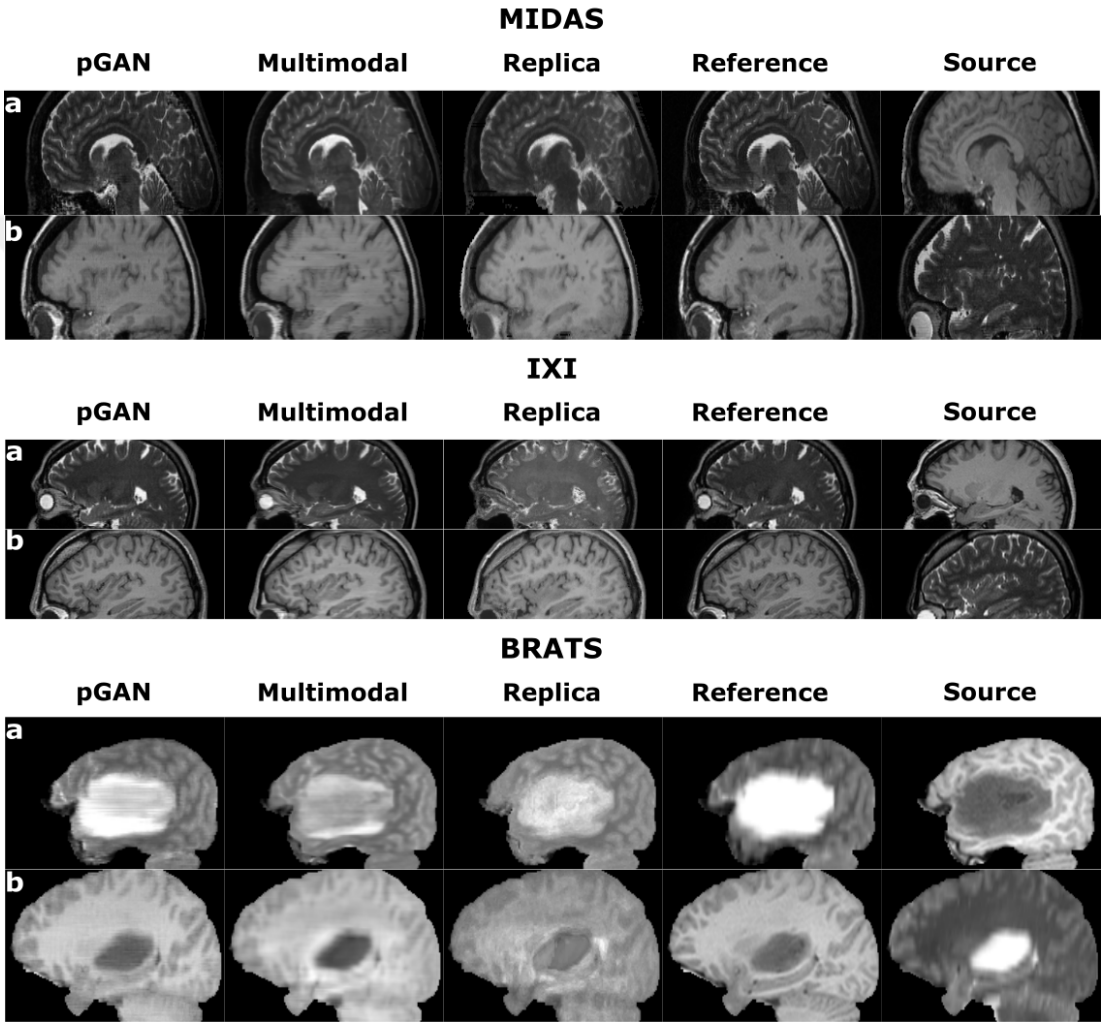

Figure 4.7: The proposed approach was demonstrated for synthesis of T_2-weighted and T_1-weighted images. $T_1 \rightarrow T_2$ and $T_2 \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_2 \rightarrow T_1$ are shown along with the true target image (reference) and the source image (source) for each dataset. The proposed method outperforms competing methods in terms of synthesis quality. The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in synthesized images compared to Multimodal and Replica that show some degree of blurring.

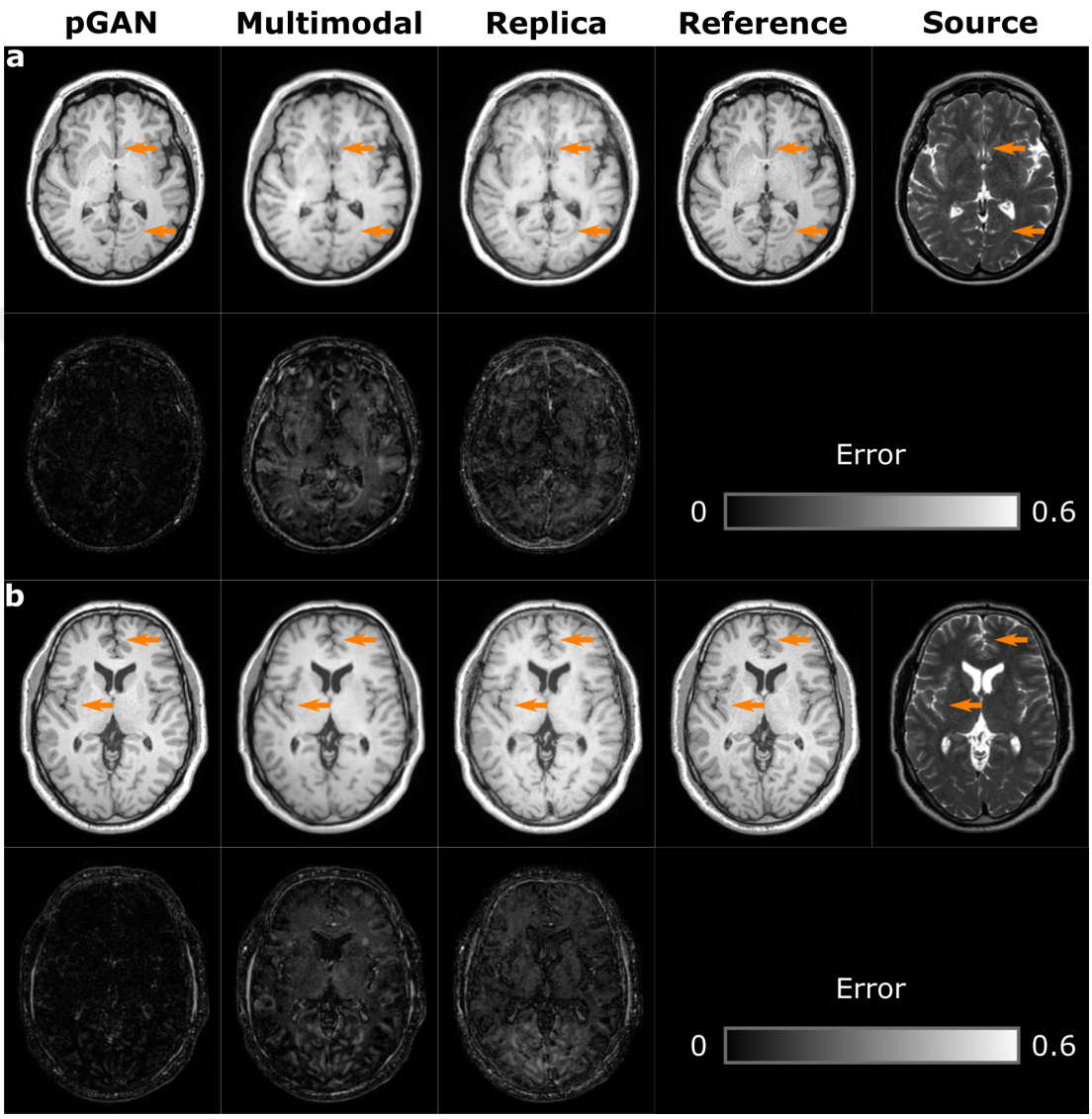

Figure 4.8: The proposed approach was demonstrated for synthesis of T_1-weighted images from T_2-weighted images in the IXI dataset. $T_2 \rightarrow T_1\#$ and $T_2\# \rightarrow T_1$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_2 \rightarrow T_1\#$, and (b) $T_2\# \rightarrow T_1$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring.

that can lead to overemphasis of low frequency information. In contrast, conditional GANs use loss functions that can more effectively capture details in the intermediate to high spatial frequency range. Thus, pGAN should synthesize sharper and more realistic images as compared to the competing methods. Table 4.6 lists PSNR and SSIM for pGAN, Replica and Multimodal ($cGAN_{reg}$ listed in Table 4.7) in the MIDAS dataset. Overall, pGAN outperforms the competing methods in all examined cases ($p<0.05$), except for SSIM in T_2 synthesis, where pGAN and Multimodal perform similarly. The proposed method is superior in depiction of detailed tissue structure as visible in Fig. 4.5 (for comparisons in coronal and sagittal cross-sections see Figs. 4.6, 4.7). Table 4.8 lists PSNR and SSIM across test images synthesized via pGAN, Replica and Multimodal ($cGAN_{reg}$ listed in Table 4.7) for the IXI dataset. Overall, pGAN outperforms the competing methods in all examined cases ($p<0.05$). The proposed method is superior in depiction of detailed tissue structure as visible in Fig. 4.8 and Fig. 4.9 (see also Figs. 4.6,4.7).

Following assessments on datasets comprising healthy subjects, we demonstrated the performance of the proposed methods on patients with pathology. To do this, we trained and tested pGAN, $cGAN_{reg}$, Replica, and Multimodal on T_1- and T_2-weighted brain images from the BRATS dataset. Similar to the previous evaluations, here we expected that the proposed method would synthesize more realistic images with improved preservation of fine-grained tissue structure. Table 4.9 lists PSNR and SSIM across test images synthesized via pGAN, Replica and Multimodal ($cGAN_{reg}$ listed in Table 4.7; for measurements on background-removed images in MIDAS, IXI and BRATS see Table 4.10). Overall, pGAN is the top performing method in all cases ($p<0.05$), except for SSIM in $T_1 \rightarrow T_2$ where pGAN and Multimodal perform similarly. Moreover, cGAN performs favorably in PSNR over competing methods. Representative images for T_2 and T_1 synthesis are displayed in Fig. 4.10 (see also Figs. 4.6, 4.7). It is observed that regions near pathologies are inaccurately synthesized by Replica and Multimodal. Meanwhile, the pGAN method enables reliable synthesis with visibly improved depiction of structural details. Across the datasets, pGAN outperforms the state-of-the-art methods by 2.85dB PSNR and 1.23% SSIM.

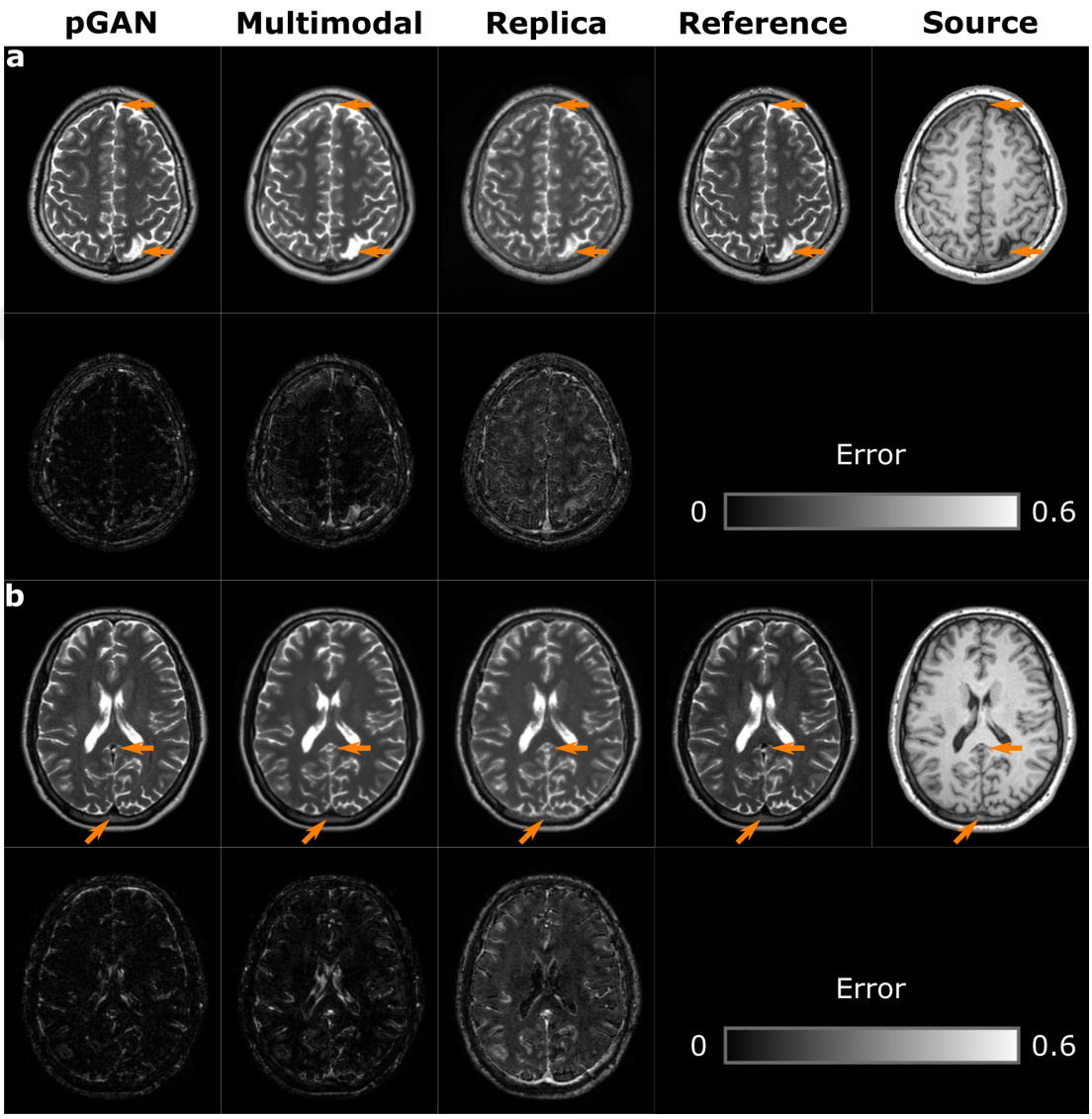

Figure 4.9: The proposed approach was demonstrated for synthesis of T_2-weighted images from T_1-weighted images in the IXI dataset. $T_1\# \rightarrow T_2$ and $T_1 \rightarrow T_2\#$ synthesis were performed with pGAN, Multimodal and Replica. Synthesis results for (a) $T_1\# \rightarrow T_2$, and (b) $T_1 \rightarrow T_2\#$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). The proposed method outperforms competing methods in terms of synthesis quality. Regions near issue boundaries that are inaccurately synthesized by the competing methods are reliably depicted by pGAN (marked with arrows). The use of adversarial loss enables improved accuracy in synthesis of intermediate-spatial-frequency texture in T_2-weighted images compared to Multimodal and Replica that show some degree of blurring.

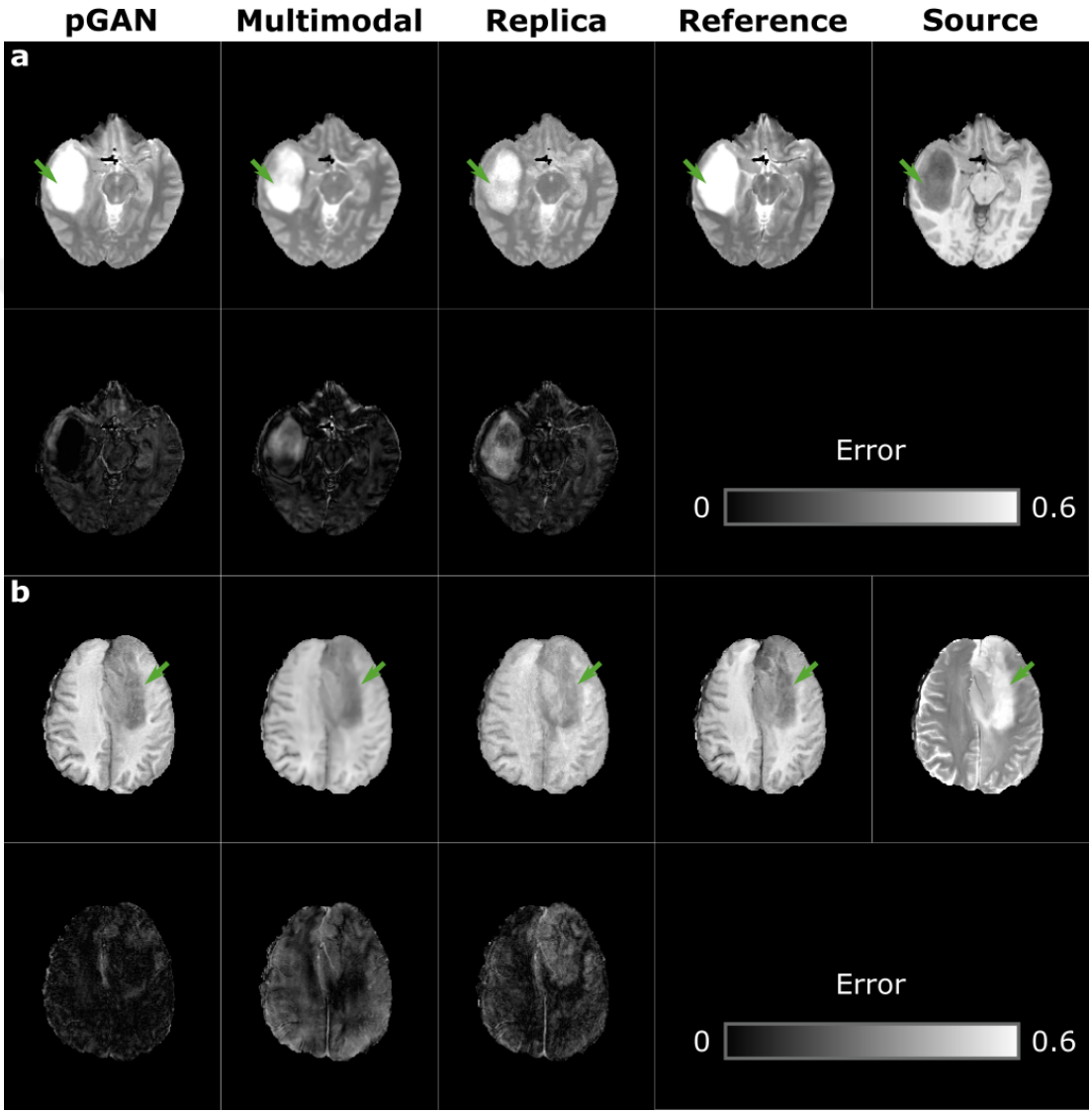

Figure 4.10: The proposed approach was demonstrated on glioma patients for synthesis of T_2-weighted images from T_1-weighted images, and T_2-weighted images from T_1-weighted images in the BRATS dataset. Synthesis results for (a) $T_1 \rightarrow T_2$, and (b) $T_1 \rightarrow T_2$ along with their corresponding error maps are shown along with the true target image (reference) and the source image (source). Regions of inaccurate synthesis with Replica and Multimodal are observed near pathologies (marked with arrows). Meanwhile, the pGAN method enables reliable synthesis with visibly improved depiction of intermediate spatial frequency information.

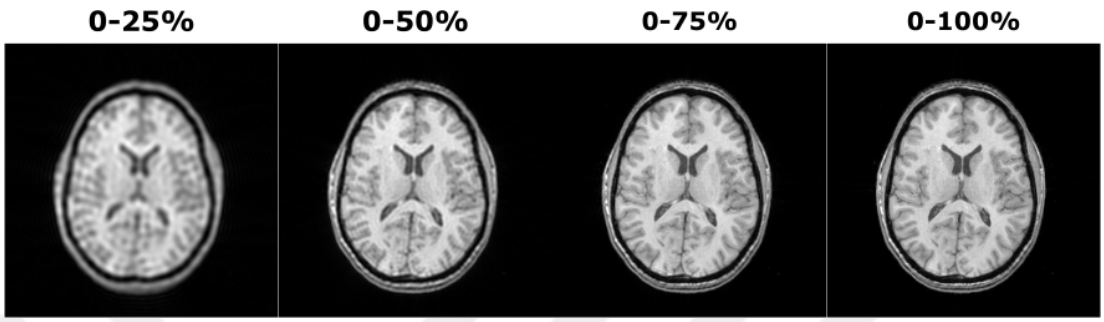

Figure 4.11: The T_1-weighted image of a sample cross-section from the MIDAS dataset was processed with an ideal filter in k-space. The filter was broadened sequentially to include higher frequencies (0-25%, 0-50%, 0-75%, 0-100% of the maximum spatial frequency). The filtered images respectively show the contribution of low, intermediate, high-intermediate and high frequency bands. The bulk shape and contrast of the imaged object is captured in the low frequency band, whereas the fine structural details such as edges are captured in the intermediate and partly high-intermediate frequency bands. There is no apparent contribution from the high frequency band.

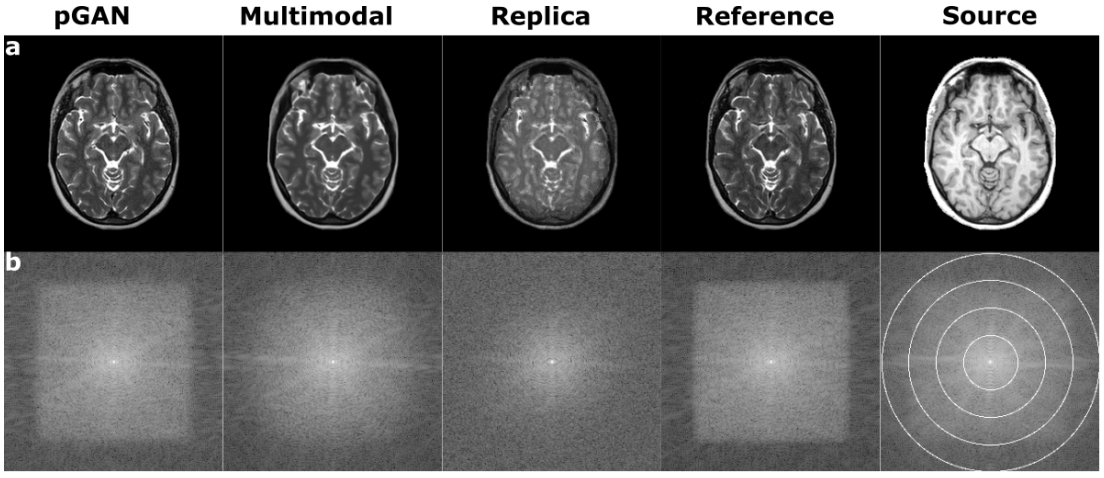

Figure 4.12: Synthesis results are shown for a sample cross section from the IXI dataset along with the true target (reference) and the source image (source). Images shown in (a) the spatial domain (b) the spatial-frequency (k-space) domain. White circular boundaries in the k-space representation of the source delineate the boundaries of the low, intermediate, high-intermediate and high frequency bands. The pGAN method more accurately synthesizes the target image as evidenced by the better match in energy distribution across k-space

Next, we performed additional control analyses via 4-fold cross validation to rule out potential biases due to subject selection. Tables 4.11-4.13 list PSNR and SSIM across test images synthesized via pGAN and Multimodal separately for all 4 folds. We find that there is minimal variability in pGAN performance across folds. Across the datasets, pGAN variability is merely 0.70% in PSNR and 0.37% in SSIM, compared to Multimodal variability of 2.26% in PSNR and 0.46% in SSIM. The results of these control analyses are also highly consistent with those in the original set of subjects reported in Table 4.7. We find that there is minimal variability in pGAN performance between the main and control analyses. Across the datasets, pGAN variability is 1.42% in PSNR and 0.73% in SSIM, compared to Multimodal variability of 2.98% in PSNR and 0.97% in SSIM.

4.3.3 Spectral density analysis

To corroborate visual observations regarding improved depiction of structural details, we measured spectral density similarity (SDS) between synthesized and reference images across low, intermediate, high-intermediate and high spatial frequencies (see Methods). Fig. 4.11 shows filtered versions of a T_1-weighted image in the MIDAS dataset, where the filter is broadened sequentially to include higher frequencies so as to visualize the contribution of individual bands. Intermediate and high-intermediate frequencies primarily correspond to edges and other structural details in MR images, so we expected pGAN to outperform competing methods in these bands. Fig. 4.12 shows representative synthesis results in the image and spatial frequency (k-space) domains. In the MIDAS dataset, pGAN outperforms the competing methods at low and intermediate frequencies ($p<0.05$), except in T_1 synthesis where it performs similarly to Multimodal. In the IXI dataset, pGAN yields superior performance to competing methods in all frequency bands ($p<0.05$). In the BRATS dataset, pGAN achieves higher SDS than the competing methods at low, intermediate and high-intermediate frequencies in T_2 synthesis and at low frequencies in T_1 synthesis ($p<0.05$). Across the datasets, pGAN outperforms the state-of-the-art methods by 0.056 at low, 0.061 at intermediate and 0.030 at high-intermediate frequencies.

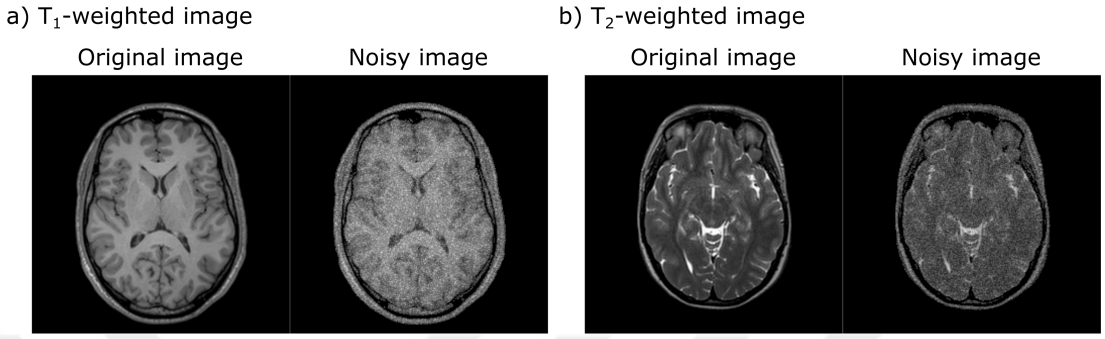

Figure 4.13: To examine reliability of the synthesis methods against noise, pGAN was trained on noisy images generated from the IXI dataset by adding artificial Rician noise. A sample cross-section from T_1-weighted (a) and T_2-weighted (b) acquisitions along with their noise-added versions are displayed.

4.3.4 Generalizability

Next, we examined synthesis methods in terms of their generalization performance. Table 4.14 lists SSIM and PSNR for pGAN, $cGAN_{reg}$, Replica and Multimodal trained on the IXI dataset and tested on the MIDAS dataset. Overall, the proposed methods are the top performers. In $T_1 \rightarrow T_2\#$, Multimodal is the leading performer with 1.9% higher SSIM SSIM ($p<0.05$) than pGAN. In $T_1\# \rightarrow T_2$, pGAN outperforms competing methods in PSNR ($p<0.05$). In $T_2 \rightarrow T_1\#$, pGAN is again the leading performer with 1.9% higher SSIM ($p<0.05$) than Multimodal. In $T_2\# \rightarrow T_1$, $cGAN_{reg}$ is the leading performer with 1.22dB higher PSNR ($p<0.05$) SSIM than pGAN. We also assessed the level of performance degradation between within-dataset synthesis (trained and tested on MIDAS) and across-dataset synthesis (trained on IXI, tested on MIDAS). Overall, pGAN and Multimodal show similar degradation levels. While pGAN is the top performer in terms of SSIM, cGAN yields a modest advantage in PSNR. On average, percentage degradation is 20.83% in PSNR and 11.70% in SSIM for pGAN, 22.22% in PSNR and 10.12% in SSIM for Multimodal, 15.85% in PSNR and 12.85% in SSIM for $cGAN_{reg}$, and 11.40% in PSNR and 14.51% in SSIM for Replica. Note that percentage degradation in PSNR is inherently limited for Replica, which yields low PSNR for within-dataset synthesis.

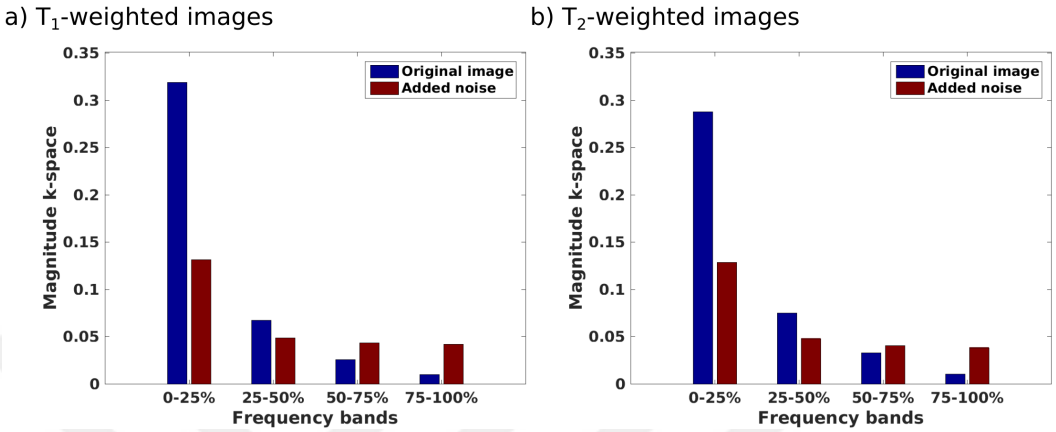

Figure 4.14: Spectral distribution of k-space intensity values was examined for T_1-weighted (a) and T_2-weighted (b) images in the IXI dataset relative to the artificial noise added to the images. Average k-space intensity was calculated within low (0-25%), intermediate (25-50%), high-intermediate (50-75%) and high (75-100%) spatial frequency bands. Bar plots show the results averaged across the test set. The contribution of added noise to low and intermediate frequency bands is small relative to the original image. In contrast, noise more dominantly contributes to high-intermediate and high frequency bands.

4.3.5 Reliability against noise

Lastly, we examined reliability of synthesis against noise (Fig. 4.13). Table 4.15 list SSIM and PSNR for pGAN and Multimodal trained on noise-added source and target images from IXI, respectively. For noisy source images, pGAN outperforms Multimodal in all examined cases ($p<0.05$) except for SSIM in $T_1T_2\#$. On average, pGAN achieves 1.74dB higher PSNR and 2.20% higher SSIM than Multimodal. For noisy target images, pGAN is the top performer in PSNR in $T_1\# \rightarrow T_2$, $T_2 \rightarrow T_1\#$ ($p<0.05$) and performs similarly to Multimodal in the remaining cases. On average, pGAN improves PSNR by 0.61dB. (Note, however, that for noisy target images, reference-based quality measurements are biased by noise particularly towards higher frequency bands; see Fig. 4.14) Naturally, synthesis performance is lowered in the presence of noise. We assessed the performance degradation when the models were trained on noise-added images as compared to when the models were trained on original images. Overall, pGAN and Multimodal show similar performance degradation with noise. For noisy

source images, degradation is 5.27% in PSNR and 2.17% in SSIM for pGAN, and 3.77% in PSNR, 2.66% in SSIM for Multimodal. For noisy target images, degradation is 16.70% in PSNR and 12.91% in SSIM for pGAN, and 15.19% in PSNR, 10.06% in SSIM for Multimodal.

4.4 Discussion

A multi-contrast MRI synthesis approach based on conditional GANs was demonstrated against state-of-the-art methods in three publicly available brain MRI datasets. The proposed pGAN method uses adversarial loss functions and correlated structure across neighboring cross-sections for improved synthesis. While many previous methods require registered multi-contrast images for training, a cGAN method was presented that uses cycle-consistency loss for learning to synthesize from unregistered images. Comprehensive evaluations were performed for two distinct scenarios where training images were registered and unregistered. Overall, both proposed methods yield synthetic images of remarkable visual similarity to reference images, and pGAN visually and quantitatively improves synthesis quality compared to state-of-the-art methods [25, 132]. These promising results warrant future studies on broad clinical populations to fully examine diagnostic quality of synthesized images in pathological cases.

Several previous studies proposed the use of neural networks for multi-contrast MRI synthesis tasks [127, 24, 26, 25, 133]. A recent method, Multimodal, was demonstrated to yield higher quality compared to conventional methods in brain MRI datasets [25]. Unlike conventional neural networks, the GAN architectures proposed here are generative networks that learn the conditional probability distribution of the target contrast given the source contrast. The incorporation of adversarial loss as opposed to typical squared or absolute error loss leads to enhanced capture of detailed texture information about the target contrast, thereby enabling higher synthesis quality. While our synthesis approach was primarily demonstrated for multi-contrast brain MRI here, architectures similar to pGAN and cGAN have been proposed in other medical

image synthesis applications such as cross-modality synthesis or data augmentation [137, 138, 142, 143, 145, 147, 148, 149, 150, 151, 154]. The discussions below highlight key differences between the current study and previous work: (1) [138, 149, 151, 154] proposed conditional GANs for cross-modality synthesis applications. One important proposed application is CT to PET synthesis [138, 149]. For instance, [138] fused the output of GANs and convolutional networks to enhance tumor detection performance from synthesized images; and [149] demonstrated competitive tumor detection results from synthesized versus real images. Another important application is MR to CT synthesis [151, 154]. In [151, 154] patch-based GANs were used for locally-aware synthesis, and contextual information was incorporated by training an ensemble of GAN models recurrently. Our approach differs in the following aspects: (i) Rather than cross-modality image synthesis, we focus on within-modality synthesis in multi-contrast MRI. MRI provides excellent delineation among soft tissues in the brain and elsewhere, with the diversity of contrasts that it can capture [170, 171, 172, 173, 174, 175]. Therefore, synthesizing a specific MRI contrast given another poses a different set of challenges than performing MR-CT or CT-PET synthesis where CT/PET shows relatively limited contrast among soft tissues [40]. (ii) We demonstrate multi-cross section models to leverage correlated information across neighboring cross-sections within a volume. (iii) We demonstrate pGAN based on both pixel-wise and perceptual losses to enhance synthesis quality. (2) Architectures similar to cGAN with cycle-consistency loss were recently proposed to address the scarcity of paired training data in MR-CT synthesis tasks [137, 142, 145, 147, 148]. [142] also utilized a gradient-consistency loss to enhance the segmentation performance on CT images synthesized from MR data. [145] performed data-augmentation for enhanced segmentation performance using MR images synthesized from CT data. [148] coupled synthesis and segmentation networks to perform improved segmentation on synthesized CT images using MR labels. Our work differs in the following aspects: (i) As aforementioned, we consider within-modality synthesis as opposed to cross-modality synthesis. (ii) We consider paired image synthesis with cGAN to comparatively evaluate its performance against two state-of-the-art methods (Replica and Multimodal) for paired image synthesis. (3) An architecture resembling pGAN was proposed for synthesizing retinal images acquired

with fundus photography given tabular structural annotations [150]. Similar to pGAN, this previous study incorporated a perceptual loss to improve synthesis quality. Our work differs in the following aspects: (i) Synthesis of vascular fundus images in the retina given annotations is a distinct task than synthesis of a target MR contrast given another source MR contrast in the brain. Unlike the relatively focused delineation between vascular structures and background in retinal images, in our case, there are multiple distinct types of brain tissues that appear at divergent signal levels in separate MR contrasts [40]. (ii) We demonstrate multi-cross section models to leverage correlated information across neighboring cross-sections within an MRI volume. (4) A recent study suggested the use of multiple cross-sections during MR-to-CT synthesis [176]. In comparison to [176], our approach is different in that: (i) We incorporate an adversarial loss function to better preserve intermediate-to-high frequency details in the synthesized images. (ii) We perform task- and model-specific optimization of the number of cross-section considering both computational complexity and performance. (iii) As aforementioned, we consider within-modality synthesis as opposed to cross-modality synthesis. Few recent studies have independently proposed GAN models for multi-contrast MRI synthesis [29, 177, 178]. Perhaps, the closest to our approach are [29] and [177] where conditional GANs with pixel-wise loss were used for improved segmentation based on synthesized FLAIR, T_1- and T_2-weighted images. Our work differs from these studies in the following aspects: (i) We demonstrate improved multi-contrast MRI synthesis via cycle-consistency loss to cope with un-registered images. (ii) We demonstrate improved multi-contrast synthesis performance via the inclusion of a perceptual loss to pGAN. (iii) We demonstrate multiple cross-section models to leverage correlated information across neighboring cross-sections within multi-contrast MRI volumes. (iv) We quantitatively demonstrate that conditional GANs better preserve detailed tissue structure in synthesized multi-contrast images compared to conventional methods [25, 132].

The proposed approach might be further improved by considering several lines of development. Here we presented multi-contrast MRI results while considering two potential directions for image registration ($T_1 \rightarrow T_2$# and T_1# $\rightarrow T_2$ for T_2

synthesis). We observed that the proposed methods yielded high-quality synthesis regardless of the registration direction. Comparisons between the two directions based on reference-based metrics are not informative because the references are inevitably distinct (e.g., T_2# versus T_2), so determining the optimal direction is challenging. Yet, with substantial mismatch between the voxel sizes in the source and target contrasts, the cGAN method learns to interpolate between the spatial sampling grids of the source and the target. To alleviate performance loss, a simple solution is to resample each contrast separately to match the voxel dimensions. Alternatively, the spatial transformation between the source and target images can first be estimated via multi-modal registration [179]. The estimated transformation can then be cascaded to the output of cGAN. A gradient cycle consistency loss can also be incorporated to prevent the network from learning the spatial transformation between the source and the target [142]. Another cause for performance loss arises when MR images for a given contrast are corrupted by higher levels of noise than typical. Our analyses on noise-added images imply a certain degree of reliability against moderate noise in T_1- or T_2-weighted images. However, an additional denoising network could be incorporated to earlier layers in GAN models when source images have higher noise, and to later layers when target images have elevated noise [177].

Synthesis accuracy can also be improved by generalizing the current approach to predict the target based on multiple source contrasts. In principle, both pGAN and cGAN can receive as input multiple source contrasts in addition to multiple cross sections as demonstrated here. In turn, this generalization can offer improved performance when a subset of the source contrast is unavailable. The performance of conditional GAN architectures in the face of missing inputs warrants further investigation. Alternatively, an initial fusion step can be incorporated that combines multi-contrast source images in the form of a single fused image fed as input to the GAN [180].

Our analyses on noise-added images indicate that, for target contrasts that are inherently noisier, a downweighing of perceptual loss might be necessary. The proposed models include a hyperparameter for adjusting the relative weighing

of the perceptual loss against other loss terms. Thus, a cross-validation procedure can be performed for the specific set of source-target contrasts at hand to optimize model parameters. It remains important future work to assess the optimal weighing of perceptual loss as a function of noise level for specific contrasts. Alternatively, denoising can be included as a preprocessing step to improve reliability against noise. Note that such denoising has recently been proposed for learning-based sampling pattern optimization in MRI [181].

An important concern regarding neural-network based methods is the availability of large datasets for successful training. The cGAN method facilitates network training by permitting the use of unregistered and unpaired multi-contrast datasets. While here we performed training on paired images for unbiased comparison, cGAN permits the use of unpaired images from distinct sets of subjects. As such, it can facilitate compilation of large datasets that would be required for improved performance via deeper networks. Yet, further performance improvements may be viable by training networks based on a mixture of paired and unpaired training data [23].

Recently, cross-modality synthesis with GANs was leveraged as a preprocessing step to enhance various medical imaging tasks such as segmentation, classification or tumor detection [138, 142, 145, 148, 149, 182, 183]. For instance, [138] fused the output of GANs and convolutional networks to enhance tumor detection from synthesized PET images, and [149] demonstrated competitive detection performance with real versus synthesized PET images. [142] trained GANs based on cycle-consistency loss to enhance segmentation performance from synthesized CT images. [145] showed that incorporating synthesized MR images with the real ones can improve the performance of a segmentation network [148]. GANs also showed enhanced performance in liver lesion classification in synthetic CT [182], and chest pathology classification in synthetic X-ray images [183]. These previous reports suggest that the multi-contrast MRI synthesis methods proposed here might also improve similar post-processing tasks. It remains future work to assess to what extent improvements in synthesis quality translate to tasks such as segmentation or detection.

4.5 Conclusion

We proposed a new multi-contrast MRI synthesis method based on conditional generative adversarial networks. Unlike most conventional methods, the proposed method performs end-to-end training of GANs that synthesize the target contrast given images of the source contrast. The use of adversarial loss functions improves accuracy in synthesis of detailed structural information in the target contrast. Synthesis performance is further improved by incorporating pixel-wise and perceptual losses in the case of registered images, and a cycle-consistency loss for unregistered images. Finally, the proposed method leverages information across neighboring cross-sections within each volume to increase accuracy of synthesis. The proposed method outperformed state-of-the-art synthesis methods in multi-contrast brain MRI datasets from healthy subjects and glioma patients. Given the prohibitive costs of prolonged exams due to repeated acquisitions, only a subset contrasts might be collected with adequate quality, particularly in pediatric and elderly patients and in large cohorts [116, 118]. Multi-contrast MRI synthesis might be helpful in those worst-case situations by offering a substitute for highly-corrupted or even unavailable contrasts. Therefore, our GAN-based approach holds great promise for improving the diagnostic information available in clinical multi-contrast MRI.

Table 4.1: Quality of Synthesis: Perceptual loss

	pGAN (with perceptual loss)		**pGAN (w/o perceptual loss)**	
MIDAS Dataset				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.926±0.014	29.34±0.592	0.917±0.013	28.71±0.513
$T_1\# \rightarrow T_2$	0.883±0.027	27.49±0.643	0.872±0.026	26.91±0.552
$T_2 \rightarrow T_1\#$	0.920±0.016	28.16±1.303	0.909±0.017	27.69±1.418
$T_2\# \rightarrow T_1$	0.887±0.023	27.42±1.127	0.874±0.023	26.80±0.924
IXI Dataset				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.948±0.014	29.77±1.568	0.942±0.014	29.20±1.373
$T_1\# \rightarrow T_2$	0.917±0.012	27.89±0.887	0.905±0.012	27.31±0.831
$T_2 \rightarrow T_1\#$	0.926±0.013	27.27±0.960	0.915±0.012	26.75±0.830
$T_2\# \rightarrow T_1$	0.953±0.012	29.55±1.423	0.948±0.013	29.23±1.316

Synthesis performance for T_1- and T_2-weighted images from MIDAS and IXI datasets was evaluated for pGAN trained with perceptual loss and pGAN trained without perceptual loss. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). SSIM and PSNR measurements are reported as mean±std across test subjects.

Table 4.2: Quality of Synthesis in the MIDAS dataset: Single cross-section models

	$cGAN_{unreg}$		**$cGAN_{reg}$**		**pGAN**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.829±0.017	23.66±0.632	0.895±0.014	26.56±0.432	0.920±0.014	28.79±0.580
$T_1\# \rightarrow T_2$	0.823±0.021	23.85±0.420	0.854±0.024	25.47±0.556	0.876±0.028	27.07±0.618
$T_2 \rightarrow T_1\#$	0.826±0.015	23.20±0.503	0.892±0.017	26.53±1.169	0.912±0.017	27.81±1.424
$T_2\# \rightarrow T_1$	0.821±0.021	22.56±1.008	0.863±0.022	26.15±0.974	0.883±0.023	27.31±0.983

$T_1\#$ is registered onto the respective T_2 image; and $T_2\#$ is registered onto the respective T_1 image; and $\rightarrow$ indicates the direction of synthesis. PSNR and SSIM measurements are reported as mean±std across test images.

Table 4.3: Quality of Synthesis: Number of cross-sections for pGAN

MIDAS Dataset								
	k=7		**k=5**		**k=3**		**k=1**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.928±0.013	29.47±0.566	0.927±0.014	29.43±0.552	0.926±0.014	29.34±0.592	0.920±0.014	28.79±0.580
$T_1\# \rightarrow T_2$	0.884±0.027	27.56±0.642	0.884±0.026	27.61±0.630	0.883±0.027	27.49±0.643	0.876±0.028	27.07±0.618
$T_2 \rightarrow T_1\#$	0.921±0.017	28.24±1.588	0.921±0.016	28.23±1.360	0.920±0.016	28.16±1.303	0.912±0.017	27.81±1.424
$T_2\# \rightarrow T_1$	0.888±0.022	27.51±1.074	0.888±0.022	27.55±1.130	0.887±0.023	27.42±1.127	0.883±0.023	27.31±0.983
IXI Dataset								
	k=7		**k=5**		**k=3**		**k=1**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.948±0.014	29.77±1.568	0.924±0.016	27.06±0.865	0.912±0.028	25.40±2.084	0.936±0.015	27.72±0.910
$T_1\# \rightarrow T_2$	0.917±0.012	27.89±0.887	0.891±0.013	26.10±0.726	0.863±0.023	24.08±1.427	0.898±0.014	26.11±0.769
$T_2 \rightarrow T_1\#$	0.926±0.013	27.27±0.960	0.894±0.014	25.89±0.704	0.865±0.013	20.46±0.921	0.895±0.015	22.61±1.105
$T_2\# \rightarrow T_1$	0.953±0.012	29.55±1.423	0.922±0.016	26.74±0.828	0.887±0.033	21.82±1.600	0.936±0.017	25.91±1.689

Performance of multiple cross-section pGAN models (k = 1, 3, 5 and 7) was evaluated in synthesis of T_1- and T_2-weighted images from MIDAS and IXI datasets. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). SSIM and PSNR measurements are reported as mea±std across test subjects.

Table 4.4: Quality of Synthesis: Number of cross-sections for $cGAN_{reg}$

MIDAS Dataset				
	k=3		**k=1**	
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.895±0.014	26.62±0.489	0.895±0.014	26.56±0.432
$T_1\# \rightarrow T_2$	0.862±0.022	25.83±0.384	0.854±0.024	25.47±0.556
$T_2 \rightarrow T_1\#$	0.900±0.017	27.04±1.238	0.892±0.017	26.53±1.169
$T_2\# \rightarrow T_1$	0.864±0.022	26.44±0.871	0.863±0.022	26.15±0.974
IXI Dataset				
	k=3		**k=1**	
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.865±0.017	23.33±0.551	0.924±0.016	27.06±0.865
$T_1\# \rightarrow T_2$	0.902±0.010	26.84±0.565	0.891±0.013	26.10±0.726
$T_2 \rightarrow T_1\#$	0.897±0.014	26.09±0.819	0.894±0.014	25.89±0.704
$T_2\# \rightarrow T_1$	0.927±0.015	27.35±0.982	0.922±0.016	26.74±0.828

Performance of multiple cross-section $cGAN_{reg}$ models (k = 1 and 3) was evaluated in synthesis of T_1- and T_2-weighted images from MIDAS and IXI datasets. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). SSIM and PSNR measurements are reported as mean±std across test subjects.

Table 4.5: Quality of Synthesis in the MIDAS dataset: Multiple cross-section models (k=3)

	$cGAN_{unreg}$		**$cGAN_{reg}$**		**pGAN**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.829±0.016	23.65±0.650	0.895±0.014	26.62±0.489	0.926±0.014	29.34±0.592
$T_1\# \rightarrow T_2$	0.797±0.027	23.37±0.604	0.862±0.022	25.83±0.384	0.883±0.027	27.49±0.643
$T_2 \rightarrow T_1\#$	0.824±0.015	24.00±0.628	0.900±0.017	27.04±1.238	0.920±0.016	28.16±1.303
$T_2\# \rightarrow T_1$	0.805±0.021	23.55±0.782	0.864±0.022	26.44±0.871	0.887±0.023	27.42±1.127

$T_1\#$ is registered onto the respective T_2 image; and $T_2\#$ is registered onto the respective T_1 image; and $\rightarrow$ indicates the direction of synthesis. PSNR and SSIM measurements are reported as mean±std across test images.

Table 4.6: Quality of Synthesis in the MIDAS dataset

	pGAN		**Replica**		**Multimodal**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.926±0.014	29.34±0.592	0.877±0.027	26.18±0.638	0.924±0.012	28.33±0.501
$T_1\# \rightarrow T_2$	0.883±0.027	27.49±0.643	0.838±0.039	25.27±0.468	0.889±0.020	26.73±0.461
$T_2 \rightarrow T_1\#$	0.920±0.016	28.16±1.303	0.840±0.028	20.00±1.207	0.886±0.022	22.13±1.325
$T_2\# \rightarrow T_1$	0.887±0.023	27.42±1.127	0.827±0.031	20.29±1.066	0.872±0.020	23.08±1.280

$T_1\#$ is registered onto the respective T_2 image; and $T_2\#$ is registered onto the respective T_1 image; and $\rightarrow$ indicates the direction of synthesis. PSNR and SSIM measurements are reported as mean±std across test images.

Table 4.7: Quality of Synthesis

	pGAN		**cGAN$_{reg}$**		**cGAN$_{unreg}$**		**Replica**		**Multimodal**	
MIDAS Dataset										
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.926±0.014	29.34±0.592	0.895±0.014	26.56±0.432	0.829±0.017	23.66±0.632	0.877±0.027	26.18±0.638	0.924±0.012	28.33 ±0.501
$T_1\# \rightarrow T_2$	0.883±0.027	27.49±0.643	0.854±0.024	25.47±0.556	0.823±0.021	23.85±0.540	0.838±0.039	25.27±0.468	0.889±0.020	26.73±0.461
$T_2 \rightarrow T_1\#$	0.920±0.016	28.16±1.303	0.892±0.017	26.53±1.169	0.826±0.015	23.20±0.503	0.840±0.028	20.00±1.207	0.886±0.022	22.13±1.325
$T_2\# \rightarrow T_1$	0.887±0.023	27.42±1.127	0.863±0.022	26.15±0.974	0.821±0.021	22.56±1.008	0.827±0.031	20.29±1.066	0.872±0.020	23.08±1.280
IXI Dataset										
	pGAN		**Replica**		**Multimodal**					
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR		
$T_1 \rightarrow T_2\#$	0.948±0.014	29.77±1.568	0.924±0.016	27.06±0.865	0.912±0.028	25.40±2.084	0.936±0.015	27.72±0.910		
$T_1\# \rightarrow T_2$	0.917±0.012	27.89±0.887	0.891±0.013	26.10±0.726	0.863±0.023	24.08±1.427	0.898±0.014	26.11±0.769		
$T_2 \rightarrow T_1\#$	0.926±0.013	27.27±0.960	0.894±0.014	25.89±0.704	0.865±0.013	20.46±0.921	0.895±0.015	22.61±1.105		
$T_2\# \rightarrow T_1$	0.953±0.012	29.55±1.423	0.922±0.016	26.74±0.828	0.887±0.033	21.82±1.600	0.936±0.017	25.91±1.689		
BRATS Dataset										
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR		
$T_1 \rightarrow T_2$	0.946±0.009	27.19±1.456	0.927±0.010	25.73±1.431	0.924±0.014	24.64±1.615	0.939±0.011	25.09±1.013		
$T_1 \rightarrow T_2$	0.940±0.009	25.80±1.867	0.913±0.008	24.97±1.179	0.917±0.007	24.49±1.230	0.935±0.010	23.78±2.080		

Model comparison in all datasets. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN, cGAN$_{reg}$, cGAN$_{unreg}$, Replica and Multimodal. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). $T_1\#$ is registered onto the respective T_2 image; and $T_2\#$ is registered onto the respective T_1 image; and $\rightarrow$ indicates the direction of synthesis. SSIM and PSNR measurements are reported as mean±std across test images.

Table 4.8: Quality of Synthesis in the IXI dataset

	pGAN		**Replica**		**Multimodal**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.948±0.014	29.77±1.568	0.912±0.028	25.40±2.084	0.936±0.015	27.72±0.910
$T_1\# \rightarrow T_2$	0.917±0.012	27.89±0.887	0.863±0.023	24.08±1.427	0.898±0.014	26.11±0.769
$T_2 \rightarrow T_1\#$	0.926±0.013	27.27±0.960	0.865±0.013	20.46±0.921	0.895±0.015	22.61±1.105
$T_2\# \rightarrow T_1$	0.953±0.012	29.55±1.423	0.887±0.033	21.82±1.600	0.936±0.017	25.91±1.689

$T_1\#$ is registered onto the respective T_2 image; and $T_2\#$ is registered onto the respective T_1 image; and $\rightarrow$ indicates the direction of synthesis. PSNR and SSIM measurements are reported as mean±std across test images.

Table 4.9: Quality of Synthesis in the BRATS dataset

	pGAN		**Replica**		**Multimodal**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$	0.946±0.009	27.19±1.456	0.924±0.014	24.64±1.615	0.939±0.011	25.09 ±1.013
$T_1 \rightarrow T_2$	0.940±0.009	25.80±1.867	0.917±0.007	24.49±1.230	0.935±0.010	23.78±2.080

$\rightarrow$ indicates the direction of synthesis. PSNR and SSIM measurements are reported as mean±std across test images.

Table 4.10: Quality of Synthesis: Modal Comparison on background removed images

MIDAS Dataset				
	pGAN	$cGAN_{reg}$	Replica	Multimodal
$T_1 \rightarrow T_2\#$	24.62±0.553	21.85±0.535	21.46±0.538	23.61±0.410
$T_1\# \rightarrow T_2$	22.85±0.570	20.84±0.609	20.63±0.353	22.09±0.401
$T_2 \rightarrow T_1\#$	23.48±1.238	21.85±1.094	15.32±1.098	17.44±1.181
$T_2\# \rightarrow T_1$	22.68±1.082	21.41±0.918	18.34±1.179	18.34±1.179
IXI Dataset				
	pGAN	$cGAN_{reg}$	Replica	Multimodal
$T_1 \rightarrow T_2\#$	24.03±1.173	21.32±0.624	19.68±1.908	21.99±0.634
$T_1\# \rightarrow T_2$	23.56±0.992	21.78±0.727	19.76±1.465	21.79±0.768
$T_2 \rightarrow T_1\#$	23.06±1.027	21.68±0.540	16.25±1.012	18.41±0.761
$T_2\# \rightarrow T_1$	23.93±0.911	21.12±0.248	16.20±1.689	20.29±1.046
BRATS Dataset				
	pGAN	$cGAN_{reg}$	Replica	Multimodal
$T_1 \rightarrow T_2\#$	19.63±1.427	18.20±1.394	17.05±1.614	17.51 ±0.947
$T_1\# \rightarrow T_2$	18.22±1.780	17.41±1.148	16.89±1.277	16.20±1.992

Model comparison in all datasets. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN, $cGAN_{reg}$, Replica and Multimodal. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). Whereas the MIDAS and IXI datasets contain untouched images, the BRATS dataset is preprocessed for brain extraction. As a result, BRATS images are free of skull and other non-brain structures and they have broader background regions. To alleviate differences in the extent of background regions, additional PSNR measurements were performed by removing background voxels from the images. PSNR measurements are reported as mean±std across test images.

Table 4.11: Quality of Synthesis in the MIDAS dataset: Control analysis

	pGAN		**Multimodal**	
Fold 1				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.918±0.024	28.73±1.121	0.915±0.015	27.26±1.141
$T_1\# \rightarrow T_2$	0.871±0.030	26.93±0.942	0.873±0.020	25.68±0.832
$T_2 \rightarrow T_1\#$	0.908±0.025	27.47±1.277	0.877±0.021	22.13±1.536
$T_2\# \rightarrow T_1$	0.876±0.029	27.01±1.095	0.849±0.029	19.58±1.317
Fold 2				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.912±0.056	29.21±1.117	0.910±0.046	27.98±0.870
$T_1\# \rightarrow T_2$	0.860±0.086	27.33±1.098	0.869±0.077	26.80±0.912
$T_2 \rightarrow T_1\#$	0.909±0.026	26.74±1.664	0.882±0.028	21.92±1.425
$T_2\# \rightarrow T_1$	0.869±0.045	26.76±1.211	0.862±0.031	22.15±1.350
Fold 3				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.919±0.032	28.94±0.982	0.914±0.029	27.64±0.768
$T_1\# \rightarrow T_2$	0.864±0.067	27.02±1.071	0.870±0.058	26.17±0.876
$T_2 \rightarrow T_1\#$	0.917±0.022	27.39±1.020	0.892±0.023	23.32±1.680
$T_2\# \rightarrow T_1$	0.882±0.025	26.54±0.940	0.873±0.022	22.12±1.546
Fold 4				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.908±0.022	28.21±0.729	0.901±0.017	26.66±0.615
$T_1\# \rightarrow T_2$	0.854±0.040	26.51±0.789	0.866±0.029	25.70±0.540
$T_2 \rightarrow T_1\#$	0.900±0.024	26.43±1.080	0.875±0.028	21.46±1.324
$T_2\# \rightarrow T_1$	0.860±0.040	26.28±1.237	0.838±0.038	20.16±1.415

Control analysis in the MIDAS dataset. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN and Multimodal on 40 subjects from the MIDAS dataset. A 4-fold cross-validation procedure was used by randomly sampling nonoverlapping training, validation and testing sets in each fold such that 25 subjects were used for training, 5 for validation and 10 for testing. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). SSIM and PSNR measurements are reported as mean±std across test images.

Table 4.12: Quality of Synthesis in the IXI dataset: Control analysis

	pGAN		**Multimodal**	
Fold 1				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.954±0.010	30.18±1.268	0.939±0.012	28.04±1.050
$T_1\# \rightarrow T_2$	0.924±0.015	28.00±1.372	0.908±0.017	26.42±1.108
$T_2 \rightarrow T_1\#$	0.930±0.014	27.04±1.533	0.906±0.019	22.77±1.261
$T_2\# \rightarrow T_1$	0.955±0.010	29.07±1.545	0.940±0.012	26.30±1.308
Fold 2				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.947±0.009	29.69±1.000	0.933±0.012	27.55±0.991
$T_1\# \rightarrow T_2$	0.920±0.013	28.24±1.086	0.899±0.016	26.21±1.025
$T_2 \rightarrow T_1\#$	0.930±0.013	27.62±1.217	0.905±0.016	24.05±1.680
$T_2\# \rightarrow T_1$	0.951±0.009	28.89±0.936	0.932±0.012	25.39±1.881
Fold 3				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.951±0.012	30.09±1.008	0.941±0.013	28.57±0.996
$T_1\# \rightarrow T_2$	0.922±0.016	28.27±1.056	0.904±0.018	26.49±1.021
$T_2 \rightarrow T_1\#$	0.934±0.013	27.74±1.167	0.909±0.019	24.26±1.667
$T_2\# \rightarrow T_1$	0.955±0.010	29.53±1.281	0.941±0.014	26.82±1.675
Fold 4				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.946±0.019	29.67±1.271	0.931±0.018	27.60±0.933
$T_1\# \rightarrow T_2$	0.911±0.026	27.63±1.585	0.894±0.023	25.69±1.025
$T_2 \rightarrow T_1\#$	0.923±0.022	27.19±1.696	0.901±0.022	24.69±1.564
$T_2\# \rightarrow T_1$	0.950±0.017	29.07±1.771	0.933±0.020	25.86±2.046

Method comparison in the IXI dataset. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN and Multimodal on 40 additional subjects divided into 4-folds of 25 training, 5 validation and 10 testing subjects. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$) for all four folds. SSIM and PSNR measurements are reported as mean±std across test images.

Table 4.13: Quality of Synthesis in the BRATS dataset: Control analysis

	pGAN		**Multimodal**	
		Fold 1		
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$#	0.939±0.019	27.03±1.779	0.932±0.019	24.61±0.831
T_1# $\rightarrow T_2$	0.938±0.016	26.25±1.943	0.938±0.020	24.35±2.393
		Fold 2		
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$	0.943±0.011	27.31±1.590	0.937±0.007	25.46±0.607
$T_1 \rightarrow T_2$	0.941±0.011	26.63±2.537	0.946±0.010	26.00±2.466
		Fold 3		
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$	0.943±0.013	27.08±1.339	0.937±0.009	24.90±0.785
$T_1 \rightarrow T_2$	0.943±0.010	26.60±2.200	0.944±0.008	25.02±3.310
		Fold 4		
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$	0.941±0.013	26.95±1.593	0.937±0.013	25.54±1.039
$T_1 \rightarrow T_2$	0.942±0.012	26.62±1.589	0.943±0.007	25.52±1.578

Control analysis in the BRATS dataset. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN and Multimodal on 40 subjects from the BRATS dataset. A 4-fold cross-validation procedure was used by randomly sampling nonoverlapping training, validation and testing sets in each fold such that 25 subjects were used for training, 5 for validation and 10 for testing. Two separate models were tested for synthesis of T_2-weighted contrast, and for synthesis of T_1-weighted contrast for all four folds. SSIM and PSNR measurements are reported as mean±std across test images.

Table 4.14: Quality of Synthesis in the Midas dataset: Model generalization

	pGAN		**cGAN$_{reg}$**		**Replica**		**Multimodal**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2$#	0.850±0.020	24.15±0.581	0.849±0.018	24.09±0.405	0.831±0.022	23.67±0.564	0.869±0.016	24.34±0.461
T_1# $\rightarrow T_2$	0.828±0.025	24.15±0.450	0.759±0.028	21.91±0.681	0.797±0.036	21.78±0.412	0.831±0.022	22.99±0.205
$T_2 \rightarrow T_1$#	0.826±0.023	21.03±0.655	0.757±0.021	20.62±0.866	0.706±0.024	17.81±0.447	0.807±0.019	16.98±0.815
T_2# $\rightarrow T_1$	0.803±0.025	21.23±0.491	0.801±0.030	22.45±0.754	0.682±0.033	18.27±0.488	0.786±0.020	15.59±0.897

Synthesis performance for T_1- and T_2-weighted images from the MIDAS dataset was evaluated for pGAN, cGAN, Replica, and Multimodal trained on the IXI dataset. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2$#, T_1# $\rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1$#, T_2# $\rightarrow T_1$). SSIM and PSNR measurements are reported as mean±std across test subjects.

Table 4.15: Quality of Synthesis in the IXI dataset: Reliability against noise

Noise-added source images				
	pGAN		**Multimodal**	
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.926±0.024	27.59±1.809	0.911±0.025	25.96±1.152
$T_1\# \rightarrow T_2$	0.891±0.012	26.17±0.725	0.862±0.018	24.33±0.935
$T_2 \rightarrow T_1\#$	0.907±0.013	26.24±0.883	0.874±0.014	22.38±1.212
$T_2\# \rightarrow T_1$	0.939±0.016	28.42±1.428	0.921±0.020	25.68±1.359
Noise-added target images				
	SSIM	PSNR	SSIM	PSNR
$T_1 \rightarrow T_2\#$	0.850±0.026	24.60±0.967	0.871±0.037	24.36±1.404
$T_1\# \rightarrow T_2$	0.779±0.033	22.97±0.790	0.799±0.027	22.00±0.721
$T_2 \rightarrow T_1\#$	0.756±0.034	21.89±1.198	0.772±0.039	18.02±1.024
$T_2\# \rightarrow T_1$	0.848±0.046	24.26±1.668	0.856±0.053	22.65±1.847

Method comparison in the noisy IXI dataset. Synthesis performance for T_1- and T_2-weighted images was evaluated for pGAN and Multimodal. Two separate models were tested for synthesis of T_2-weighted contrast ($T_1 \rightarrow T_2\#$, $T_1\# \rightarrow T_2$), and for synthesis of T_1-weighted contrast ($T_2 \rightarrow T_1\#$, $T_2\# \rightarrow T_1$). SSIM and PSNR measurements are reported as mean±std across test images.

Chapter 5

Prior-Guided Image Reconstruction for Accelerated Multi-Contrast MRI via Generative Adversarial Networks

The contents of this chapter reflect the work reported in the following publications:

- S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, and T. Çukur, "Synergistic reconstruction and synthesis via generative adversarial networks for accelerated multi-contrast mri," *arXiv preprint*, 2018
- S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, B. Tınaz, and T. Çukur, "Joint recovery of variably accelerated multi-contrast MRI acquisitions via generative adversarial networks," in *27th Annual Meeting of ISMRM*, p. 0666, 2019
- S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, B. Tınaz, and T. Çukur, "Prior-guided image reconstruction for accelerated multi-contrast mri via

generative adversarial networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 6, pp. 1072–1087, 2020

5.1 Introduction

Magnetic resonance imaging (MRI) is a preferred modality for assessment of soft tissues due to diversity of contrasts that it can provide. A typical MRI protocol comprises a set of pulse sequences that capture images of the same anatomy under different contrasts, with the aim to enhance diagnostic information. For instance, in neuroimaging protocols, T_1-weighted images are useful for delineation of gray and white matter, whereas T_2-weighted images are more useful for delineation of fluids and fat. Although acquisition of multiple distinct contrasts is desirable, it may not be feasible due to scan time limitations or uncooperative patients. Thus, methods for accelerating MRI acquisitions without compromising image quality are of great interest for multiple-contrast applications.

The predominant approach for accelerated MRI relies on undersampled k-space acquisitions for scan time reduction, and on reconstruction algorithms for recovery of missing samples based on the collected evidence (i.e., acquired samples) [186, 115, 1, 2, 75]. Given the compressible nature of MR images, the state-of-the-art approach is sparse recovery [1, 2], which employs variable-density random undersampling in k-space to capture most of the energy in the MR images while ensuring low coherence of aliasing artifacts. The inverse problem of image reconstruction from sub-Nyquist sampled data is then solved via regularization from known transform domains [1, 2], learned transform domains [49] or end-to-end deep neural networks [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. Despite the promise of deep models for image reconstruction, the evidence collected on the target MR image diminishes towards high acceleration factors due to undersampling. In turn, this significantly degrades the reconstruction performance, and causes loss in particularly high-spatial-resolution image features that may be relevant for diagnosis.

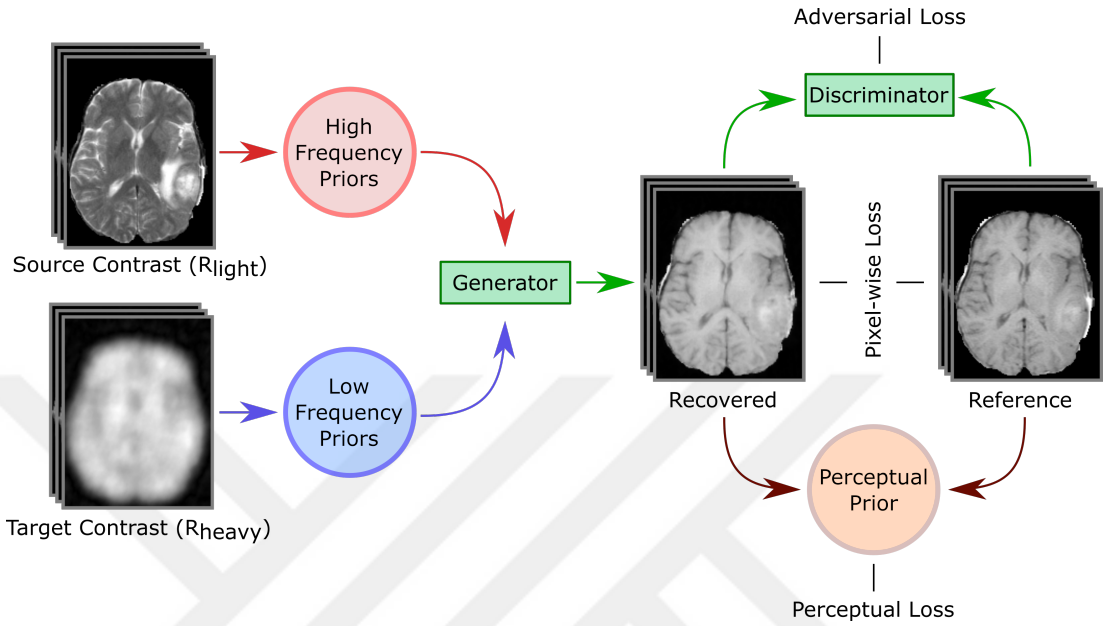

Figure 5.1: The proposed rsGAN method synergistically recovers undersampled multi-contrast MRI acquisitions by complementarily using three priors: shared high-frequency priors available in fully-sampled or lightly undersampled acquisitions of one or more source contrasts to preserve high-spatial-frequency details, low-frequency priors available in highly undersampled acquisitions of one or more target contrasts to prevent feature leakage/loss, and a perceptual prior to improve recovery of high-level features. The input-to-output mapping is implemented using a conditional adversarial network with a generator and a discriminator. The generator learns to recover realistic high-quality target-contrast images by minimizing a pixel-wise, a perceptual and an adversarial loss function. The discriminator learns to discriminate between synthetic and real pairs of multi-contrast images by maximizing the adversarial loss function.

A fundamentally different approach for accelerated MRI is to perform fully-sampled acquisitions of a subset of the desired contrasts (i.e., source contrasts), and then to synthesize missing contrasts (i.e., target contrasts). This approach requires an intensity-based mapping model estimated using a collection of image pairs in both source and target contrast [22, 126, 127, 130, 128, 23, 129, 131, 123, 21, 125, 24, 25, 26, 124, 132, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 187, 19]. The model can be based on sparse linear mapping between source and target patches [21], or deep neural networks for enhanced accuracy [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38]. Although deep models for synthesis are promising, local inaccuracies may occur in synthesized images when the source

contrast is less sensitive to differences in relaxation parameters of two tissues compared to the target contrast, or vice versa. For instance, inflammation can be more clearly delineated from normal tissues in T_2-weighted as opposed to T_1-weighted images. In such cases, synthesized images might contain artificial pathology or fail to depict existing pathology.

Here, we propose a new approach for synergistic recovery of undersampled multi-contrast MRI acquisitions by complementarily using reconstruction and synthesis models. The reconstruction branch takes as a prior the low-spatial-frequency information available in the collected evidence for the target contrast, whereas the synthesis branch takes as a prior the high-spatial-frequency information available in the fully sampled or lightly undersampled source contrast. These low-level spatial-frequency priors are complemented with a perceptual prior that improves recovery of higher-level image features [134]. The input-to-output mapping is implemented using conditional generative adversarial networks (GAN), which were recently shown to outperform traditional deep network models for image reconstruction [3, 4, 5, 10] and synthesis tasks [27, 28]. The proposed reconstructing-synthesizing GAN (rsGAN) contains a generator for estimating the target-contrast image given heavily undersampled target-contrast evidence and either fully sampled or lightly undersampled source-contrast image; and a discriminator to ensure that recovered images are as realistic as possible [110]. Low-spatial-frequencies are densely sampled in both target and source acquisitions, but reconstructions of the target contrast will inherently focus on low-frequency information in target acquisitions. Because the heavily undersampled target contrast misses high-frequency samples at large, the source contrast serves as the primary basis of high-frequency information. The proposed rsGAN model learns to fuse this multitude of input information in a data-driven manner.

Deep neural networks were previously proposed for recovery of multi-contrast MR acquisitions where each acquisition was accelerated at an identical rate [188, 189, 190]. Despite improved recovery compared to isolated reconstruction of individual contrasts, joint reconstruction may still suffer from loss of high-spatial-frequency information towards higher acceleration factors. Deep neural networks were also proposed for enhanced recovery of target-contrast images by

incorporating structural information from fully sampled images of a separate contrast [191, 192, 193]. Compared to [191, 192, 193] that employ loss terms based on mean square/absolute errors or structural similarity, rsGAN leverages an adversarial loss that demonstrated improved capture of high-spatial-frequency information [27]. A recent, independent study proposed a GAN model for super-resolution in a target contrast acquisition via the aid of fully-sampled images of a source contrast [194]. There are several technical differences between rsGAN and the model in [194]. In [194] sources have to be fully sampled, whereas rsGAN also enables light undersampling of source contrasts. For improved recovery, rsGAN further includes a perceptual prior. Lastly, the proposed rsGAN architecture can handle multi-coil complex MRI datasets, and enable reliable recovery at acceleration factors up to 50.

We demonstrated the proposed approach on several datasets: two public datasets containing normal subjects, a public dataset containing patients with high- or low-grade glioma, and a multi-coil dataset containing normal subjects. To comparatively evaluate the proposed method, following competing methods were considered: a reconstructing network (rGAN) that recovers the target-contrast image given undersampled images of the targets contrasts accelerated at identical rates, a reconstructing network (jGAN) that recovers the target-contrast image given undersampled images of the both source and target contrasts accelerated at identical rates, a synthesizing network (sGAN) that synthesizes the target-contrast image given fully sampled images of the source contrast, a joint super resolution reconstructing network (sr-sGAN) [194] that recovers the target-contrast image given undersampled images of the target contrasts and fully sampled images of the source contrast, and a variant of rsGAN deprived of the perceptual prior (rsGAN^{-}). Our results indicate that rsGAN yields enhanced performance compared to the competing methods. In particular, rsGAN enables higher acceleration factors compared to rGAN and jGAN since it more reliably recovers high-spatial-frequency information. Compared to sGAN, rsGAN achieves improved reliability against artificial feature loss or leakage since it uses collected evidence from the target contrast to prevent hallucination. Compared to sr-sGAN, rsGAN achieves enhanced recovery at low to intermediate acceleration

factors (up to 20x). Compared to rsGAN$^-$, rsGAN improves reliability of high-level features. Overall, the proposed approach can successfully recover MR images of at acceleration factors up to 50x in the target contrasts, enabling a significant improvement in multi-contrast MRI.

Contributions

1. To our knowledge, this is the first GAN-based architecture that simultaneously leverages low-spatial-frequency, high-spatial-frequency and perceptual priors to accelerate multi-contrast MRI acquisitions.
2. The proposed approach can enable high acceleration factors up to 50x by incorporating information from both source and target contrasts.
3. The proposed approach can successfully recover pathologies that are either missing in the source contrast or are not clearly visible in the undersampled acquisitions of the target contrast.
4. The proposed approach can jointly reconstruct and synthesize the target contrast even when the source contrasts are moderately undersampled.

5.2 Theory and Methods

5.2.1 Accelerated MRI

Two mainstream approaches that can be used to accelerate MR acquisitions and enhance the diversity of acquired contrasts are reconstruction of a target contrast given randomly undersampled acquisitions of the same contrast, and synthesis of a target contrast based on fully-sampled acquisitions of a distinct source contrast. Both approaches incorporate prior information about image structure to improve the conditioning of the inverse problem of recovering images of the target contrast.

However, they differ fundamentally in the type of prior information used. The problem formulations for reconstruction and synthesis are overviewed below.

5.2.1.1 Reconstruction

In this case, MR acquisitions are accelerated commonly via variable-density random undersampling patterns:

$$F_u m_1 = y_{1a} \tag{5.1}$$

where F_u is the partial Fourier operator defined at the k-space sampling locations, and m_1 is the image of the target contrast, y_{1a} are the acquired k-space data. The reconstruction task is then to recover the target image given the collected evidence (i.e., acquired data). Note that the problem in Eq. 5.1 is ill-posed, thus successful recovery requires additional prior information about the image. In the CS framework, this prior information reflects the sparsity of the image in a known transform domain (i.e., wavelet, TV transforms). The prior can be incorporated into the inverse problem as a regularization term:

$$\widehat{m_1} = \arg\min_{m_1} \quad \lambda||F_u m_1 - y_{1a}||_2 + R(m_1) \tag{5.2}$$

where the first term enforces consistency of the reconstructed and acquired data in k-space, $R(m_1)$ is the regularization term reflecting the prior, and λ controls the relative weighting of data consistency against the prior. $R(m_1)$ typically involves the ℓ_0 or ℓ_1-norm of transform coefficients.

Recent studies have proposed neural-network methods to adaptively learn both nonlinear transform domains directly from MRI data and how to recover images from these domains. In the training stage, a large dataset of pairs of undersampled and fully-sampled acquisitions are leveraged to learn the network-based solution to the inverse problem:

$$\mathcal{L}_{rec}(\theta) = E_{m_{1t}^u, m_{1t}}||G(m_{1t}^u; \theta) - m_{1t}||_p \tag{5.3}$$

where m_{1t}^u and m_{1t} represent undersampled and fully-sampled training images, $G(m_{1t}^u; \theta)$ is the reconstructed output of the neural network based on network

parameters θ, and $||.||_p$ denotes ℓ_p-norm (where p is typically 1 or 2). Once the network parameters that minimize the objective in Eq. 5.3 have been learned, the following optimization problem can be cast to obtain reconstructions of undersampled acquisitions:

$$\widehat{m_1} = \arg\min_{m_1} \quad \lambda||F_u m_1 - y_{1a}||_2 + ||G(m_1^u; \theta^*) - m_1||_2 \tag{5.4}$$

where m_1^u is the undersampled image, $G(m_1^u; \theta^*)$ is the reconstruction by the trained network with parameters θ^*, and $\widehat{m_1}$ is final recovered image. In Eq. 5.4, the first term again enforces consistency of reconstructed and acquired data. The second term is analogous to $R(m_1)$ in Eq. 5.2, and it enforces consistency of the recovered image to the network reconstruction.

5.2.1.2 Synthesis

In the synthesis case, fully-sampled images of the source contrast are assumed to be available. The task is then to recover target-contrast images (m_1) given source-contrast images (m_2) of the same anatomy. A learning-based procedure is used to estimate a mapping between the source and target contrast images. In the training stage, a large dataset of pairs of fully-sampled images from the source and target contrasts are used (m_{2t}, m_{1t}). In the CS-based synthesis framework, patch-based dictionaries (Φ_2, Φ_1) are formed for both source and target contrasts using m_{2t} and m_{1t}. These dictionaries are analogous to the sparsifying transform domains used in CS reconstructions. The aim is to express each patch in the source contrast images m_2 as a sparse linear combination of transform coefficients of the corresponding dictionary atoms:

$$\alpha(j) = \arg\min_{\alpha(j)} \quad ||m_2(j) - \Phi_2.\alpha(j)||_2 + ||\alpha(j)||_1 \tag{5.5}$$

where $\alpha(j)$ is the learned combination coefficients for the jth patch, $m_2(j)$ denotes the jth patch in the source contrast, and Φ_2 denotes the dictionary formed using patches from m_{2t}. The first term ensures consistency of the synthesized patch to the true patch. The second term enforces sparsity of the vector of combination coefficients. Once the combination is learned, it can be used to synthesize target

contrast images:

$$\widehat{m_1}(j) = \Phi_1.\alpha(j) \tag{5.6}$$

where Φ_1 denotes the dictionary formed using patches from m_{1t}, and $\widehat{m_1}(j)$ is the jth patch of the final synthesized image.

Recent studies have proposed neural-network methods to directly learn an adaptive, non-linear mapping from the source contrast to the target contrast. In the training stage, network parameters are optimized based on a loss function that reflects the error between the network output and the true target image:

$$\mathcal{L}_{synth}(\theta) = E_{m_{1t},m_{2t}}||G(m_{2t};\theta) - m_{1t}||_p \tag{5.7}$$

where m_{1t} and m_{2t} represent pairs of source and target images, and $G(m_{2t};\theta)$ is the mapping from source to target contrast characterized by parameters θ. Once the network parameters that minimize the objective in Eq. 5.7 are learned, the network output can be directly calculated to obtain the synthesis results:

$$\widehat{m_1} = G(m_2;\theta^*) \tag{5.8}$$

where $\widehat{m_1}$ is the prediction using the mapping $G(m_2;\theta^*)$ with parameters θ^*. Unlike the reconstruction task, here there is no evidence that has been collected about the target contrast. Therefore, no optimization procedures are needed for synthesis in the testing stage.

5.2.2 Joint Reconstruction-Synthesis via Conditional GANs

In the reconstruction task, the inverse problem solution uses undersampled acquisitions of the target contrast as evidence, and intrinsic image properties such as sparsity as prior. As the acceleration factor grows, evidence becomes scarce particularly towards high spatial frequencies that are sparsely covered by variable-density patterns. This in turn elevates the degree of aliasing artifacts; and if heavier weighting is given to the prior as a remedy, important features may be

lost in the recovered images. Meanwhile, in the synthesis task, the inverse problem solution uses fully-sampled acquisitions of a distinct source contrast of the same anatomy as a prior. When the source and target contrasts exhibit similar levels of sensitivity to differences in tissue parameters, this prior can enable successful solution of the inverse problem. However, when the source and target show differential sensitivity, then features that are not supposed to be in the target may leak from the source onto the synthesized image, or features that must be present in the target may be missed.

To address the limitations of pure reconstruction or synthesis, we proposed to synergistically combine the two approaches with the aim to enhance recovery of multi-contrast MRI images. As such, the proposed approach consists of two branches: (1) A reconstruction branch that aggregates information from the target contrasts in the form of magnitude and phase images. (2) A synthesis branch that aggregates information from the source contrasts in the form of magnitude images.

Given k target contrasts and $n-k$ source contrasts, the joint recovery problem can be formulated as:

$$\begin{aligned} \widehat{m}_{1,2,3,\ldots,n} = \underset{m_{1,2,\ldots,n}}{\arg\min} \quad & \lambda \sum_{i=1}^{k} ||F_u m_i^{hu} - y_{ia}||_2 + \\ & \lambda \sum_{j=k+1}^{n} ||F_u m_j^{lu} - y_{ja}||_2 + \\ & R(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu}) \end{aligned} \tag{5.9}$$

where $R(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu})$ is a regularization term based on prior information, m_i^{hu} is the ith contrast that is heavily undersampled (i.e., target contrast), and m_j^{lu} is jth contrast that is lightly undersampled (i.e., source contrast), and $y_i a$ denotes the acquired data for the ith contrast. We recast Eq. 5.9 using a neural-network based formulation:

$$\widehat{m}_{1,2,3,\ldots,n} = \underset{m_{1,2,\ldots,n}}{\arg\min} \quad \lambda \sum_{i=1}^{k} ||F_u m_i^{hu} - y_{ia}||_2 + \lambda \sum_{j=k+1}^{n} ||F_u m_j^{lu} - y_{ja}||_2 + \sum_{l=1}^{n} ||G(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu}; \theta^*)[l] - m_l||_2 \tag{5.10}$$

Here, multiple separate channels for network output are considered since multiple contrast images can be recovered simultaneously. In Eq. 5.10, $G(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu}; \theta^*)[l]$ denotes the lth channel of the network output, among a total of n channels for the entire set of contrasts. The first two terms respectively enforce the consistency of reconstructed data to acquired data in the target and source contrasts. The last term enforces consistency of the network outputs to the recovered images. Solution of Eq. 5.10 yields estimates of the images for each contrast separately as:

$$y_{ir}(k) = \begin{cases} \dfrac{F\{G(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu}; \theta^*)[i]\}(k) + \lambda y_{ia}(k)}{1+\lambda} & \text{if } k \in \Omega \\ F\{G(m_1^{hu}, \ldots, m_k^{hu}, m_{k+1}^{lu}, \ldots, m_n^{lu}; \theta^*)[i]\}(k) & \text{otherwise} \end{cases}$$

$$\widehat{m}_i = F^{-1}\{y_{ir}\} \tag{5.11}$$

where y_{ir} denotes the k-space representation of the image for the ith contrast, Ω is the set of acquired k-space samples, F is the Fourier transform operator, and F^{-1} is the inverse Fourier transform operator. The solution stated above performs two subsequent projections on the input images. The first projection takes undersampled acquisitions to generate the network predictions. The second

projection enforces data consistency between data samples that were originally acquired and those that are predicted by the network.

Based on the recent progress by generative adversarial networks in MR image synthesis and reconstruction tasks, we chose to build the joint recovery network using a conditional GAN architecture. Our network contains two subnetworks: a generator and a discriminator. The task of the generator is to learn a mapping from undersampled acquisitions onto fully-sampled acquisitions of source and target images. Both synthesis and reconstruction branches are provided to the generator part of the network as separate input channels. During the training, the network learns to adaptively fuse this information in a data-driven way. Meanwhile, the task of the discriminator is to differentiate between the images predicted by the generator and the actual images. As such, an adversarial loss function is typically used to train both subnetworks. Here for stabilized training, we used adversarial loss as in LSGAN [156]:

$$\begin{aligned}\mathcal{L}_{condAdv}(\theta_D, \theta_G) = &-E_{\mathbf{m_t}}[(D(\mathbf{m_t}; \theta_D) - 1)^2] \\ &-E_{\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}}[D(G(\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}; \theta_G); \theta_D)^2]\end{aligned} \tag{5.12}$$

where $\mathbf{m_t}$ represents the MR images aggregated across n contrasts $(m_1, m_2, \ldots, m_n)$ in the training dataset, $\mathbf{m_t^{hu}}$ represents the heavily undersampled acquisitions aggregated across k target contrasts $(m_1, m_2, \ldots, m_k)$, $\mathbf{m_t^{lu}}$ represents the lightly undersampled acquisitions aggregated across $n-k$ source contrasts $(m_{k+1}, m_{k+2}, \ldots, m_n)$, G is the generator with parameters θ_G, D is the discriminator with parameters θ_G, and $\mathcal{L}_{condAdv}(\theta_D, \theta_G)$ is the adversarial loss function. G was trained to minimize $E_{\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}}[(D(G(\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}; \theta_G)) - 1)^2]$ instead of $-E_{\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}}[D(G(\mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}; \theta_G))^2]$. To ensure reliable recovery in each channel, a pixel-wise loss function was incorporated to the generator:

$$\mathcal{L}_{L1}(\theta_G) = E_{\mathbf{m_t}, \mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}}\Big[||G(\mathbf{m_t}, \mathbf{m_t^{hu}}, \mathbf{m_t^{lu}}; \theta_G) - \mathbf{m_t}||_1\Big] \tag{5.13}$$

Recent studies on MRI reconstruction and synthesis suggest that incorporating an additional prior in the form of a perceptual loss can further enhance the image quality [28, 27]. The perceptual loss relies on high-level features extracted via networks pretrained on natural images for more general tasks. Following [134], we

extracted feature maps right before the second max-pooling layer of the VGG16 model trained on the ImageNet dataset [88] for object classification. The loss function can be expressed as:

$$\mathcal{L}_{perc}(\theta_G) = E_{\mathbf{m_t},\mathbf{m_t^{hu}},\mathbf{m_t^{lu}}}\Big[||V(G(\mathbf{m_t},\mathbf{m_t^{hu}},\mathbf{m_t^{lu}};\theta_G)) - V(\mathbf{m_t})||_1\Big] \tag{5.14}$$

where $V(.)$ represents the features extracted via VGG16. The adversarial, pixel-wise and perceptual losses are finally combined to train the proposed reconstructing-synthesizing GAN (rsGAN) model:

$$\begin{aligned}\mathcal{L}_{rsGAN}(\theta_D,\theta_G) = \lambda_p\mathcal{L}_{L1}(\theta_G)+\\ \lambda_{perc}\mathcal{L}_{perc}(\theta_G)+\\ \mathcal{L}_{condAdv}(\theta_D,\theta_G)\end{aligned} \tag{5.15}$$

where λ_p and λ_{perc} are the relative weightings of the pixel-wise and perceptual loss functions.

5.2.3 Competing Methods

To evaluate the effectiveness of rsGAN, we compared it against other GAN architectures. A GAN trained to only perform synthesis of the target-contrast images based on the respective source-contrast images. Source-contrast images were taken to be fully-sampled, high-quality images. We will refer to this network as the synthesizing GAN (sGAN). A GAN trained to only perform reconstruction of the target-contrast images based on undersampled acquisitions of all target contrasts accelerated at identical rates. We will refer to this network as the reconstructing GAN (rGAN). A GAN trained to only perform reconstruction of the target-contrast images based on undersampled acquisitions of all source and target contrasts accelerated at identical rates. We will refer to this network as the joint reconstructing GAN (jGAN). In the public datasets, rsGAN was also compared against a variant of rsGAN deprived of the perceptual prior, referred to as rsGAN$^-$, and a GAN trained to only perform recovery of the target-contrast

images based on fully sampled acquisitions of source contrasts and low-resolution acquisitions of target contrasts, referred to as the super-resolution synthesis GAN (sr-sGAN). Note that sGAN, jGAN, rGAN and sr-sGAN were also trained using the perceptual prior.

5.2.4 MRI Datasets

We demonstrated the proposed approach on three different public datasets and a multi-coil dataset containing multi-contrast MRI images. The public datasets MIDAS [87] and IXI (`http://brain-development.org/ixi-dataset/`) comprised images collected in healthy normals. BRATS (`https://sites.google.com/site/braintumorsegmentation/home/brats2015`) comprised images collected in patients with low-grade glioma (LGG) or high-grade glioma (HGG). Relevant details about each dataset are given below.

5.2.4.1 MIDAS Dataset

T_1-weighted and T_2-weighted images in the MIDAS dataset were considered. Data from 40 subjects were analyzed. The scan protocols were as follows:

T_1-weighted images: 3D Gradient-Echo sequence, repetition time (TR)=14ms, echo time (TE)=7.7ms, flip angle=25°, volume size=256×176×256, voxel dimensions=1mm×1mm×1mm.

T_2-weighted images: 2D Spin-Echo sequence, repetition time (TR)=7730ms, echo time (TE)=80ms, flip angle=180°, volume size=256×192×256, voxel dimensions=1mm×1mm×1mm.

5.2.4.2 IXI Dataset

T_1-weighted, T_2-weighted and PD-weighted images in the IXI dataset were considered. Data from 40 subjects were analyzed. The scan protocols were as follows:

T_1-weighted images: repetition time (TR)=9.813ms, echo time (TE)=4.603ms, flip angle=8°, volume size=256×256×150, voxel dimensions=0.94mm×0.94mm×1.2mm.

T_2-weighted images: repetition time (TR)=8178ms, echo time (TE)=100ms, flip angle=90°, volume size=256×256×130, voxel dimensions= 0.94mm×0.94mm×1.2mm.

PD-weighted images: repetition time (TR)=8178ms, echo time (TE)=8ms, flip angle=90°, volume size=256×256×130, voxel dimensions=0.94mm×0.94mm×1.2mm.

5.2.4.3 BRATS Dataset

T_1-weighted, T_2-weighted and FLAIR images in the BRATS dataset were considered. Data from 40 Glioma patients were analyzed. Since the data were acquired in various different sites, no single scan protocol existed. In BRATS, all contrasts were already pre-registered and skull-stripped as publicly shared.

5.2.4.4 Multi-Coil MR Images

T_1-weighted, T_2-weighted and PD-weighted brain images from 10 subjects were acquired at Bilkent University. Images were acquired on a 3T Siemens Tim Trio scanner (maximum gradient strength of 45mT/m and slew rate of 200 T/m/s) using a 32-channel receive only coil. The scan protocols were as follows:

T_1-weighted images: 3D MP-RAGE sequence, repetition time (TR)=2000ms, echo time (TE)=5.53ms, flip angle=20°, volume size=256×192×88, voxel dimensions=1mm×1mm×2mm, acquisition time (TA)= 6:26.

T_2-weighted images: 3D Spin-Echo sequence, repetition time (TR)=1000ms,

echo time (TE)=118ms, flip angle=90°, volume size=256×192×88, voxel dimensions=1mm×1mm×2mm, acquisition time (TA)= 17:39.

PD-weighted images: 3D Spin-Echo sequence, repetition time (TR)=750ms, echo time (TE)=12ms, flip angle=90°, volume size=256×192×88, voxel dimensions=1mm×1mm×2mm, acquisition time (TA)= 13:14.

In the public datasets, (25, 5, 10) subjects were used for (training, validation, testing). In the IXI dataset, one test subject was discarded due to poor registration quality. Within each subject, around 100 central cross-sections that contained brain tissues and that were relatively free of artifacts were selected. Each model was trained using a batch size of 1. This corresponds to nearly 2400-2600 iterations per epoch. In the multi-coil dataset, (7, 1, 2) subjects were used for (training, validation, testing). For each subject, around 155 central cross-sections that contained brain tissues and that were relatively free of artifacts were selected. Models were trained using a batch size of 1. This corresponds to nearly 1085 iterations per epoch.

5.2.5 Image Registration

Since the multi-contrast volumes in the MIDAS, IXI and multi-coil datasets were unregistered, these images were registered before training and testing. For the MIDAS dataset, T_2-weighted images of each subject were registered onto T_1-weighted images of the same subject using a rigid transformation. Images were registered based on mutual information loss. For the IXI dataset, T_2- and PD-weighted images of each subject were registered onto T_1-weighted images of each subject using an affine transformation. In the multi-coil dataset, T_2- and PD-weighted images of each subject were registered onto T_1-weighted images of each subject using a rigid body transformation. Images were registered based on mutual information loss. Registrations were carried out using FSL [161, 160].

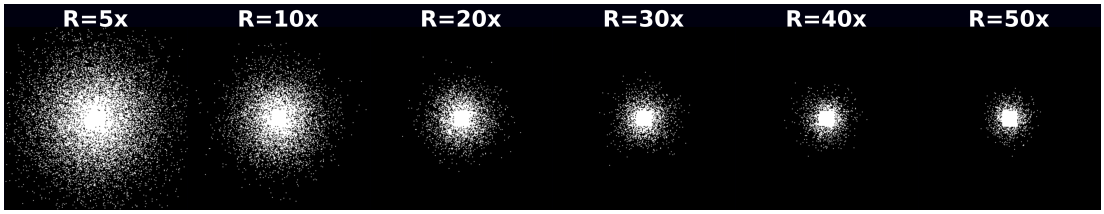

Figure 5.2: Examples of undersampling patterns for acquisitions accelerated in a broad range (R= 5x, 10x, 20x, 30x, 40x, 50x)

5.2.6 Undersampling Patterns

For heavily undersampled acquisitions of the target contrast, we examined acceleration factors in a broad range (R= 5x, 10x, 20x, 30x, 40x, 50x; Fig. 5.2). For lightly undersampled acquisitions of the source contrast, we examined acceleration factors in a relatively limited range (R=1x, 2x, 3x). For rsGAN and rGAN, variable-density undersampling was used [1]. The undersampling patterns were generated using bi-variate normal probability density functions. Covariance of the density functions was separately adjusted for each value of R. Fully-sampled images were Fourier transformed, and then retrospectively sampled using the generated patterns. Distinct random patterns were generated for each subject within each dataset.

For proof of concept demonstration at high acceleration rates, we simulated 2D undersampling in the transversal plane for the public datasets. For multi-coil acquisitions, 2D undersampling was performed in the coronal plane on a 192x88 grid, so a relative narrower range of accelerations were considered (R=5x, 10x, 15x, 20x, 25x, 30x).

5.2.7 Model Training Procedures

All GAN-based models were trained using an identical set of procedures. To train each conditional GAN, we adopted the generator and

discriminator from [134] and [155]. The generator consisted of the following convolutional layers (Conv) connected in series: Conv (kernel-size=7, output-features=64, stride=1, activation=ReLU), Conv (kernel-size=3, output-features=64, stride=2, activation=ReLU), Conv (kernel-size=3, output-features=256, stride=2, activation=ReLU), 9x resnet blocks (kernel-size=3, output-features=256, stride=1, activation=ReLU), fractionally-strided Conv (kernel-size=3, output-features=128, stride=2, activation=ReLU), fractionally-strided Conv (kernel-size=3, output-features=64, stride=2, activation=ReLU), Conv (kernel-size=7, output-features=1, stride=1, activation=none). The discriminator consisted of the following convolutional layers (Conv) connected in series: Conv (kernel-size=4, output-features=64, stride=2, activation=leakyReLU), Conv (kernel-size=4, output-features=128, stride=2, activation=leakyReLU), Conv (kernel-size=4, output-features=256, stride=2, activation=leakyReLU), Conv (kernel-size=4, output-features=512, stride=1, activation=leakyReLU), Conv (kernel-size=4, output-features=1, stride=1, activation=none).

Generator and discriminator networks were trained for 100 epochs using the Adam optimizer [94], with decay rates for the first and second moment estimates set as 0.5 and 0.999. For the generator, the learning rate was set as 0.0002 for the initial 50 epochs and then linearly decayed to 0 during the remaining epochs. For the discriminator, the learning rate was set as 0.0001 for the first 50 epochs and then linearly decayed to 0 during the remaining epochs. Dropout regularization was used to enhance the generalizability of the network model, with a dropout rate of 0.5. Instance normalization was applied [164]. All model weights were randomly initialized based on a normally-distributed variable with 0 mean and 0.02 standard deviation.

The optimal weightings of pixel-wise loss (λ_p) and perceptual loss (λ_{perc}) terms were determined via a cross-validation procedure supplemented by visual inspection. Using the training data, separate models were obtained for λ_p in [10 150] and λ_{perc} in [10 150]. Weight selection was then performed by maximizing PSNR on the validation data. Recovered validation images were also visually inspected.

When needed, selected weights were further fine-tuned to prevent low-quality recovery due to artifacts. Following these procedures, a common λ_p =100 value was chosen for all datasets that yielded near-optimal results consistently across datasets and acceleration factors. Note that our optimum λ_p value closely matches weighting reported for conditional GAN models in the literature [27, 155]. Meanwhile, a separate λ_{perc} value was chosen for each dataset and for each acceleration factor. Relative weighting of data consistency against the prior (λ) was set to infinity.

Note that although the public datasets used in this study contain only coil combined magnitude images, the Fourier reconstructions of undersampled acquisitions are complex valued. Therefore, for each input contrast, two channels were designated to represent the magnitude and phase image components. For each target contrast, separate networks were trained to recover fully-sampled magnitude images. In the multi-coil dataset, first GCC [92] was used to reduce computational complexity by decreasing the number of coils from 32 to 5. For each input contrast, 5 channels were designated to represent magnitude components. In practice, the comparative performance of rsGAN models without and with phase depends on the benefits of added phase information against the disadvantages of fitting a more complex model. In the multi-coil dataset with 5 virtual coils, adding phase information for each individual contrast amounts to 5 extra input channels, considerably expanding model complexity. Since we observed that additional phase channels caused a slight decline in performance, we preferred to use rsGAN models without phase information in the multi-coil analyses of rsGAN. For each target contrast, separate networks were trained to recover fully-sampled coil-combined magnitude images. Reference coil-combined images were obtained by using coil sensitivity maps estimated via ESPIRiT [86]. In case of multiple acquisitions other coil-combined methods can also be deployed [195, 196].

To maximize model performance, a separate model was trained for each unique collection of source and target contrasts, and acceleration factors. For generalizing the rsGAN model to also handle light undersampling of the source contrast,

a separate rGAN model was first trained to recover undersampled source acquisitions at each acceleration factor. In the testing phase, the reconstructed source contrast was then fed to the rsGAN model.

For multi-coil data, rsGAN was first trained to recover a coil-combined magnitude image for the target contrast from undersampled multi-coil magnitude images for source and target contrasts. Second, a coil-combined complex image for the target was obtained by adding onto the recovered magnitude image the phase of the coil-combined undersampled images of the target. Third, the coil-combined complex target image was back-projected onto individual coils using coil sensitivity maps. Data consistency was enforced on the resultant multi-coil complex target data, and a coil-combined complex target image was then obtained. As such, phase information in undersampled acquisitions was leveraged to enable data consistency projections.

5.2.8 Experiments

5.2.8.1 Main Experiments

To evaluate the comparative performance of the proposed approach, rsGAN, rGAN, jGAN, and sGAN were individually trained and tested on multi-contrast MRI datasets. Theoretically, as R approaches 1x, rsGAN, rGAN and jGAN should show nearly identical performance that is superior to sGAN since sGAN has no evidence collected about the target contrast. As R goes to infinity, rsGAN and sGAN should show nearly identical performance that is superior to rGAN, since no evidence from the target contrast will be available to any of the networks. In intermediate R values, we reasoned that rsGAN would outperform rGAN and jGAN in terms of reliability in recovery of high-frequency information since variable-density patterns suboptimally sample high spatial frequencies in the target contrast. We also reasoned that rsGAN would outperform sGAN especially when the source and target contrasts showed differential sensitivity to differences in tissue parameters. Based on these notions, we measured the performance of

all four methods across a broad range of acceleration factors. To evaluate the effects of perceptual prior and variable-density sampling patterns, rsGAN was also compared against rsGAN$^-$ and sr-sGAN in the public datasets. We reasoned that incorporation of the perceptual prior should enhance performance. We also reasoned that as R approaches 1, rsGAN should perform better than sr-sGAN since rsGAN contains more high-spatial frequency information from the target contrast. As R approaches infinity, rsGAN should perform similar to sr-sGAN since the variable-density sampling patterns in rsGAN approach the central sampling patterns in sr-sGAN at these acceleration rates.

In both MIDAS and BRATS datasets, we considered two main scenarios. First, T_1-weighted acquisitions were taken as the source contrast (R=1x), and T_2-weighted acquisitions were taken as the target contrast (R=5x, 10x, 20x, 30x, 40x, 50x). Second, T_2-weighted acquisitions were taken as the source (R=1x), and T_1-weighted acquisitions were taken as the target (R=5x, 10x, 20x, 30x, 40x, 50x).

Two distinct scenarios were examined in both IXI and multi-coil datasets. First, T_1-weighted acquisitions were taken as the source contrast (R=1x), and both T_2- and PD-weighted acquisitions were taken as the target contrasts (R=5x, 10x, 20x, 30x, 40x, 50x in the IXI dataset, and R=5x, 10x, 15x, 20x, 25x, 30x in the multi-coil dataset). Since T_2- and PD-weighted acquisitions are typically performed using similar sequences, the acceleration factors for these two contrasts were always matched. Second, the source T_1-weighted acquisitions were lightly undersampled (R=2x, 3x), and T_1-, T_2-, and PD-weighted images were jointly recovered. The overall scan time for an accelerated multi-contrast protocol depends on the distribution of R across contrasts, and individual scan times for all contrasts. To systematically examine scan efficiency, we measured recovery performance for jGAN and rsGAN with the same overall scan time. Analyses were performed on the in vivo multi-coil datasets for a fixed scan time of 250 sec, where T_1 was the source contrast and T_2 and PD were the target contrasts. For jGAN this corresponds to R=8.9x across all contrasts, whereas for rsGAN this corresponds to R_{T_1} =3x for the source contrast and R=15x for target contrasts.

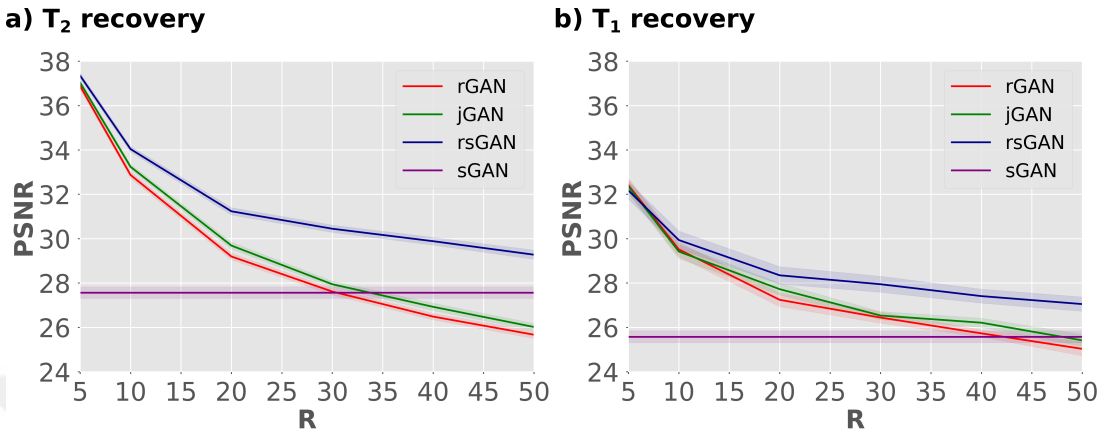

Figure 5.3: The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1- and T_2-weighted images from the MIDAS dataset. The acquisition for the source contrast was fully sampled, and the acquisition for the target contrast was undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. PSNR was measured between recovered and fully-sampled reference target-contrast images. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_1 is the source contrast and T_2 is the target contrast. **(b)** PSNR (mean±standard error) when T_2 is the source contrast and T_1 is the target contrast. The performance of sGAN remains constant across R since it does not use any evidence from the target-contrast acquisitions. As expected, the performance of rGAN, jGAN, and rsGAN gradually decreases for higher values of R where the evidence from the target contrast becomes scarce. However, rsGAN performs well even at very high acceleration factors..

5.2.8.2 Control Experiments

Here, for more efficient model training, we preferred to focus on cross-sections that contained brain tissue. To rule out potential biases in model generalizability due to this selection, we conducted control experiments where rGAN, jGAN and rsGAN were trained on all available cross-sections in the IXI dataset without any selection (referred to as $rGAN_{All}$, $jGAN_{All}$ and $rsGAN_{All}$). These models were compared with rGAN, jGAN and rsGAN trained on the originally selected central cross-sections. Performance comparisons were carried out on independent test sets containing all cross-sections within subjects without any selection procedures.

In the public datasets containing coil-combined magnitude images, phase is only introduced during retrospective undersampling of k-space data, and the

phase values are often small. Thus, in theory, an rsGAN model that receives as inputs only magnitude images should perform similarly to one that receives both magnitude and phase images. To test this prediction, we conducted additional analyses in the BRATS dataset where a variant of rsGAN model with only magnitude channels as inputs was trained, referred to as rsGAN_{m}. rsGAN_{m} was then compared against rsGAN consisting of both magnitude and phase channels as inputs. T_1 was set as the source contrast and T_2 was set as the target contrast.

Here, rsGAN had different model complexity compared to the competing methods (rGAN, jGAN and sGAN). To rule out potential biases due to model complexity, we implemented additional control experiments in the BRATS dataset with rGAN, jGAN and sGAN models with matching complexity to rsGAN. Complexity was balanced across models by maintaining an identical number of input channels to the generator. In these experiments, T_1 was set as the source contrast and T_2 was set as the target contrast. Input to sGAN consisted of magnitude images of fully sampled T_1 contrast concatenated with magnitude and phase images of undersampled T_1 contrast. Input to rGAN consisted of magnitude and phase images of highly undersampled T_2 contrast concatenated with magnitude images of highly undersampled T_2 contrast. Input to jGAN consisted of magnitude and phase images of highly undersampled T_2 contrast concatenated with magnitude images of highly undersampled T_1 contrast. These models are referred to as rGAN_{MC}, jGAN_{MC} and sGAN_{MC}. These models were compared with rsGAN, and regular rGAN, jGAN and sGAN.

In this study, rsGAN was mainly demonstrated for the recovery of T_1-, T_2-, and PD-weighted contrasts. Several diagnostic protocols also include FLAIR acquisitions. To examine the ability of rsGAN to recover FLAIR acquisitions, we also trained models for recovery of FLAIR images in the BRATS dataset. T_1 was used as the source contrast, and FLAIR was used as the target contrast.

To maximize performance for individual contrasts, here the recovery of each target contrast was taken as a separate task. When multi-target-contrast images were considered, a separate rsGAN model was constructed to recover each target contrast. To assess the benefits of this strategy, we conducted additional

Table 5.1: Quality of Recovered Images in the MIDAS Dataset

	rGAN		jGAN		rsGAN(R_{T_1} =1)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	36.88 ±0.13	97.69 ±0.10	37.03 ±0.11	97.76 ±0.10	**37.35**† **±0.12**	**97.96**† **±0.10**
R=10x	32.87 ±0.14	95.18 ±0.20	33.24 ±0.12	95.60 ±0.19	**34.04**† **±0.13**	**96.30**† **±0.19**
R=20x	29.20 ±0.15	90.88 ±0.34	29.69 ±0.15	91.64 ±0.33	**31.23**† **±0.17**	**93.97**† **±0.30**
R=30x	27.61 ±0.15	87.98 ±0.42	27.95 ±0.14	88.82 ±0.40	**30.45**† **±0.16**	**93.25**† **±0.33**
R=40x	26.49 ±0.14	85.75 ±0.47	26.93 ±0.17	86.88 ±0.49	**29.89**† **±0.17**	**92.76**† **±0.35**
R=50x	25.67 ±0.17	83.77 ±0.53	26.02 ±0.16	84.87 ±0.52	**29.28**† **±0.21**	**92.09**† **±0.41**
sGAN						
	PSNR			SSIM		
	27.56±0.27			90.91± 0.48		

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for sGAN, rGAN, jGAN, and rsGAN. T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font, and the significantly better performing values ($p<0.05$) are marked with the '†' symbol.

experiments in the BRATS dataset where two distinct sets of rsGAN models were constructed. The first set consisted of the original rsGAN models (named $rsGAN_1$) trained to recover target contrasts individually (one model recovering T_2 from undersampled T_2 and fully-sampled T_1 acquisitions, and another model recovering FLAIR from undersampled FLAIR and fully-sampled T_1 acquisitions). The second set consisted of a unified rsGAN model (named $rsGAN_2$) trained to jointly recover T_2 and FLAIR from undersampled T_2 and FLAIR, and fully-sampled T_1 acquisitions. The two sets of models were compared in terms of average performance in recovery of T_2 and FLAIR images.

All network models and conventional reconstruction and synthesis techniques were trained and tested on the same instances of data and undersampling patterns. To quantitatively assess the quality of recovered images, the fully-sampled

reference images were used. All images were first normalized to the range [0 1]. Then, peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) were calculated between the recovered and reference images. Statistical significance of differences in PSNR and SSIM between methods were assessed via a nonparametric Wilcoxon signed-rank test. In the public datasets, the statistical significance tests were performed across test subjects. In the multi-coil dataset, due to limited number of test subjects, the statistical significance tests were performed across cross-sections.

5.3 Results

5.3.1 Main Experiments

5.3.1.1 Public Datasets

We first demonstrated the proposed rsGAN method against rGAN, jGAN and sGAN on the MIDAS dataset. We considered two separate models: a model to recover T_2-weighted images given T_1-weighted images as source contrast, and another to recover T_1-weighted images given T_2-weighted images as source contrast. Tables 5.1 and 5.2 list the respective PSNR and SSIM measurements for each model, and Fig. 5.3 illustrates performance as a function of R.

T_2- and T_1-weighted images in the MIDAS dataset recovered while R is varied from 10x to 50x are displayed in Figs. 5.4 and 5.5, respectively. Representative T_2- and T_1-weighted images recovered with ZF, jGAN, sGAN and rsGAN at R=50x are shown in Fig. 5.6. As expected, the similarity between rsGAN and jGAN results increases towards R=10x, and that between rsGAN and sGAN increases towards R=50x. Furthermore, rsGAN recovers images of higher visual quality and acuity than both competing methods, particularly at intermediate R values. These results indicate that the incorporation of a fully-sampled acquisitions of the source contrast enables rsGAN to more reliably recover high-frequency

information compared to rGAN and jGAN, and that the use of evidence collected on the target contrast ensures that rsGAN yields more accurate recovery compared to sGAN. Next, we demonstrated the proposed method on a dataset acquired in patients with high- or low-grade gliomas. We considered two models on the BRATS dataset: a model to recover T_2-weighted images given T_1-weighted images, and another to recover T_1-weighted images given T_2-weighted images. Tables 5.3 and 5.4 list the respective PSNR and SSIM values, and Fig. 5.7 illustrates model performance as a function of R. Representative T_2- and T_1-weighted images in the BRATS dataset recovered with ZF, jGAN, sGAN and rsGAN at R=50x are shown in Fig. 5.8. Note that multi-contrast images can show differential sensitivity to tumor tissue, where tumors can be more easily delineated in T_2- versus T_1-weighted images particularly in patients with low-grade glioma. As a result, sGAN suffers from either loss of features in the target contrast or synthesis of artefactual features. Meanwhile, jGAN suffers from excessive loss of high spatial frequency information at high R. In comparison, rsGAN achieves higher spatial acuity while preventing feature losses and artefactual synthesis. Thus, the rsGAN method enables more reliable and accurate recovery when the source contrast is substantially less or more sensitive to differences in relaxation parameters of two tissues compared to the target contrast.

Next, we demonstrated the utility of rsGAN to recover multiple target contrasts simultaneously. The specific model tested on the IXI dataset was aimed to recover both T_2- and PD-weighted images given T_1-weighted images as source contrast. We examined the effect of light undersampling performed on the source contrast (R_{T_1}=1x, 2x, 3x) in addition to heavy undersampling on the target contrasts (R=5x, 10x, 20x, 30x, 40x, 50x). Tables 5.5 and 5.6 list the PSNR and SSIM measurements for T_2- and PD-weighted images, respectively. Fig. 5.9 illustrates model performance as a function of R_{T_1} and R. Representative T_2- and PD-weighted images in the IXI dataset recovered with ZF, jGAN, sGAN and rsGAN at R_{T_1}=2x, R=30x are shown in Fig. 5.10. The rsGAN method yields sharper images and improved suppression of aliasing artifacts compared to jGAN and sGAN, even when the source contrast acquisitions are accelerated. Across all public datasets, rsGAN achieves 1.66dB higher PSNR and 3.45% higher SSIM

compared to rGAN, 1.40dB higher PSNR and 2.80% higher SSIM compared to jGAN, and 5.18dB higher PSNR and 3.83% higher SSIM compared to sGAN.

We also examined the effects of perceptual prior and variable-density sampling patterns in rsGAN. Tables 5.7-5.12 list the PSNR and SSIM measurements across the recovered images in all public datasets. We find that the original rsGAN model outperforms rsGAN$^-$ on average by 0.53dB PSNR and 0.37% SSIM across the datasets. This result demonstrates the benefit of the perceptual prior for recovery performance. Comparisons among rsGAN and sr-sGAN indicate that rsGAN shows superior performance to sr-sGAN at all acceleration factors up to R=20 where rsGAN achieves 1.01dB higher PSNR and 0.40% higher SSIM, and the two methods perform similarly for R>20 where the differences are 0.16dB PSNR and 0.12% SSIM. Similar performance at very high accelerations is expected since the variable-density sampling patterns in rsGAN approach the central sampling patterns in sr-sGAN at these acceleration rates.

5.3.1.2 Multi-Coil Dataset

We then demonstrated the proposed approach on complex multi-coil dataset. The model was aimed to recover both T_2- and PD-weighted images given T_1-weighted images as source contrast. We examined the effect of light undersampling performed on the source contrast (R_{T_1}=1x, 2x, 3x) in addition to heavy undersampling on the target contrasts (R=5x, 10x, 15x, 20x, 25x, 30x). Tables5.13 and 5.14 list the PSNR and SSIM measurements for T_2- and PD-weighted images, respectively. Fig. 5.11 illustrates model performance as a function of R_{T_1} and R. Overall, rsGAN is the leading performer. rsGAN (R_{T_1}=1) achieves 0.67 dB higher PSNR and 1.81% higher SSIM than rGAN, 0.58 dB higher PSNR and 1.56% higher SSIM than jGAN, and 5.61dB higher PSNR and 5.95% higher SSIM than sGAN. Even at R_{T_1}=3 rsGAN outperforms both rGAN and sGAN in terms of PSNR across values of R>10x. The only exception is at R=5x, where rGAN increases T_2 recovery quality over rsGAN. This result suggests that at very low accelerations, the benefits of added prior information from the source can be outweighed by the added model complexity in rsGAN.

Representative T_2- and PD-weighted images in the multi-coil dataset recovered with ZF, rGAN, sGAN and rsGAN at R_{T_1} =1x, R=10x are shown in Fig. 5.12. The rsGAN method yields sharper images and improved suppression of aliasing artifacts compared to rGAN and sGAN.

Next we measured recovery performance for jGAN and rsGAN with the same overall scan time of 250 sec. For jGAN this corresponds to R=8.9x across all contrasts, whereas for rsGAN this corresponds to R_{T_1} =3x for the source contrast and R=15x for target contrasts. Across all three contrasts, rsGAN significantly outperforms jGAN in SSIM ($p<0.05$) by 0.63% while the two methods have similar PSNR. For the fixed scan time of 250 sec, we also compared the recovery performance of rsGAN explicitly for the source contrast. We observe that rsGAN outperforms jGAN by 4.69 dB in PSNR, and 6.16% in SSIM ($p<0.05$). This indicates that rsGAN is superior in recovery of the source contrasts as expected. These results showcase a scenario where rsGAN with nonuniform acceleration is preferable to jGAN with uniform acceleration.

5.3.2 Control Experiments

To rule out potential biases in model generalizability due to selection of cross-sections, we conducted control experiments in the IXI dataset where we considered two sets of models (see Methods for rGAN, jGAN, rsGAN and $rGAN_{All}$, $jGAN_{All}$, $rsGAN_{All}$). Table 5.15 lists PSNR and SSIM measurements across the recovered images. Overall, $rGAN_{All}$ outperforms rGAN by 0.26dB in PSNR and 0.35% in SSIM, $jGAN_{All}$ outperforms jGAN by 0.30dB in PSNR and 0.36% in SSIM, and $rsGAN_{All}$ outperforms rsGAN by 0.32dB in PSNR and 0.44% in SSIM. Note that the slight performance improvement is natural since the test set contained peripheral cross-sections that were intentionally removed from the training set of rGAN, jGAN and rsGAN, but included in the training set of $rGAN_{All}$, $jGAN_{All}$ and $rsGAN_{All}$. Second, we observe that results of the control experiments are consistent with the original experiments in demonstrating the superiority of rsGAN over alternative models. Overall, $rsGAN_{All}$ achieves 1.80dB higher PSNR

and 4.27% higher SSIM than rGAN_{All}, and 1.63dB higher PSNR and 3.74% higher SSIM than jGAN_{All}. Note that rsGAN also achieves 1.48dB higher PSNR and 3.84% higher SSIM than rGAN_{All}, and 1.30dB higher PSNR and 3.31% higher SSIM than jGAN_{All}.

To examine the effects of input phase channels in recovery of coil-combined magnitude images, two sets of models were considered (rsGAN_{m} and rsGAN) for T_2 recovery in the BRATS dataset. Table 5.16 lists PSNR and SSIM measurements across the recovered images. We find that removing the phase channels from the rsGAN model decreases average PSNR and SSIM by 0.30dB and 0.05%. This difference might be attributed to the nature of phase images that typically emphasize information about tissue boundaries.

Next, we conducted additional experiments to rule out any bias that might have occurred due to differences in model complexities among rGAN, jGAN, sGAN and rsGAN. These experiments were conducted on the BRATS dataset where T_1 was set as the source contrast and T_2 was set as the target contrast. Table 5.17 lists PSNR and SSIM measurements across the recovered images. We find that rsGAN still outperforms rGAN_{MC}, jGAN_{MC} and sGAN_{MC} that were matched to rsGAN in model complexity. Overall rsGAN outperforms rGAN_{MC} by 1.39dB PSNR and 1.50% SSIM, jGAN_{MC} by 0.99dB PSNR and 1.19% SSIM, and sGAN_{MC} by 7.81dB PSNR and 4.99% SSIM. Furthermore, changing network complexity has minor effects in terms of model performance. Overall, performance in rGAN changes by 0.03dB PSNR and 0.02% SSIM, in jGAN changes by 0.19dB PSNR and 0.04% SSIM, and in sGAN changes by 0.56dB PSNR and 0.19% SSIM. Taken together, these experiments indicate that our results are not unduly biased by variability in model complexity.

We also evaluated the ability of rsGAN in recovering FLAIR images. rGAN, jGAN and rsGAN were compared in terms of average performance on the BRATS dataset. Table 5.18 lists PSNR and SSIM measurements across the recovered images. We find that rsGAN outperforms both rGAN and jGAN (please see Fig. 5.13 for representative images). In this task, rsGAN outperforms rGAN by 0.74dB PSNR and 0.85% SSIM, and jGAN by 0.62dB PSNR and 0.65% SSIM.

These results suggest that the proposed rsGAN model has potential to synthesize a broader selection of contrasts.

Lastly, we compared the original rsGAN model that independently recovers all targets in a multi-target-contrast setting (rsGAN_1) against a unified rsGAN model that simultaneously recovers all target contrasts (rsGAN_2). Comparisons were performed for recovery of T_2 and FLAIR images in the BRATS dataset. Table 5.19 lists PSNR and SSIM measurements across the recovered images. We find that rsGAN_1 yields 0.25dB higher PSNR and 0.10% higher SSIM than rsGAN_2. Note that this moderate performance drop in the unified model is expected, as rsGAN_2 has to compromise between recovery losses for the two target contrasts.

5.4 Discussion

A synergistic reconstruction-synthesis approach based on conditional GANs was presented for highly accelerated multi-contrast MRI. In this approach, several source- and target-contrast acquisitions accelerated to various degrees are taken as input, and high-quality images for individual contrasts are then recovered. The proposed rsGAN method yielded superior recovery performance against state-of-the-art reconstruction and synthesis methods in three public MRI datasets and a multi-coil dataset. While rsGAN was demonstrated for multi-contrast MRI here, it may also offer improved performance in recovery of images in accelerated multi-modal datasets.

Several previous studies considered joint reconstructions of multi-contrast acquisitions to better use shared structural information among contrasts. In the CS framework, a typical scenario involves multiple acquisitions with nearly identical acceleration rates [197, 198]. Undersampled data are jointly processed, and a joint-sparsity regularization term improves recovery of shared features across contrasts. Another scenario involves the fully-sampled acquisition of a reference contrast that is then used as a structural prior for other contrasts [199].

Prior-guided reconstructions use regularization terms that enforce consistency of the magnitude and direction of image gradients across distinct contrasts. These previous approaches yield enhanced quality over independent processing of each contrast. However, hand-crafted regularization terms based on transforms such as total variation or wavelet reflect often suboptimal assumptions about structural similarity among separate contrasts. The proposed rsGAN method instead employs a data-driven approach to learn to utilize information from source contrast during recovery of target contrasts.

Few recent studies proposed a learning-based method for joint reconstruction of multi-contrast MRI data [188, 189, 190]. Acquisitions for separate contrasts were accelerated at identical rates. Convolutional neural network architectures were used with a subset of network weights shared across contrasts to better capture structural similarities among contrasts. While these previous methods were shown to outperform conventional CS and parallel imaging reconstructions, these are pure reconstruction approaches that can suffer from scarce sampling of high spatial frequencies at high acceleration rates. In contrast, rsGAN employs detailed structural information in a source contrast to enhance the recovery of high-frequency samples in target contrasts. Since the source acquisitions are fully-sampled or lightly undersampled, rsGAN shows improved reliability against losses in resolution. Furthermore, GANs have been shown to better learn the distribution of target images compared to conventional network architectures.

Several independent studies proposed convolutional neural networks for recovery of a target contrast by making use of structural information from a source contrast [191, 192, 193, 194]. Perhaps, the closest to our study is [194], where recovery from a low-resolution target-contrast image was enhanced by incorporating fully-sampled acquisition of a separate contrast [194]. Our study is different from [194] in the following aspects: 1) We demonstrate the proposed approach for reconstruction of multi-coil complex MR images, whereas [194] consider a model to post-process coil-combined images that were already reconstructed. 2) We demonstrate that the proposed approach can jointly reconstruct and synthesize the target contrast even when the source contrasts are undersampled. 3) We demonstrate that the proposed approach can enable high acceleration factors up

to 50x by incorporating information from both source and target contrasts. 4) We incorporate an additional perceptual prior to improve recovery of high-level image features. In addition to these technical differences, we also demonstrated superiority of rsGAN over [194] at all acceleration factors up to 20 (Supp. Tables I-VI).

The synthesis framework is an alternative for recovery of images of a target contrast, where data are only available in a different source contrast. A powerful approach is to construct dictionaries from multi-resolution image patches, and to learn a mapping between the source and target dictionaries [22, 126, 128, 23, 21]. Segregation of the dictionary extraction and mapping stages might yield suboptimal performance. Network-based approaches offer a remedy to this problem by unifying the two stages [127, 24, 25, 26]. We recently proposed GAN-based synthesis for multi-contrast MRI that yielded enhanced performance compared to conventional methods [27]. Yet, due to lack of evidence on the target contrast, a pure synthesis approach can suffer from artificial sensitivity or insensitivity to image features. The rsGAN method, on the other hand, always collects a moderate to small amount of evidence. This helps avoid artefactual feature leakage from the source to the target contrast or loss of target-contrast features that are not apparent in the source contrast.

An important query about the proposed approach is selection of source and target contrasts. Note that rsGAN performs heavy undersampling for only target contrasts, while moderately undersampling source contrasts to preserve as much information regarding detailed tissue structure as possible. This suggests that, in a given MRI protocol, contrasts with better capture of structural details and relatively shorter scan times should be designated as source contrasts, whereas contrasts with relatively limited capture of structural details and longer scan times should be designated as target contrasts. Clinical protocols typically start with a high-resolution T_1-weighted acquisition, so we considered T_1 as a natural source contrast in the current study. Meanwhile, remaining contrasts including T_2, PD or FLAIR with relatively lower capture of structural details and longer scan times were designated as target contrasts. In cases where multiple candidates exist, a selection might be necessary to minimize the overall scan time.

For example, diagnostic protocols for glioma typically include T_1-weighted, postcontrast T_1-weighted, T_2-weighted, and FLAIR acquisitions. In this case, both T_1 and postcontrast T_1 acquisitions are possible candidates of source contrasts. Yet, given the benefits of postcontrast T_1 over T_1 in terms of diagnostic accuracy and/or tumor segmentation, postcontrast T_1 can be selected.

Here, the experiment for a fixed scan time was based on T_1-weighted acquisitions with a 3D MP-RAGE sequence, and T_2- and PD-weighted acquisitions with a 3D Spin-Echo sequence. Naturally the optimal distribution of acceleration factors across contrasts might vary depending on the specific sequences prescribed for each contrast. When other sequences or different sets of sequence parameters are prescribed, the distribution of acceleration factors across contrasts can be re-tuned empirically to maximize reconstruction performance. To ensure optimal performance, separate networks can be trained for each MR protocol and overall scan time. But since network training is performed off-line prior to MR scans, the recovery time would not be affected.

The superiority of rsGAN over jGAN in recovering source contrasts might motivate its use in several scenarios. First, rsGAN can be used to recover a specific source contrast with higher quality within an MRI protocol. The enhanced recovery can prove useful when task-critical information is largely concentrated in this source contrast. For instance, high-quality T_1-weighted brain images are vital for accurate segmentation of white and gray matter, or high-quality postcontrast T_1-weighted images serve as a gold-standard tool for tumor localization. Another example would be to recover multiple phase-cycle balanced steady state free precision (bssfp) [200, 201, 202, 203, 204, 172, 173, 174, 175] images by setting one of the phase-cycle images as a source contrast. Second, rsGAN can be employed in longitudinal imaging studies [205, 206], where subjects are scanned in multiple sessions over extended periods of time. This can help reduce scan time and increase patient comfort in multi-session MRI exams. In such cases, fully-sampled source contrasts can be acquired during the initial session, and they can then aid recovery in subsequent sessions. Lastly, rsGAN might offer utility in facilitating efficient re-acquisition of problematic images in a multi-contrast exam. When a subset of acquisitions suffers from severe patient motion or artifacts, the

acquisitions can be repeated at significantly higher acceleration factors to minimize undesirable increase in scan time. The resulting multi-contrast acquisitions with non-uniform undersampling across contrasts can then be recovered with rsGAN. When the source contrast is affected by motion, it could be reacquired at a moderate acceleration factor and recovered using rGAN. Alternatively, a separate network or algorithm for suppression of motion artifacts could be trained [207, 208] and used as a pre-processing step to rsGAN.

Several technical developments are viable for improving the current implementation of the proposed method. First, the model can be generalized to simultaneously process multiple neighboring cross-sections in addition to multiple contrasts. Correlated tissue structure across cross sections might enhanced recovery despite the increase in model complexity. Second, when multiple source contrasts are present, a weight sharing method can be used to enforce a shared latent representation among contrasts for improved performance. Lastly, a cycleGAN-based model [155] might be implemented to allow for learning on unpaired multi-contrast MRI datasets that are relatively more available than paired datasets.

5.5 Conclusion

We proposed a synergistic reconstruction-synthesis method for accelerated multi-contrast MRI based on conditional generative adversarial networks. End-to-end trained GANs are used to recover high-quality images of target and source contrasts given undersampled acquisitions. Unlike pure learning-based reconstruction, rsGAN uses high-spatial-frequency prior information in the source contrast to enhance recovery of the target contrast. Unlike pure learning-based synthesis, rsGAN bases recovered images on evidence collected through heavily undersampled acquisitions of the target contrast. The proposed method outperforms state-of-the-art reconstruction and synthesis methods, with enhanced recovery of high-frequency tissue structure, and improved reliability against feature leakage or loss. The rsGAN method holds great promise for highly accelerated multi-contrast MRI in clinical practice.

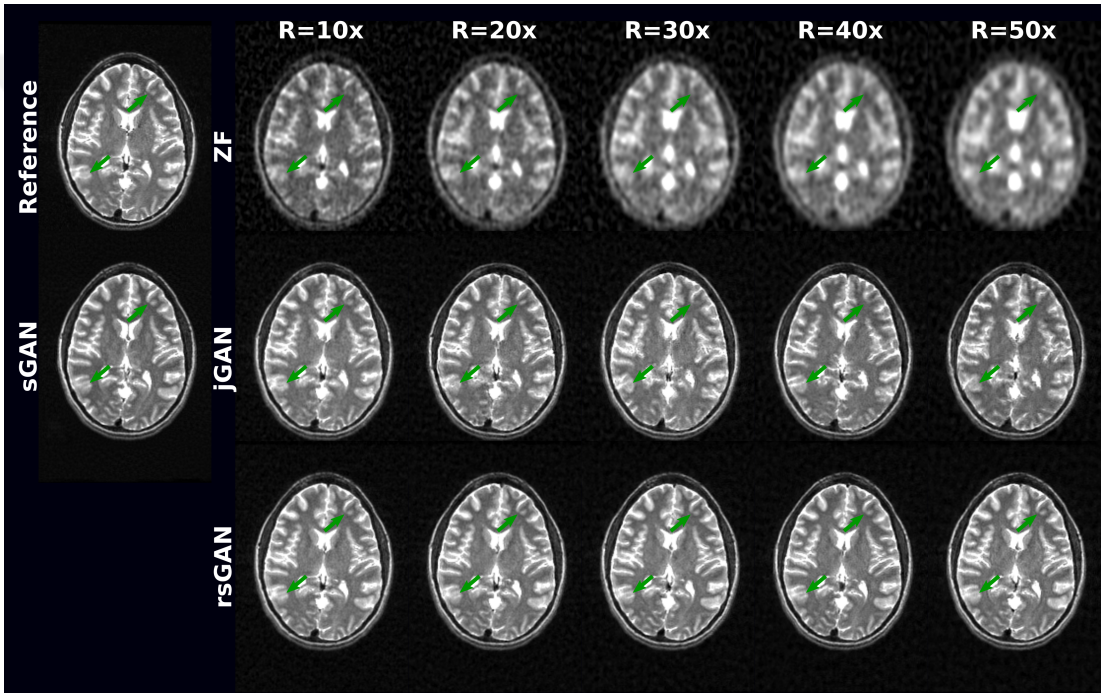

Figure 5.4: T_2-weighted images in the MIDAS dataset were recovered from heavily undersampled acquisitions (R=10x, 20x, 30x, 40x, 50x). The acquisition for the source contrast (T_1-weighted) was fully sampled. Target-contrast images recovered by ZF (zero-filled Fourier reconstruction), sGAN, jGAN, and rsGAN are shown with the fully-sampled reference image. As the value of R increases the performance of jGAN degrades significantly. Meanwhile, rsGAN maintains high-quality recovered images due to use of additional information from the source contrast. Regions with enhanced recovery in rsGAN are marked with arrows.

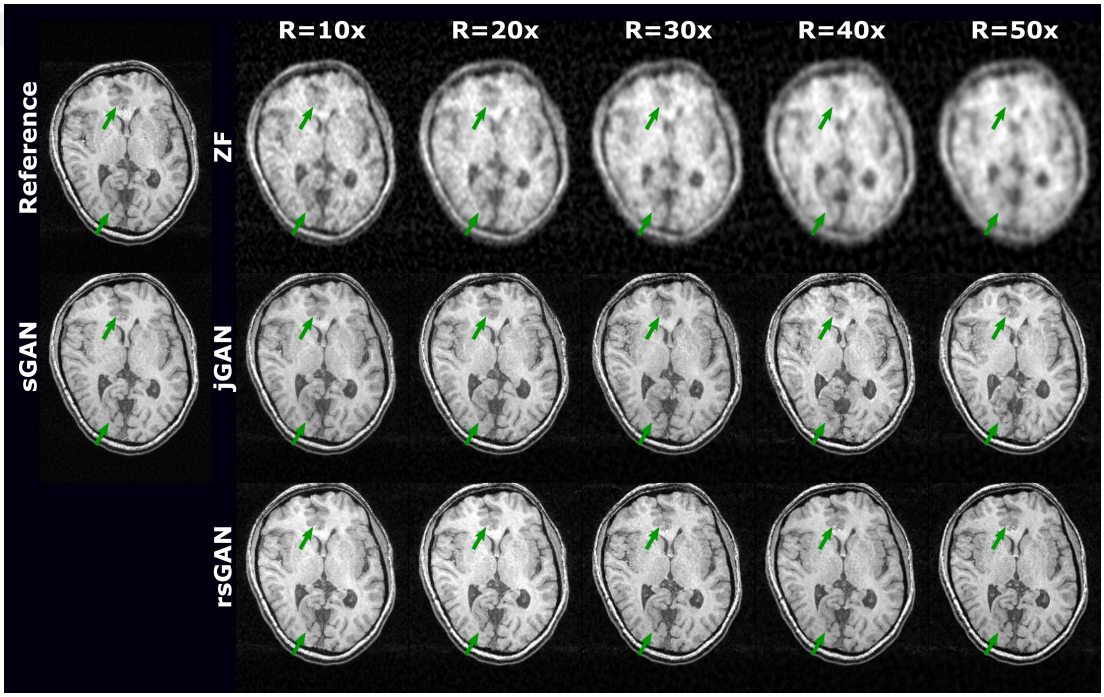

Figure 5.5: T_1-weighted images in the MIDAS dataset were recovered from heavily undersampled acquisitions (R=10x, 20x, 30x, 40x, 50x). The acquisition for the source contrast (T_2-weighted) was fully sampled. Target-contrast images recovered by ZF, sGAN, jGAN, and rsGAN are shown with the fully-sampled reference image. Regions with enhanced recovery in rsGAN are marked with arrows.

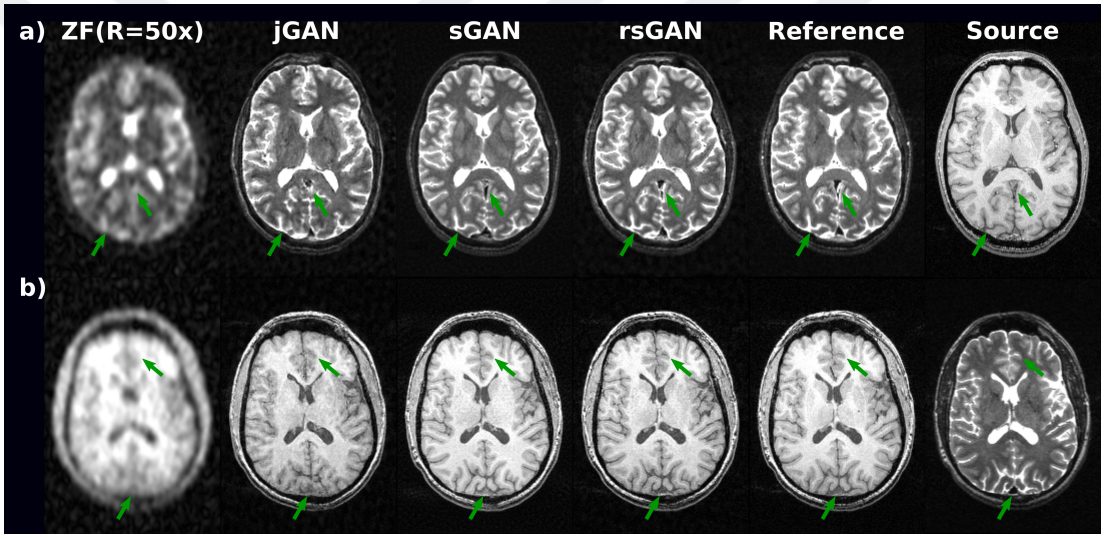

Figure 5.6: Multi-contrast images in the MIDAS dataset were recovered, where the source contrast was fully sampled and the target contrast was undersampled at R=50x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images are shown along with the fully-sampled reference image and the source-contrast image. **(b)** Recovered T_1-weighted images are shown along with the fully-sampled reference image and the source-contrast image. rsGAN yields visually accurate recovery of the target-contrast image compared to sGAN and jGAN. Sample regions that are better recovered by rsGAN are marked with arrows.

Table 5.2: Quality of Recovered Images in the MIDAS Dataset

	rGAN		jGAN		rsGAN(R_{T_2} =1)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	**32.41** ±**0.31**	94.11 ±0.29	32.35 ±0.31	94.33 ±0.30	32.17 ±0.39	**94.53**† ±**0.31**
R=10x	29.52 ±0.33	90.75 ±0.40	29.42 ±0.31	91.32 ±0.41	**29.94**† ±**0.41**	**92.23**† ±**0.45**
R=20x	27.24 ±0.31	86.15 ±0.50	27.72 ±0.26	87.77 ±0.52	**28.35**† ±**0.39**	**90.03**† ±**0.52**
R=30x	26.44 ±0.25	83.91 ±0.60	26.53 ±0.18	85.44 ±0.55	**27.94**† ±**0.36**	**89.28**† ±**0.56**
R=40x	25.73 ±0.16	82.10 ±0.66	26.21 ±0.21	83.98 ±0.65	**27.41**† ± **0.31**	**88.66**† ±**0.60**
R=50x	25.03 ±0.32	80.52 ±0.66	25.41 ±0.27	82.43 ±0.65	**27.05**† ±**0.32**	**88.21**† ±**0.66**
sGAN						
PSNR			SSIM			
25.57±0.27			85.94± 0.82			

T_2-weighted acquisitions were taken as the source contrast, and T_1-weighted acquisitions were taken as the target contrast.

Table 5.3: Quality of Recovered Images in the BRATS Dataset

	rGAN		jGAN		rsGAN(R_{T_1} =1)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	35.54 ±0.77	98.58 ±0.10	**36.21 ±0.75**	98.69 ±0.07	36.15 ±0.74	**98.77 ±0.06**
R=10x	32.61 ±0.66	97.34 ±0.11	32.81 ±0.64	97.49 ±0.10	**33.43**† **±0.64**	**97.73**† **±0.08**
R=20x	30.47 ±0.60	95.38 ±0.18	31.17 ±0.57	95.79 ±0.15	**31.54**† **±0.55**	**96.55**† **±0.12**
R=30x	29.58 ±0.52	93.91 ±0.22	29.87 ±0.54	94.38 ±0.20	**30.97**† **±0.49**	**95.79**† **±0.15**
R=40x	28.84 ±0.57	92.85 ±0.26	28.94 ±0.57	93.34 ±0.23	**30.00**† **±0.50**	**95.28**† **±0.17**
R=50x	28.12 ±0.53	91.81 ±0.29	28.35 ±0.50	92.39 ±0.25	**30.09**† **±0.39**	**94.88**† **±0.19**
sGAN						
PSNR		SSIM				
24.78±0.48		91.69± 0.44				

T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast.

Table 5.4: Quality of Recovered Images in the BRATS Dataset

	rGAN		jGAN		rsGAN(R_{T_2} =1)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	35.93 ±0.57	98.71 ±0.08	36.20 ±0.62	98.78 ±0.09	**36.46 ±0.60**	**98.86† ±0.10**
R=10x	32.56 ±0.51	97.55 ±0.14	32.63 ±0.58	97.60 ±0.17	**33.58† ±0.58**	**97.96† ±0.16**
R=20x	29.80 ±0.51	95.82 ±0.19	30.12 ±0.54	96.03 ±0.20	**31.18† ±0.51**	**96.82† ±0.22**
R=30x	28.52 ±0.49	94.55 ±0.19	28.94 ±0.48	94.94 ±0.20	**30.16† ±0.55**	**96.29† ±0.25**
R=40x	27.77 ±0.49	93.47 ±0.19	28.03 ±0.52	93.96 ±0.21	**29.53† ±0.61**	**95.86† ±0.29**
R=50x	27.04 ±0.46	92.53 ±0.21	27.26 ±0.56	93.10 ±0.22	**29.19† ±0.61**	**95.58† ±0.30**
sGAN						
PSNR				SSIM		
24.78±0.81				91.69± 0.44		

T_2-weighted acquisitions were taken as the source contrast, and T_1-weighted acquisitions were taken as the target contrast.

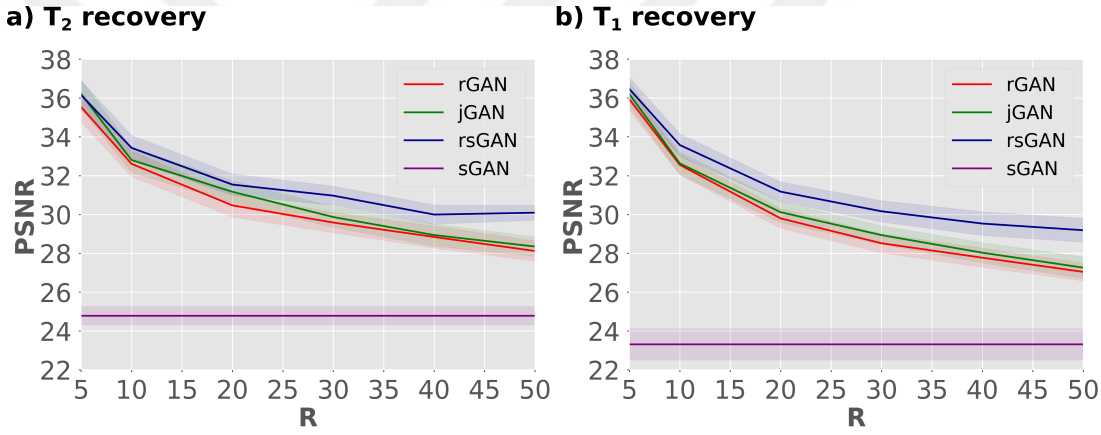

Figure 5.7: The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1- and T_2-weighted images from the BRATS dataset. The acquisition for the source contrast was fully sampled, and the acquisition for the target contrast was undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. PSNR was measured between recovered and fully-sampled reference target-contrast images. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_1 is the source contrast and T_2 is the target contrast. **(b)** PSNR (mean±standard error) when T_2 is the source contrast and T_1 is the target contrast.

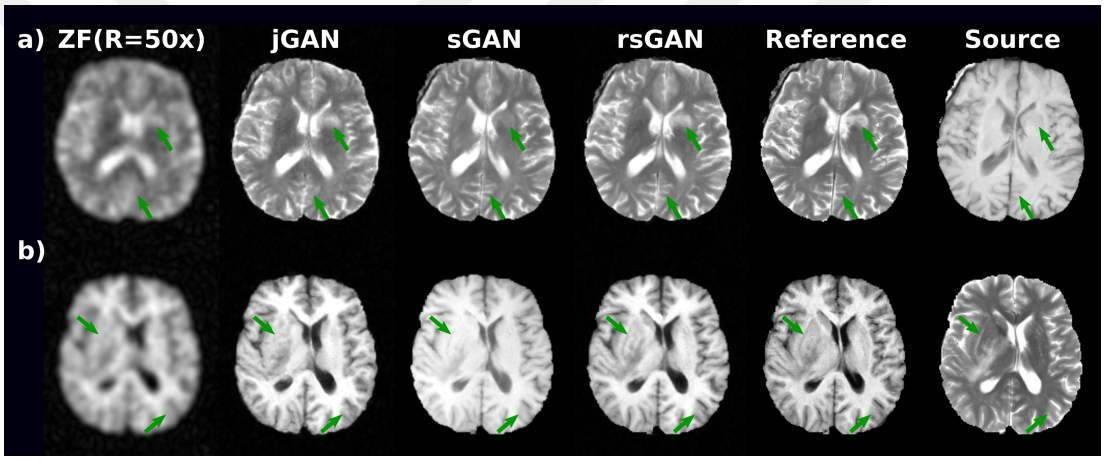

Figure 5.8: Multi-contrast images in the BRATS dataset were recovered, where the source contrast was fully sampled and the target contrast was undersampled at R=50x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images along with the fully-sampled reference image and the source-contrast image. **(b)** Recovered T_1-weighted images along with the fully-sampled reference image and the source-contrast image. rsGAN yields visually superior images compared to sGAN and jGAN. Note that sGAN suffers from either loss of features in the target contrast or synthesis of artefactual features. Meanwhile, jGAN suffers from excessive loss of high spatial frequency information. Sample regions that are more accurately recovered by rsGAN are marked with arrows.

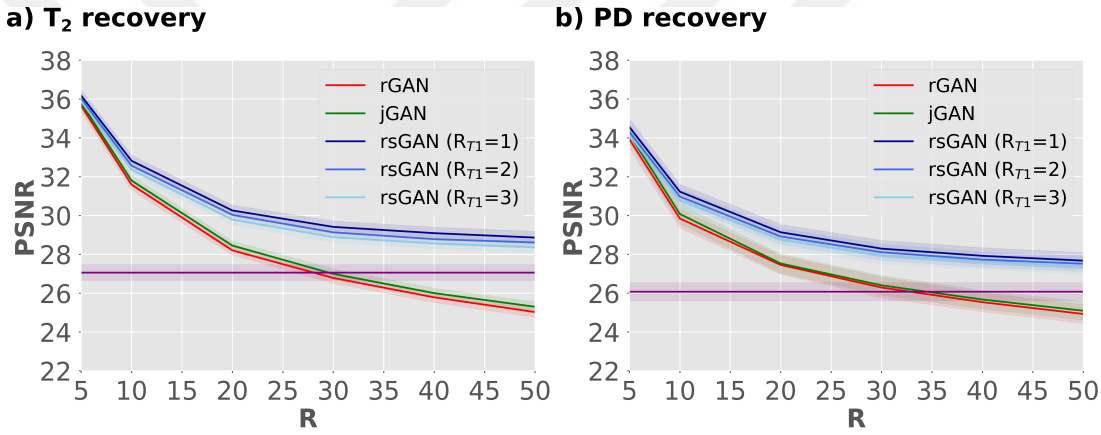

Figure 5.9: The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1-, T_2- and PD-weighted images from the IXI dataset. The acquisition for the source contrast (T_1-weighted) was lightly undersampled by R_{T_1} =1x, 2x, 3x, and the acquisitions for the target contrasts (T_2- and PD-weighted) were heavily undersampled by R=5x, 10x, 20x, 30x, 40x, 50x. **(a)** PSNR (mean±standard error) across the test subjects for rsGAN, rGAN, jGAN, and sGAN when T_2 is the target contrast. **(b)** PSNR (mean±standard error) for sGAN, rGAN, jGAN, and rsGAN when PD is the target contrast. As expected, rsGAN outperforms sGAN, rGAN, and jGAN at all R. At the same time, performance of rsGAN is highly similar for distinct values of R_{T_1} .

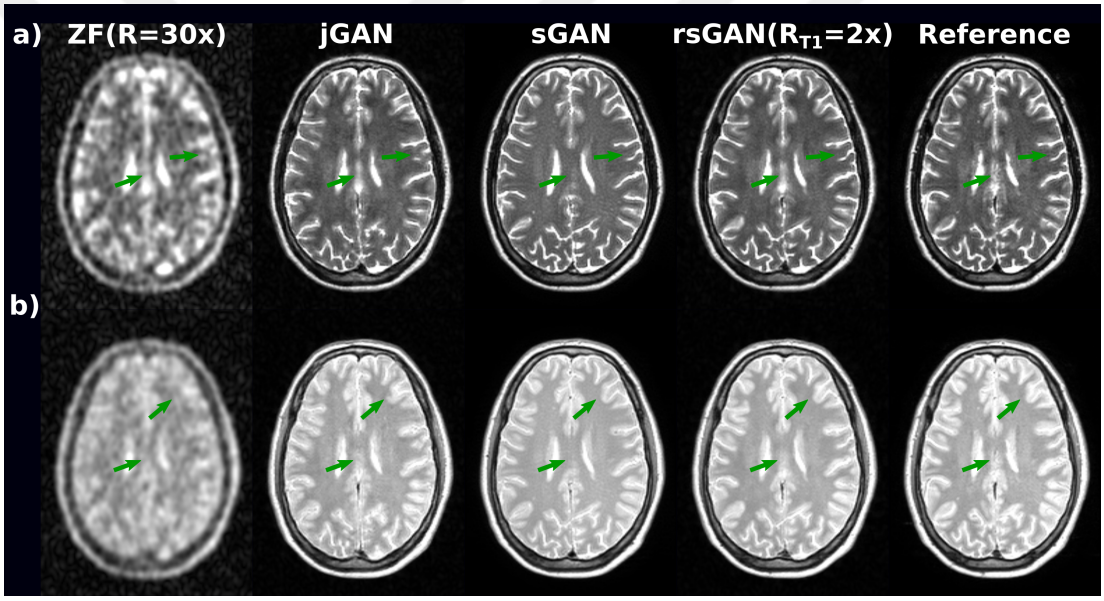

Figure 5.10: Multi-contrast images in the IXI dataset were recovered, where the source contrast (T_1-weighted) was lightly undersampled at R_{T_1} =2x, and the target contrasts (T_2- and PD-weighted) were heavily undersampled at R=30x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images. **(b)** Recovered PD-weighted images. Samples regions where rsGAN yields sharper images and improved suppression of aliasing artifacts are marked with arrows.

Table 5.5: Quality of recovered T_2-weighted images in the IXI dataset

	rGAN		**jGAN**		**rsGAN (R_{T_1} =1)**		**rsGAN (R_{T_1} =2)**		**rsGAN (R_{T_1} =3)**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	35.66 ±0.32	97.34 ± 0.16	35.80 ±0.29	97.46 ± 0.15	**36.18**† **±0.31**	**97.70**† **± 0.14**	36.01 ±0.29	97.58 ± 0.15	35.88 ±0.30	97.50 ± 0.15
R=10x	31.59 ±0.28	94.43 ± 0.34	31.80 ±0.27	94.76 ± 0.32	**32.82**† **±0.29**	**95.91**† **± 0.26**	32.56 ±0.29	95.63 ± 0.28	32.37 ±0.29	95.41 ± 0.29
R=20x	28.20 ±0.25	90.08 ± 0.55	28.45 ±0.24	90.69 ± 0.53	**30.26**† **±0.28**	**93.87**† **± 0.40**	30.03 ±0.26	93.46 ± 0.41	29.78 ±0.26	93.11 ± 0.42
R=30x	26.78 ±0.27	87.30 ± 0.69	26.99 ±0.26	88.09 ± 0.66	**29.41**† **±0.30**	**93.09**† **± 0.46**	29.13 ±0.29	92.59 ± 0.47	28.88 ±0.29	92.18 ± 0.50
R=40x	25.78 ±0.27	85.12 ± 0.82	26.00 ±0.26	85.83 ± 0.76	**29.09**† **±0.32**	**92.71**† **± 0.51**	28.78 ±0.28	92.18 ± 0.51	28.56 ±0.27	91.76 ± 0.52
R=50x	25.02 ±0.28	83.16 ± 0.95	25.30 ±0.26	84.08 ± 0.87	**28.86**† **±0.33**	**92.48**† **± 0.52**	28.61 ±0.30	91.96 ± 0.53	28.36 ±0.29	91.51 ± 0.55

sGAN	
PSNR	SSIM
27.06±0.41	90.97± 0.72

T_1-weighted acquisitions accelerated to various degrees (R_{T_1}) were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts. PSNR and %SSIM values (mean±standard error) for T_2-weighted images across the test subjects are listed for rGAN, jGAN, sGAN, and rsGAN. The highest PSNR and SSIM values in each row are marked in bold font, and the significantly better performing values ($p<0.05$) among rGAN, jGAN, sGAN, and rsGAN(T_1 =1) are marked with the '†' symbol.

Table 5.6: Quality of recovered PD-weighted images in the IXI dataset

	rGAN		**jGAN**		**rsGAN (R_{T_1} =1)**		**rsGAN (R_{T_1} =2)**		**rsGAN (R_{T_1} =3)**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	33.93 ±0.48	97.46 ± 0.16	34.19 ±0.42	97.63 ± 0.15	**34.56**† **±0.44**	**97.91**† **± 0.13**	34.30 ±0.41	97.79 ± 0.14	34.15 ±0.43	97.71 ± 0.14
R=10x	29.84 ±0.49	94.62 ± 0.33	30.08 ±0.51	94.99 ± 0.31	**31.23 ±0.39**	**96.12 ± 0.25**	30.97 ±0.40	95.84 ± 0.26	30.76 ±0.39	95.64 ± 0.27
R=20x	27.45 ±0.48	90.35 ± 0.58	27.53 ±0.49	91.03 ± 0.55	**29.13**† **±0.41**	**94.05**† **± 0.38**	28.92 ±0.41	93.62 ± 0.39	28.74 ±0.42	93.30 ± 0.41
R=30x	26.29 ±0.54	87.68 ± 0.74	26.40 ±0.50	88.39 ± 0.73	**28.29**† **±0.43**	**93.24**† **± 0.46**	28.11 ±0.42	92.74 ± 0.48	27.94 ±0.44	92.35 ± 0.50
R=40x	25.53 ±0.49	85.42 ± 0.91	25.67 ±0.48	86.20 ± 0.85	**27.92**† **±0.42**	**92.74**† **± 0.51**	27.73 ±0.40	92.21 ± 0.52	27.63 ±0.39	91.79 ± 0.54
R=50x	24.92 ±0.48	83.70 ± 1.02	25.09 ±0.47	84.63 ± 0.95	**27.67**† **±0.42**	**92.56**† **± 0.53**	27.51 ±0.39	92.00 ± 0.53	27.37 ±0.40	91.56 ± 0.56

sGAN	
PSNR	SSIM
26.07±0.47	91.22± 0.67

T_1-weighted acquisitions accelerated to various degrees (R_{T_1}) were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts.

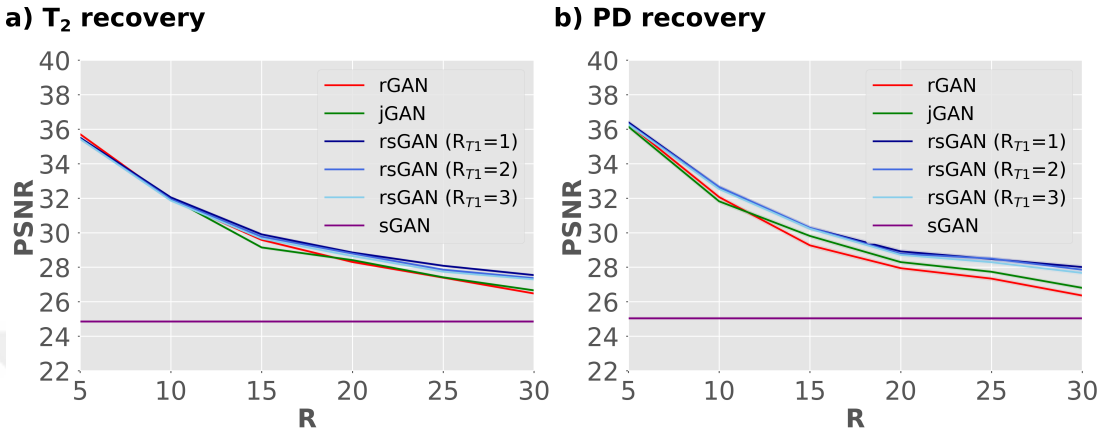

Figure 5.11: The proposed rsGAN method was demonstrated for synergistic reconstruction-synthesis of T_1-, T_2- and PD-weighted images from the multi-coil dataset. The acquisition for the source contrast (T_1-weighted) was lightly undersampled by R_{T_1} =1x, 2x, 3x, and the acquisitions for the target contrasts (T_2- and PD-weighted) were heavily undersampled by R=5x, 10x, 15x, 20x, 25x, 30x. **(a)** PSNR (mean±standard error) across the test images (coronal cross-sections) for rsGAN, rGAN, jGAN, and sGAN when T_2 is the target contrast. **(b)** PSNR (mean±standard error) for sGAN, rGAN, jGAN, and rsGAN when PD is the target contrast. As expected, rsGAN outperforms sGAN, rGAN, and jGAN at high values of R. At the same time, performance of rsGAN is highly similar for distinct values of R_{T_1}.

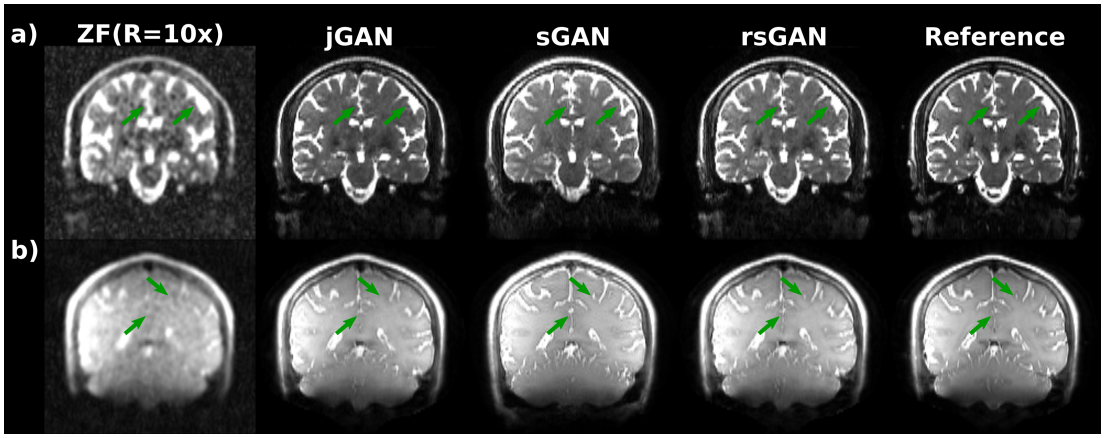

Figure 5.12: Multi-contrast images in the multi-coil dataset were recovered, where the source contrast (T_1-weighted) was fully sampled, and the target contrasts (T_2- and PD-weighted) were heavily undersampled at R=10x. Images were recovered using ZF, sGAN, jGAN and rsGAN. **(a)** Recovered T_2-weighted images. **(b)** Recovered PD-weighted images. Sample regions that are better recovered by rsGAN are marked with arrows.

Table 5.7: Quality of Recovered T_2 Images in the MIDAS Dataset

	rsGAN (R_{T_1} =1)		**sr-sGAN (R_{T_1} =1)**		**rsGAN$^-$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**37.35 ±0.12**	**97.96 ±0.10**	35.71 ±0.20	97.51 ±0.17	36.53 ±0.08	97.57 ±0.11
R=10x	**34.04 ± 0.13**	**96.30 ±0.19**	33.23 ±0.15	95.94 ±0.23	33.36 ±0.12	95.73 ±0.21
R=20x	**31.23 ±0.17**	**93.97 ±0.30**	30.96 ±0.14	93.89 ±0.30	31.03 ±0.15	93.87 ±0.29
R=30x	**30.45 ±0.16**	**93.25 ±0.33**	30.12 ±0.18	92.95 ±0.36	30.02 ±0.17	92.78 ±0.35
R=40x	**29.89 ±0.17**	**92.76 ±0.35**	29.68 ±0.18	92.51 ±0.38	29.75 ±0.17	92.50 ±0.39
R=50x	29.28 ±0.21	92.09 ±0.41	**29.46 ± 0.18**	**92.30 ±0.38**	29.20 ±0.19	91.99 ±0.38

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.8: Quality of Recovered T_1 Images in the MIDAS Dataset

	rsGAN (R_{T_2} =1)		**sr-sGAN (R_{T_2} =1)**		**rsGAN$^-$ (R_{T_2} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**32.17 ±0.39**	**94.53 ±0.31**	30.97 ±0.33	93.37 ±0.46	31.76 ±0.41	93.89 ±0.33
R=10x	**29.94 ±0.41**	**92.23 ±0.45**	29.62 ±0.34	91.77 ±0.51	29.47 ±0.38	91.45 ±0.41
R=20x	**28.35 ±0.39**	**90.03 ±0.52**	27.91 ±0.38	89.62 ±0.58	28.15 ±0.36	89.50 ±0.47
R=30x	**27.94 ± 0.36**	**89.28 ±0.56**	27.43 ±0.31	88.83 ±0.62	27.68 ±0.36	89.01 ±0.58
R=40x	27.41 ±0.31	88.66 ±0.60	27.41 ±0.32	**88.71 ±0.58**	27.38 ±0.31	88.61 ±0.61
R=50x	27.05 ±0.32	88.21 ±0.66	**27.34 ± 0.28**	**88.50 ±0.60**	27.12 ±0.33	88.17 ±0.68

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_2-weighted acquisitions were taken as the source contrast, and T_1-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.9: Quality of Recovered T_2 Images in the BRATS Dataset

	rsGAN (R_{T_1} =1)		**sr-sGAN (R_{T_1} =1)**		**rsGAN$^-$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**36.15 ±0.74**	**98.77 ±0.06**	33.26 ±0.82	98.29 ±0.07	34.40 ±0.74	98.46 ±0.08
R=10x	**33.43 ±0.64**	**97.73 ±0.08**	31.82 ±0.61	97.20 ±0.09	32.16 ±0.58	97.29 ±0.09
R=20x	**31.54 ±0.55**	**96.55 ±0.12**	30.55 ±0.51	96.08 ±0.17	30.51 ±0.57	96.01 ±0.13
R=30x	**30.97 ±0.49**	**95.79 ±0.15**	29.52 ±0.51	95.27 ±0.18	29.83 ±0.49	95.28 ±0.15
R=40x	**30.00 ± 0.50**	**95.28 ±0.17**	29.02 ±0.44	94.82 ±0.19	28.88 ±0.54	94.63 ±0.18
R=50x	**30.09 ±0.39**	**94.88 ±0.19**	29.12 ±0.44	94.53 ±0.20	28.57 ±0.46	94.28 ±0.19

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.10: Quality of Recovered T_1 Images in the BRATS Dataset

	rsGAN (R_{T_2} =1)		**sr-sGAN (R_{T_2} =1)**		**rsGAN$^-$ (R_{T_2} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**36.46 ±0.60**	**98.86 ±0.10**	34.03 ±0.64	98.37 ±0.14	35.15 ±0.57	98.60 ±0.10
R=10x	**33.58 ±0.58**	**97.96 ±0.16**	32.43 ±0.60	97.51 ±0.20	32.51 ±0.48	97.61 ±0.15
R=20x	**31.18 ±0.51**	**96.82 ±0.22**	30.97 ±0.60	96.51 ±0.26	30.76 ±0.52	96.48 ±0.22
R=30x	30.16 ±0.55	**96.29 ±0.25**	**30.25 ±0.62**	95.98 ±0.30	29.83 ±0.57	95.94 ±0.25
R=40x	29.53 ±0.61	**95.86 ±0.29**	**29.80 ±0.63**	95.65 ±0.31	29.49 ±0.49	95.53 ±0.27
R=50x	29.19 ±0.61	**95.58 ±0.30**	**29.20 ±0.58**	95.35 ±0.32	28.82 ±0.44	95.07 ±0.29

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_2-weighted acquisitions were taken as the source contrast, and T_1-acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.11: Quality of Recovered T_2 Images in the IXI Dataset

	rsGAN (R_{T_1} =1)		**sr-sGAN (R_{T_1} =1)**		**rsGAN$^-$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**36.18 ±0.31**	**97.70 ±0.14**	34.73 ±0.27	97.25 ±0.16	35.29 ±0.27	97.32 ±0.16
R=10x	**32.82 ±0.29**	**95.91 ±0.26**	32.14 ±0.26	95.63 ±0.27	32.14 ±0.28	95.38 ±0.29
R=20x	**30.26 ±0.28**	**93.87 ±0.40**	30.17 ±0.30	93.80 ±0.42	30.12 ±0.27	93.68 ±0.39
R=30x	29.41 ±0.30	93.09 ±0.46	**29.55 ±0.29**	**93.20 ±0.46**	29.37 ±0.33	92.93 ±0.48
R=40x	29.09 ±0.32	92.71 ±0.51	**29.30 ±0.32**	**92.89 ±0.49**	28.92 ±0.30	92.41 ±0.50
R=50x	**28.86 ±0.33**	**92.48 ±0.52**	28.83 ±0.30	92.45 ±0.52	28.54 ±0.29	92.12 ±0.53

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_1-weighted acquisitions were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.12: Quality of Recovered PD Images in the IXI Dataset

	rsGAN (R_{T_1} =1)		**sr-sGAN (R_{T_1} =1)**		**rsGAN$^-$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**34.56 ±0.44**	**97.91 ±0.13**	33.21 ±0.36	97.48 ±0.16	33.91 ±0.38	97.55 ±0.15
R=10x	**31.23 ±0.39**	**96.12 ±0.25**	30.68 ±0.39	95.80 ±0.28	30.68 ±0.38	95.52 ±0.29
R=20x	**29.13 ±0.41**	**94.05 ±0.38**	28.98 ±0.39	93.99 ±0.41	28.84 ±0.42	93.76 ±0.40
R=30x	28.29 ±0.43	**93.24 ±0.46**	**28.42 ±0.40**	93.19 ±0.46	28.17 ±0.42	93.00 ±0.48
R=40x	27.92 ±0.42	92.74 ±0.51	**28.21 ±0.38**	**92.90 ±0.50**	27.85 ±0.43	92.56 ±0.53
R=50x	27.67 ±0.42	**92.56 ±0.53**	**27.68 ±0.44**	92.51 ±0.54	27.62 ±0.42	92.26 ±0.55

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, sr-GAN, and rsGAN$^-$. T_1-weighted acquisitions were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.13: Quality of recovered T_2-weighted images in the multi-coil dataset

	rGAN		**jGAN**		**rsGAN (R_{T_1} =1)**		**rsGAN (R_{T_1} =2)**		**rsGAN (R_{T_1} =3)**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	**35.71**† **±0.08**	**97.62 ± 0.03**	35.47 ±0.08	97.53 ± 0.03	35.51 ±0.08	97.61 ± 0.03	35.44 ±0.08	97.56 ± 0.03	35.44 ±0.08	97.53 ± 0.03
R=10x	31.93 ±0.07	95.27 ± 0.06	32.01 ±0.07	95.40 ± 0.06	**32.05 ±0.08**	**95.64**† **± 0.05**	31.94 ±0.08	95.52 ± 0.06	31.86 ±0.08	95.42 ± 0.06
R=15x	29.58 ±0.07	92.78 ± 0.09	29.15 ±0.08	91.92 ± 0.11	**29.90**† **±0.09**	**93.66**† **± 0.09**	29.77 ±0.09	93.42 ± 0.09	29.66 ±0.09	93.22 ± 0.10
R=20x	28.30 ±0.07	90.87 ± 0.11	28.41 ±0.07	91.22 ± 0.10	**28.85**† **±0.08**	**92.42**† **± 0.10**	28.79 ±0.08	92.21 ± 0.10	28.65 ±0.08	91.96 ± 0.11
R=25x	27.40 ±0.07	89.59 ± 0.12	27.41 ±0.07	89.33 ± 0.13	**28.08**† **±0.07**	**91.21**† **± 0.11**	27.86 ±0.07	90.67 ± 0.12	27.74 ±0.07	90.40 ± 0.11
R=30x	26.48 ±0.06	87.80 ± 0.14	26.66 ±0.07	87.97 ± 0.13	**27.54**† **±0.08**	**90.56**† **± 0.13**	27.37 ±0.07	90.06 ± 0.13	27.29 ±0.07	89.77 ± 0.13

sGAN	
PSNR	SSIM
28.45±0.07	87.16± 0.19

T_1-weighted acquisitions accelerated to various degrees (R_{T_1}) were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts.

Table 5.14: Quality of recovered PD-weighted images in the multi-coil dataset

	rGAN		**jGAN**		**rsGAN (R_{T_1} =1)**		**rsGAN (R_{T_1} =2)**		**rsGAN (R_{T_1} =3)**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	36.31 ±0.11	97.55 ± 0.05	36.15 ±0.12	97.53 ± 0.05	**36.40**† **±0.12**	**97.74**† **± 0.05**	36.32 ±0.11	97.66 ± 0.05	36.28 ±0.11	97.61 ± 0.05
R=10x	32.06 ±0.12	94.88 ± 0.09	31.82 ±0.12	95.19 ± 0.08	32.64† ±0.11	**95.77**† **± 0.07**	**32.65 ±0.11**	95.58 ± 0.08	32.53 ±0.11	95.41 ± 0.08
R=15x	29.27 ±0.12	91.72 ± 0.16	29.81 ±0.11	92.35 ± 0.14	30.28† ±0.12	**93.91**† **± 0.10**	**30.29 ±0.11**	93.61 ± 0.11	30.23 ±0.11	93.41 ± 0.11
R=20x	27.94 ±0.11	89.83 ± 0.17	28.30 ±0.11	90.32 ± 0.17	**28.92**† **±0.12**	**92.05**† **± 0.16**	28.78 ±0.11	91.76 ± 0.16	28.71 ±0.11	91.51 ± 0.16
R=25x	27.34 ±0.11	88.28 ± 0.20	27.73 ±0.09	88.96 ± 0.18	28.48† ±0.11	**92.01**† **± 0.13**	**28.50 ±0.11**	91.62 ± 0.14	28.29 ±0.11	91.18 ± 0.14
R=30x	26.36 ±0.10	86.07 ± 0.25	26.80 ±0.10	87.52 ± 0.20	**28.01**† **±0.11**	**91.38**† **± 0.14**	27.85 ±0.11	90.86 ± 0.14	27.67 ±0.11	90.44 ± 0.15

sGAN	
PSNR	SSIM
25.03±0.08	88.27± 0.17

T_1-weighted acquisitions accelerated to various degrees (R_{T_1}) were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts.

Table 5.15: Quality of Recovered Images in the IXI Dataset

	rGAN		**jGAN**		**rsGAN(R_{T_1} =1)**		**rGAN $_{All}$**		**jGAN$_{All}$**		**rsGAN$_{All}$(R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
	T_2-recovery											
R=5x	35.22 ±0.18	97.22 ±0.11	35.32 ±0.17	97.33 ±0.11	35.63 ±0.19	97.51 ±0.11	35.57 ±0.19	97.33 ±0.12	35.65 ±0.20	97.44 ±0.12	**35.95 ±0.19**	**97.66 ±0.11**
R=10x	31.12 ±0.17	93.99 ±0.26	31.21 ±0.17	94.21 ±0.25	32.22 ±0.19	95.42 ±0.20	31.50 ±0.17	94.40 ±0.24	31.71 ±0.17	94.68 ±0.23	**32.52 ± 0.19**	**95.66 ±0.20**
R=20x	27.93 ±0.14	89.59 ±0.36	28.07 ±0.15	90.06 ±0.35	29.67 ±0.19	92.98 ±0.27	28.11 ±0.14	89.88 ±0.37	28.33 ±0.17	90.48 ±0.36	**29.90 ±0.17**	**93.38 ±0.27**
R=30x	26.53 ±0.16	86.82 ±0.46	26.68 ±0.16	87.46 ±0.44	28.79 ±0.24	91.95 ±0.35	26.72 ±0.15	87.15 ±0.48	26.89 ±0.16	87.81 ±0.45	**29.15 ± 0.21**	**92.55 ±0.32**
R=40x	25.57 ±0.16	84.69 ±0.56	25.78 ±0.15	85.25 ±0.51	28.39 ±0.21	91.42 ±0.36	25.70 ±0.16	84.96 ±0.57	26.00 ±0.18	85.77 ±0.56	**28.56 ±0.20**	**91.89 ±0.36**
R=50x	24.91 ±0.16	82.79 ±0.67	25.15 ±0.15	83.66 ±0.60	28.16 ±0.23	91.08 ±0.38	25.12 ±0.17	83.48 ±0.65	25.26 ±0.17	84.02 ±0.62	**28.30 ±0.22**	**91.58 ±0.38**
	PD-recovery											
R=5x	33.63 ±0.26	97.29 ±0.12	33.74 ±0.26	97.44 ±0.11	34.11 ±0.25	97.72 ±0.10	34.02 ±0.33	97.45 ±0.11	34.16 ±0.30	97.56 ±0.11	**34.55 ±0.29**	**97.80 ±0.10**
R=10x	29.55 ±0.32	94.07 ±0.26	29.76 ±0.31	94.52 ±0.22	30.71 ±0.24	95.55 ±0.19	29.92 ±0.34	94.45 ±0.24	30.15 ±0.37	94.79 ±0.22	**31.05 ± 0.27**	**95.81 ±0.19**
R=20x	27.03 ±0.30	89.61 ±0.40	27.16 ±0.30	90.24 ±0.38	28.44 ±0.23	92.87 ±0.28	27.20 ±0.34	89.87 ±0.42	27.40 ±0.36	90.48 ±0.40	**28.79 ±0.25**	**93.36 ±0.29**
R=30x	25.87 ±0.32	86.84 ±0.52	25.90 ±0.28	87.42 ±0.51	27.66 ±0.25	91.88 ±0.34	26.09 ±0.32	87.09 ±0.55	26.22 ±0.35	87.84 ±0.51	**28.04 ±0.26**	**92.36 ±0.34**
R=40x	25.15 ±0.28	84.49 ±0.65	25.32 ±0.28	85.25 ±0.61	27.23 ±0.26	91.16 ±0.38	25.38 ±0.28	84.96 ±0.66	25.63 ±0.30	85.69 ±0.63	**27.66 ±0.25**	**91.87 ±0.36**
R=50x	24.57 ±0.25	82.77 ±0.73	24.74 ±0.26	83.60 ±0.69	26.97 ±0.26	90.81 ±0.39	24.86 ±0.29	83.26 ±0.75	24.96 ±0.31	84.15 ±0.69	**27.38 ±0.27**	**91.59 ±0.38**

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rGAN, jGAN, rsGAN, rGAN$_{\text{All}}$, jGAN$_{\text{All}}$, and rsGAN$_{\text{All}}$. T_1-weighted acquisitions were taken as the source contrast, and T_2- and PD-weighted acquisitions were taken as the target contrasts. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.16: Quality of Recovered T_2 Images in the BRATS Dataset

	rsGAN (R_{T_1} =1)		**rsGAN$_m$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**36.15±0.74**	**98.77± 0.06**	35.93±0.71	98.74± 0.06
R=10x	**33.43±0.64**	**97.73± 0.08**	33.42±0.59	97.69± 0.08
R=20x	31.54±0.55	**96.55± 0.12**	**31.74±0.51**	96.50± 0.11
R=30x	**30.97±0.49**	**95.79± 0.15**	30.48±0.43	95.69± 0.15
R=40x	**30.00±0.50**	**95.28± 0.17**	29.41±0.49	95.21± 0.20
R=50x	**30.09±0.39**	**94.88± 0.19**	29.45±0.40	94.84± 0.19

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN, and rsGAN$_{\text{m}}$. T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.17: Quality of Recovered T_2 Images in the BRATS Dataset

	rGAN$_{MC}$		**rGAN**		**jGAN$_{MC}$**		**jGAN**		**rsGAN**	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
R=5x	35.94 ±0.77	98.69 ±0.08	35.54 ±0.77	98.58 ±0.10	35.87 ±0.79	98.71 ±0.07	**36.21 ± 0.75**	98.69 ±0.07	36.15 ±0.74	**98.77 ±0.06**
R=10x	32.37 ±0.68	97.32 ±0.12	32.61 ±0.66	97.34 ±0.11	32.63 ±0.66	97.45 ±0.10	32.81 ±0.64	97.49 ±0.10	**33.43 ±0.64**	**97.73 ±0.08**
R=20x	30.56 ±0.59	95.41 ±0.18	30.47 ±0.60	95.38 ±0.18	30.49 ±0.62	95.69 ±0.16	31.17 ±0.57	95.79 ±0.15	**31.54 ±0.55**	**96.55 ±0.12**
R=30x	29.55 ±0.54	93.89 ±0.22	29.58 ±0.52	93.91 ±0.22	29.75 ±0.52	94.32 ±0.19	29.87 ±0.54	94.38 ±0.20	**30.97 ±0.49**	**95.79 ±0.15**
R=40x	28.77 ±0.56	92.79 ±0.26	28.84 ±0.57	92.85 ±0.26	29.01 ±0.56	93.32 ±0.23	28.94 ±0.57	93.34 ±0.23	**30.00 ±0.50**	**95.28 ±0.17**
R=50x	28.17 ±0.53	91.88 ±0.30	28.12 ±0.53	91.81 ±0.29	28.47 ±0.50	92.40 ±0.24	28.35 ±0.50	92.39 ±0.25	**30.09 ±0.39**	**94.88 ±0.19**

sGAN$_{MC}$		**sGAN**	
PSNR	SSIM	PSNR	SSIM
24.22±0.39	91.50±0.38	24.78±0.48	91.69±0.44

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rGAN$_{MC}$, rGAN, jGAN$_{MC}$, jGAN, sGAN, sGAN$_{MC}$, and rsGAN (R_{T_1}=1). T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.18: Quality of Recovered FLAIR Images in the BRATS Dataset

	rGAN		**jGAN**		**rsGAN (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	35.57 ±0.50	97.89 ±0.16	35.77 ±0.52	97.94 ±0.15	**35.89 ±0.55**	**97.99 ±0.16**
R=10x	32.35 ±0.44	95.88 ±0.26	32.68 ±0.46	96.09 ±0.27	**32.91 ±0.45**	**96.22 ±0.27**
R=20x	30.16 ±0.39	93.32 ±0.35	30.22 ±0.43	93.69 ±0.37	**30.84 ±0.42**	**94.15 ±0.36**
R=30x	29.08 ±0.35	91.84 ±0.39	29.08 ±0.35	92.15 ±0.41	**29.73 ±0.36**	**92.96 ±0.41**
R=40x	28.23 ±0.38	90.56 ±0.44	28.30 ±0.42	91.07 ±0.46	**29.22 ±0.39**	**92.10 ±0.42**
R=50x	27.58 ±0.45	90.45 ±0.54	27.60 ±0.38	90.23 ±0.48	**28.79 ±0.40**	**91.64 ±0.47**

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rGAN, jGAN, and rsGAN. T_1-weighted acquisitions were taken as the source contrast, and FLAIR acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

Table 5.19: Quality of Recovered T_2 and FLAIR Images in the BRATS Dataset

	rsGAN$_1$ (R_{T_1} =1)		**rsGAN$_2$ (R_{T_1} =1)**	
	PSNR	SSIM (%)	PSNR	SSIM (%)
R=5x	**36.02±0.49**	98.38± 0.09	35.73±0.47	98.38± 0.09
R=10x	33.17±0.39	96.98± 0.15	**33.21±0.40**	**96.99± 0.14**
R=20x	**31.19±0.39**	95.35± 0.21	31.11±0.38	**95.36± 0.21**
R=30x	**30.35±0.35**	**94.38± 0.24**	30.11±0.37	94.22± 0.26
R=40x	**29.61±0.37**	**93.69± 0.25**	29.38±0.34	93.51± 0.28
R=50x	**29.44±0.34**	**93.26± 0.28**	28.74±0.32	92.98± 0.30

PSNR and %SSIM values (mean±standard error) across the test subjects are listed for rsGAN$_1$, and rsGAN$_2$. T_1-weighted acquisitions were taken as the source contrast, and T_2-weighted and FLAIR acquisitions were taken as the target contrast. The highest PSNR and SSIM values in each row are marked in bold font.

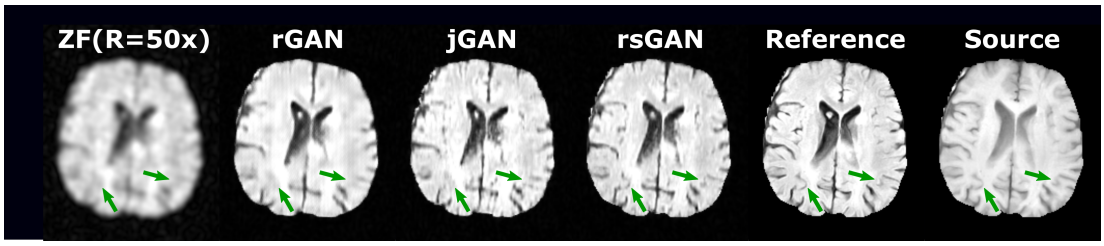

Figure 5.13: FLAIR images in the BRATS dataset were recovered, where the source contrast (T_1) was fully sampled and the target contrast (FLAIR) was undersampled at R=50x. Images recovered using ZF, rGAN, jGAN and rsGAN along with the fully-sampled reference image and the source-contrast image are shown. rsGAN yields visually superior images compared to rGAN and jGAN. Both rGAN and jGAN suffer from excessive loss of high spatial frequency information. Sample regions that are more accurately recovered by rsGAN are marked with arrows.

Chapter 6

Conclusion

The broad aim of this thesis was to address the issues associated with traditional MR image synthesis and reconstruction approaches, and provide solutions to mitigate those issues. Comparative evaluations on brain MR image datasets of healthy subjects and patients show superiority of the proposed methods against current state-of-the art methods.

The main contributions of this thesis are as follows:

- A transfer learning approach was proposed to address the issue of data scarcity in deep neural networks for MR image reconstruction. Demonstrations on brain MR images indicate that domain transferred network fine-tuned on just a few target domain images achieve on par performance with the networks trained on thousands of target domain images. The proposed approach might facilitate the use of deep neural networks for MRI reconstruction in applications where data are relatively scarce.

- Conditional generative adversarial networks were demonstrated for efficient recovery of intermediate-to-high spatial frequency in MR image synthesis. The adversarial loss enables recovery of fine structural details while the pixel-wise and perceptual losses lead to enhanced synthesis performance.

Demonstrations on healthy individuals and Glioma patients indicate superior performance of the proposed approach over current state-of-the art methods. The proposed approach might be helpful by offering a substitute for highly-corrupted or even unavailable contrasts.

- A joint reconstruction-synthesis method was proposed for recovery of highly undersampled images of target contrasts, and lightly undersampled images of source contrasts. The proposed synergistic method enables efficient recovery of highly undersampled contrasts while increasing immunity to loss or artefactual synthesis of features. The proposed method holds great promise for accelerated MR imaging in clinical practice.

Discussion and future directions

Here, we proposed multiple techniques to address the issues with DNN based MR image reconstruction and synthesis approaches. Several improvements are viable for enhanced and data-efficient recovery.

Transfer learning for MR image reconstruction

While transferring networks trained on readily available large datasets, we directly transferred the models without taking into account any prior knowledge regarding the target data. The transfer learning approach can be further improved by incorporating such knowledge by initially training the networks on images having features matching with the target domain images. This can lead to even reduced number of fine-tuning samples in the target domain.

Another line of improvement can be incorporation of scan-specific information during inference. This can lead to enhanced generalization on test images, and reduce dependency on large training sets. For instance, in case of sub-optimal population driven priors due to differences in training and testing domains, they can be complemented with scan-specific subject driven priors obtained via parallel

imaging algorithms. Alternatively, scan-specific priors can also be incorporated by tuning networks on just the acquired k-space points in a self-supervised manner, thus leading to enhanced generalization.

Another possible future research direction can be domain transfer between MR images of different organs. In such, the anatomy of interest might occupy different portions of the field of view (FOV) due to inherent shape and size differences among organs. These differences might in turn limit reconstruction performance of domain-transferred networks.

GANs for MR image synthesis

Another important avenue of the future work can be addressing the issue of limited fully sampled training in deep neural networks for MR image synthesis. Practically, it might not be feasible to acquire fully sampled images of some contrasts and the readily available datasets might only constitute undersampled images. In such cases, semi-supervised techniques can be deployed by training models on only acquired k-space data points of both source and target contrasts.

In case of unpaired images, cycle consistency loss can be employed to learn mapping from the images of source to target contrasts. However, a substantial mismatch between the voxel sizes in the source and target contrasts can lead to interpolation between the spatial sampling grids of the source and the target. In such cases, a simple solution is to resample each contrast separately to match the voxel dimensions. Alternatively, the spatial transformation between the source and target images can first be estimated via multi-modal registration. The estimated transformation can then be cascaded to the output of network. A gradient cycle consistency loss can also be incorporated to prevent the network from learning the spatial transformation between the source and the target.

Joint reconstruction and synthesis for accelerated MRI

An important practical issue can be availability of co-registered multi-contrast MR data. While this can be addressed via incorporation of a cycle-consistency loss in one-to-one source to target MR image synthesis, it can be challenging while recovering information from unpaired images of multiple source contrasts, especially when one of them is highly undersampled as in rsGAN. In such cases, a pre-registration network can incorporated to the network which can register the highly undersampled images form the target contrasts onto lightly undersampled images from the source contrast.

Here we perform synthesis and reconstruction based on 2D-models which can fail to capture global contextual information. An alternative would be to employ 3D-models that learn volumetric mappings. However, these models are data hungry as one subject is treated as a single sample. To mitigate the limitations of both 2D and 3D models, progressive 2D models across multiple orientations can be used for efficient recovery. This can enable the networks to capture global contextual information while keeping the models data-efficient.

Bibliography

[1] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[2] M. Lustig, D. Donoho, J. Santos, and J. Pauly, "Compressed Sensing MRI," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72–82, 2008.

[3] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, "Compressed sensing MRI reconstruction with cyclic loss in generative adversarial networks," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1488–1497, 2018.

[4] S. Yu, H. Dong, G. Yang, G. Slabaugh, P. L. Dragotti, X. Ye, F. Liu, S. Arridge, J. Keegan, D. Firmin, and Y. Guo, "DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1310–1321, 2018.

[5] M. Mardani, E. Gong, J. Y. Cheng, S. Vasanawala, G. Zaharchuk, M. Alley, N. Thakur, S. Han, W. Dally, J. M. Pauly, and L. Xing, "Deep generative adversarial neural networks for compressive sensing MRI," *IEEE Transactions on Medical Imaging*, vol. 38, no. 1, pp. 167–179, 2019.

[6] D. Lee, J. Yoo, S. Tak, and J. C. Ye, "Deep residual learning for accelerated mri using magnitude and phase networks," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 9, pp. 1985–1995, 2018.

[7] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a Variational Network for Reconstruction of Accelerated MRI Data," *Magnetic Resonance in Medicine*, vol. 79, no. 6, pp. 3055–3071, 2017.

[8] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep ADMM-Net for Compressive Sensing MRI," in *Advances in Neural Information Processing Systems*, 2016.

[9] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A Deep Cascade of Convolutional Neural Networks for MR Image Reconstruction," pp. 647–658, Springer, Cham, 2017.

[10] M. Mardani, H. Monajemi, V. Papyan, S. Vasanawala, D. Donoho, and J. Pauly, "Recurrent Generative Adversarial Networks for Proximal Learning and Automated Compressive Image Recovery," *arXiv preprint*, 2017.

[11] M. Akçakaya, S. Moeller, S. Weingärtner, and K. Uğurbil, "Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: Database-free deep learning for fast imaging," *Magnetic Resonance in Medicine*, vol. 81, no. 1, pp. 439–453, 2019.

[12] K. Kwon, D. Kim, and H. Park, "A parallel MR imaging method using multilayer perceptron," *Medical Physics*, vol. 44, no. 12, pp. 6209–6224, 2017.

[13] B. Zhu, J. Z. Liu, B. R. Rosen, and M. S. Rosen, "Image reconstruction by domain transform manifold learning," *Nature*, vol. 555, no. 7697, pp. 487–492, 2018.

[14] C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee, and J. K. Seo, "Deep learning for undersampled MRI reconstruction," *Physics in Medicine and Biology*, vol. 63, no. 13, p. 135007, 2018.

[15] S. Wang, Z. Su, L. Ying, X. Peng, S. Zhu, F. Liang, D. Feng, and D. Liang, "Accelerating magnetic resonance imaging via deep learning," in *IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pp. 514–517, 2016.

[16] Y. Han, J. Yoo, H. H. Kim, H. J. Shin, K. Sung, and J. C. Ye, "Deep learning with domain adaptation for accelerated projection-reconstruction MR," *Magnetic Resonance in Medicine*, vol. 80, no. 3, pp. 1189–1205, 2018.

[17] J. Y. Cheng, F. Chen, M. T. Alley, J. M. Pauly, and S. S. Vasanawala, "Highly Scalable Image Reconstruction using Deep Neural Networks with Bandpass Filtering," *arXiv preprint*, 2018.

[18] S. U. H. Dar, M. Özbey, A. B. Çatlı, and T. Çukur, "A transfer-learning approach for accelerated MRI using deep neural networks," *Magnetic Resonance in Medicine*, vol. 84, no. 2, pp. 663–685, 2020.

[19] M. Yurt, M. Özbey, S. U. H. Dar, B. Tınaz, K. K. Oğuz, and T. Çukur, "Progressively volumetrized deep generative models for data-efficient contextual learning of mr image recovery," *arXiv preprint*, 2020.

[20] M. Özbey, M. Yurt, S. U. H. Dar, B. Tınaz, K. K. Oğuz, and T. Çukur, "Three dimensional mr image synthesis with progressive generative adversarial networks," *arXiv preprint*, 2020.

[21] S. Roy, A. Carass, and J. Prince, "A compressed sensing approach for mr tissue contrast synthesis," in *Information Processing in Medical Imaging*, pp. 371–383, 2011.

[22] S. Roy, A. Carass, and J. L. Prince, "Magnetic resonance image example-based contrast synthesis," *IEEE Transactions on Medical Imaging*, vol. 32, no. 12, pp. 2348–2363, 2013.

[23] Y. Huang, L. Shao, and A. F. Frangi, "Cross-modality image synthesis via weakly-coupled and geometry co-regularized joint dictionary learning," *IEEE Transactions on Medical Imaging*, vol. 37, no. 3, pp. 815 – 827, 2018.

[24] H. Van Nguyen, K. Zhou, and R. Vemulapalli, "Cross-domain synthesis of medical images using efficient location-sensitive deep network," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 677–684, 2015.

[25] A. Chartsias, T. Joyce, M. V. Giuffrida, and S. A. Tsaftaris, "Multimodal MR synthesis via modality-invariant latent representation," *IEEE Transactions on Medical Imaging*, vol. 37, no. 3, pp. 803–814, 2018.

[26] T. Joyce, A. Chartsias, and S. A. Tsaftaris, "Robust multi-modal MR image synthesis," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 347–355, 2017.

[27] S. U. H. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast MRI with conditional generative adversarial networks," *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2375–2388, 2019.

[28] Q. Yang, N. Li, Z. Zhao, X. Fan, E. I-Chao Chang, and Y. Xu, "MRI image-to-image translation for cross-modality image registration and segmentation," *arXiv preprint*, 2018.

[29] B. Yu, L. Zhou, L. Wang, J. Fripp, and P. Bourgeat, "3D cGAN based cross-modality MR image synthesis for brain tumor segmentation," in *IEEE 15th International Symposium on Biomedical Imaging*, pp. 626–630, IEEE, 2018.

[30] B. Yu, L. Zhou, L. Wang, Y. Shi, J. Fripp, and P. Bourgeat, "Ea-GANs: Edge-Aware Generative Adversarial Networks for Cross-Modality MR Image Synthesis," *IEEE Transactions on Medical Imaging*, vol. 38, no. 7, pp. 1750–1762, 2019.

[31] A. Sharma and G. Hamarneh, "Missing MRI Pulse Sequence Synthesis using Multi-Modal Generative Adversarial Network," *arXiv preprint*, 2019.

[32] S. Olut, Y. H. Sahin, U. Demir, and G. Unal, "Generative Adversarial Training for MRA Image Synthesis Using Multi-contrast MRI," in *Predictive Intelligence in Medicine*, pp. 147–154, Springer, Cham, 2018.

[33] M. Yurt, S. U. Dar, A. Erdem, E. Erdem, and T. Çukur, "mustGAN: Multi-Stream Generative Adversarial Networks for MR Image Synthesis," *arXiv preprint*, 2019.

[34] D. Lee, J. Kim, W.-J. Moon, and J. C. Ye, "CollaGAN: Collaborative GAN for Missing Image Data Imputation," *arXiv preprint*, 2019.

[35] P. Welander, S. Karlsson, and A. Eklund, "Generative Adversarial Networks for Image-to-Image Translation on Multi-Contrast MR Images - A Comparison of CycleGAN and UNIT," *arXiv preprint*, 2018.

[36] H. Li, J. C. Paetzold, A. Sekuboyina, F. Kofler, J. Zhang, J. S. Kirschke, and B. Menze, "DiamondGAN: Unified Multi-Modal Generative Adversarial Networks for MRI Sequences Synthesis," *arXiv preprint*, 2019.

[37] A. Hagiwara, Y. Otsuka, M. Hori, Y. Tachibana, K. Yokoyama, S. Fujita, C. Andica, K. Kamagata, R. Irie, S. Koshino, T. Maekawa, L. Chougar, A. Wada, M. Y. Takemura, N. Hattori, and S. Aoki, "Improving the Quality of Synthetic FLAIR Images with Deep Learning Using a Conditional Generative Adversarial Network for Pixel-by-Pixel Image Translation.," *AJNR. American Journal of Neuroradiology*, vol. 40, no. 2, pp. 224–230, 2019.

[38] R. Mehta and T. Arbel, "RS-Net: Regression-Segmentation 3D CNN for Synthesis of Full Resolution Missing Brain MRI in the Presence of Tumours," *arXiv preprint*, 2018.

[39] S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, B. Tınaz, and T. Çukur, "Prior-guided image reconstruction for accelerated multi-contrast mri via generative adversarial networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 6, pp. 1072–1087, 2020.

[40] D. G. Nishimura, *Principles of magnetic resonance imaging*. Stanford University, 1996.

[41] S. U. H. Dar, M. Özbey, A. B. Çatlı, and T. Çukur, "A transfer-learning approach for accelerated mri using deep neural networks," *arXiv preprint*, 2017.

[42] S. U. H. Dar and T. Çukur, "Transfer learning for reconstruction of accelerated MRI acquisitions via neural networks," in *26th Annual Meeting of ISMRM*, p. 0569, 2018.

[43] K. T. Block, M. Uecker, and J. Frahm, "Undersampled radial MRI with multiple coils. Iterative image reconstruction using a total variation constraint," *Magnetic Resonance in Medicine*, vol. 57, no. 6, pp. 1086–1098, 2007.

[44] T. Çukur, M. Lustig, and D. G. Nishimura, "Improving non-contrast-enhanced steady-state free precession angiography with compressed sensing," *Magnetic Resonance in Medicine*, vol. 61, no. 5, pp. 1122–1131, 2009.

[45] H. Jung and J. C. Ye, "Performance evaluation of accelerated functional MRI acquisition using compressed sensing," in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 702–705, 2009.

[46] M. I. Menzel, E. T. Tan, K. Khare, J. I. Sperl, K. F. King, X. Tao, C. J. Hardy, and L. Marinelli, "Accelerated diffusion spectrum imaging in the human brain using compressed sensing," *Magnetic Resonance in Medicine*, vol. 66, no. 5, pp. 1226–1233, 2011.

[47] M. Doneva, P. Börnert, H. Eggers, C. Stehning, J. Sénégas, and A. Mertins, "Compressed sensing reconstruction for magnetic resonance parameter mapping," *Magnetic Resonance in Medicine*, vol. 64, no. 4, pp. 1114–1120, 2010.

[48] S. Ilbey, C. B. Top, A. Gungor, T. Çukur, E. U. Saritas, and H. E. Guven, "Fast System Calibration With Coded Calibration Scenes for Magnetic Particle Imaging," *IEEE Transaction on Medical Imaging*, vol. 38, pp. 2070–2080, Sep 2019.

[49] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1028–1041, 2011.

[50] K. Khare, C. J. Hardy, K. F. King, P. A. Turski, and L. Marinelli, "Accelerated MR imaging using compressive sensing with no free parameters," *Magnetic Resonance in Medicine*, vol. 68, no. 5, pp. 1450–1457, 2012.

[51] D. S. Weller, S. Ramani, J.-F. Nielsen, and J. A. Fessler, "Monte Carlo SURE-based parameter selection for parallel magnetic resonance imaging reconstruction," *Magnetic Resonance in Medicine*, vol. 71, no. 5, pp. 1760–1770, 2014.

[52] Y. Jun, T. Eo, T. Kim, J. Jang, and D. Hwang, "Deep Convolutional Neural Network for Acceleration of Magnetic Resonance Angiography (MRA)," in *25th Annual Meeting of ISMRM*, p. 0686, 2017.

[53] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A Deep Cascade of Convolutional Neural Networks for MR Image Reconstruction," in *International Conference on Information Processing in Medical Imaging*, pp. 647–658, 2017.

[54] Y. Han and J. C. Ye, "k-Space Deep Learning for Accelerated MRI," *arXiv preprint*, 2018.

[55] K. Kwon, D. Kim, H. Seo, J. Cho, B. Kim, and H. Park, "Learning-based Reconstruction using Artificial Neural Network for Higher Acceleration," in *24th Annual Meeting of ISMRM*, p. 1801, 2016.

[56] E. Gong, G. Zaharchuk, and J. Pauly, "Improving the PI+CS Reconstruction for Highly Undersampled Multi-contrast MRI using Local Deep Network," in *25th Annual Meeting of ISMRM*, p. 5663, 2017.

[57] S. Wang, N. Huang, T. Zhao, Y. Yang, L. Ying, and D. Liang, "1D Partial Fourier Parallel MR imaging with deep convolutional neural network," in *25th Annual Meeting of ISMRM*, p. 0642, 2017.

[58] D. Lee, J. Yoo, and J. C. Ye, "Compressed sensing and Parallel MRI using deep residual learning," in *25th Annual Meeting of ISMRM*, p. 0641, 2017.

[59] B. Zhu, J. Z. Liu, B. R. Rosen, and M. S. Rosen, "Neural Network MR Image Reconstruction with AUTOMAP: Automated Transform by Manifold Approximation," in *25th Annual Meeting of ISMRM*, p. 0640, 2017.

[60] M. Akçakaya, S. Moeller, S. Weingärtner, and K. Ugurbil, "Scan-specific Robust Artificial-neural-networks for k-space Interpolation (RAKI):

Database-free Deep Learning Reconstruction for Fast Imaging," in *26th Annual Meeting of ISMRM*, p. 0576, 2018.

[61] K. Pawar, Z. Chen, N. Shah, and G. Egan, "ReconNet: A Deep Learning Framework for Transforming Image Reconstruction into Pixel Classification," in *26th Annual Meeting of ISMRM*, p. 0574, 2018.

[62] Z. Fu, S. Mandava, M. Keerthivasan, D. Martin, M. Altbach, and A. Bilgin, "A Multi-scale Deep ResNet for Radial MR Parameter Mapping," in *26th Annual Meeting of ISMRM*, p. 0568, 2018.

[63] P. Chang, M. Liu, D. Chow, M. Khy, C. Filippi, J. Lupo, and C. Hess, "Inception-CS: Deep Learning For Sparse MR Reconstruction in Glioma Patients," in *26th Annual Meeting of ISMRM*, p. 0571, 2018.

[64] J. Lee and J. C. Ye, "Deep convolutional framelet neural network for reference-free EPI ghost correction," in *26th Annual Meeting of ISMRM*, p. 0575, 2018.

[65] S. Moeller, S. Weingärtner, K. Ugurbil, and M. Akcakaya, "Application of a Scan-Specific Deep Learning Reconstruction to Multiband/SMS Imaging," in *26th Annual Meeting of ISMRM*, p. 0577, 2018.

[66] F. Chen, V. Taviani, I. Malkiel, J. Y. Cheng, J. I. Tamir, J. Shaikh, S. T. Chang, C. J. Hardy, J. M. Pauly, and S. S. Vasanawala, "Variable-Density Single-Shot Fast Spin-Echo MRI with Deep Learning Reconstruction by Using Variational Networks," *Radiology*, vol. 289, pp. 366–373, Nov 2018.

[67] M. Mardani, E. Gong, J. Cheng, S. Vasanawala, G. Zaharchuk, L. Xing, and J. Pauly, "Deep Generative Adversarial Neural Networks for Compressed Sensing (GANCS) Automates MRI," in *26th Annual Meeting of ISMRM*, p. 0573, 2018.

[68] K. Hammernik, F. Knoll, D. K. Sodickson, and T. Pock, "Learning a Variational Model for Compressed Sensing MRI Reconstruction," in *24th Annual Meeting of ISMRM*, p. 1088, 2016.

[69] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a variational network for reconstruction of accelerated MRI data," *Magnetic Resonance in Medicine*, vol. 79, no. 6, pp. 3055–3071, 2018.

[70] F. Knoll, K. Hammernik, E. Kobler, T. Pock, M. P. Recht, and D. K. Sodickson, "Assessment of the generalization of learned image reconstruction and the potential for transfer learning," *Magnetic Resonance in Medicine*, vol. 81, pp. 116–128, Jan 2019.

[71] J. Y. Cheng, M. Mardani, M. T. Alley, J. M. Pauly, and S. S. Vasanawala, "DeepSPIRiT: Generalized Parallel Imaging using Deep Convolutional Neural Networks," in *26th Annual Meeting of ISMRM*, p. 0570, 2018.

[72] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[73] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition," in *Proceedings of the 31st International Conference on Machine Learning* (E. P. Xing and T. Jebara, eds.), vol. 32 of *Proceedings of Machine Learning Research*, (Bejing, China), pp. 647–655, PMLR, 2014.

[74] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 3320–3328, Curran Associates, Inc., 2014.

[75] M. Lustig and J. M. Pauly, "SPIRiT: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space.," *Magnetic Resonance in Medicine*, vol. 64, no. 2, pp. 457–71, 2010.

[76] M. Shahdloo, E. Ilicak, M. Tofighi, E. U. Saritas, A. E. Çetin, and T. Çukur, "Projection onto epigraph sets for rapid self-tuning compressed sensing

mri," *IEEE transactions on medical imaging*, vol. 38, no. 7, pp. 1677–1689, 2018.

[77] T. Çukur, J. M. Santos, J. M. Pauly, and D. G. Nishimura, "Variable-density parallel imaging with partially localized coil sensitivities," *IEEE Transaction on Medical Imaging*, vol. 29, pp. 1173–1181, May 2010.

[78] E. Biyik, E. Ilicak, and T. Çukur, "Reconstruction by calibration over tensors for multi-coil multi-acquisition balanced SSFP imaging," *Magnetic Resonance in Medicine*, vol. 79, pp. 2542–2554, May 2018.

[79] E. Ilicak, L. K. Senel, E. Biyik, and T. Çukur, "Profile-encoding reconstruction for multiple-acquisition balanced steady-state free precession imaging," *Magnetic Resonance in Medicine*, vol. 78, pp. 1316–1329, Oct 2017.

[80] E. Kopanoglu, A. Güngör, T. Kilic, E. U. Saritas, K. K. Oguz, T. Çukur, and H. E. Güven, "Simultaneous use of individual and joint regularization terms in compressive sensing: Joint reconstruction of multi-channel multi-contrast MRI acquisitions," *NMR in Biomedicine*, vol. 33, p. e4247, Apr 2020.

[81] L. K. Senel, T. Kilic, A. Gungor, E. Kopanoglu, H. E. Guven, E. U. Saritas, A. Koc, and T. Çukur, "Statistically Segregated k-Space Sampling for Accelerating Multiple-Acquisition MRI," *IEEE Transaction on Medical Imaging*, vol. 38, pp. 1701–1714, Jul 2019.

[82] E. Ilicak, S. Cetin, E. Bulut, K. K. Oguz, E. U. Saritas, G. Unal, and T. Çukur, "Targeted vessel reconstruction in non-contrast-enhanced steady-state free precession angiography," *NMR in Biomedicine*, vol. 29, pp. 532–544, May 2016.

[83] T. Çukur, "Accelerated phase-cycled SSFP imaging with compressed sensing," *IEEE Transaction on Medical Imaging*, vol. 34, pp. 107–115, Jan 2015.

[84] T. Çukur, M. Lustig, E. U. Saritas, and D. G. Nishimura, "Signal compensation and compressed sensing for magnetization-prepared MR angiography," *IEEE Transaction on Medical Imaging*, vol. 30, pp. 1017–1027, May 2011.

[85] H. H. Bauschke and J. M. Borwein, "On Projection Algorithms for Solving Convex Feasibility Problems," *SIAM Review*, vol. 38, pp. 367–426, Sep 1996.

[86] M. Uecker, P. Lai, M. J. Murphy, P. Virtue, M. Elad, J. M. Pauly, S. S. Vasanawala, and M. Lustig, "ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA," *Magnetic Resonance in Medicine*, vol. 71, no. 3, pp. 990–1001, 2014.

[87] E. Bullitt, D. Zeng, G. Gerig, S. Aylward, S. Joshi, J. K. Smith, W. Lin, and M. G. Ewend, "Vessel tortuosity and brain tumor malignancy: A blinded study," *Academic Radiology*, vol. 12, no. 10, pp. 1232–1240, 2005.

[88] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[89] K. T. Kwon, H. H. Wu, T. Shin, T. Çukur, M. Lustig, and D. G. Nishimura, "Three-dimensional magnetization-prepared imaging using a concentric cylinders trajectory," *Magnetic Resonance in Medicine*, vol. 71, pp. 1700–1710, May 2014.

[90] N. K. Bangerter, T. Çukur, B. A. Hargreaves, B. S. Hu, J. H. Brittain, D. Park, G. E. Gold, and D. G. Nishimura, "Three-dimensional fluid-suppressed T2-prep flow-independent peripheral angiography using balanced SSFP," *Magnetic Resonance Imaging*, vol. 29, pp. 1119–1124, Oct 2011.

[91] T. Çukur, A. Shimakawa, H. Yu, B. A. Hargreaves, B. S. Hu, D. G. Nishimura, and J. H. Brittain, "Magnetization-prepared IDEAL bSSFP: a flow-independent technique for noncontrast-enhanced peripheral angiography," *Journal of Magnetic Resonance Imaging*, vol. 33, pp. 931–939, Apr 2011.

[92] T. Zhang, J. M. Pauly, S. S. Vasanawala, and M. Lustig, "Coil compression for accelerated imaging with Cartesian sampling.," *Magnetic Resonance in Medicine*, vol. 69, no. 2, pp. 571–82, 2013.

[93] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[94] D. P. Kingma and J. L. Ba, "Adam: a Method for Stochastic Optimization," in *International Conference on Learning Representations*, 2015.

[95] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[96] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, and G. Brain, "TensorFlow: Large-Scale Machine Learning on Heterogenous Distributed Systems," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)*, pp. 265–284, 2016.

[97] P. Domingos, "A few useful things to know about machine learning," *Communications of the ACM*, vol. 55, no. 10, p. 78, 2012.

[98] G. J. Burton and I. R. Moorhead, "Color and spatial structure in natural scenes," *Applied Optics*, vol. 26, pp. 157–170, Jan 1987.

[99] E. Çelik, S. U. H. Dar, Ö. Yılmaz, Ü. Keleş, and T. Çukur, "Spatially informed voxelwise modeling for naturalistic fMRI experiments," *Neuroimage*, vol. 186, pp. 741–757, Feb 2019.

[100] M. Shahdloo, E. Çelik, and T. Çukur, "Biased competition in semantic representation during natural visual search," *Neuroimage*, vol. 216, p. 116383, 2020.

[101] Ö. Yılmaz, E. Çelik, and T. Çukur, "Informed feature regularization in voxelwise modeling for naturalistic fMRI experiments," *European Journal Neuroscience*, vol. 52, pp. 3394–3410, Sep 2020.

[102] S. Tokgoz, D. Aydogdu, B. Ilhan, Y. Sahin, N. Bariseri, B. M. Ozturkler, and T. Çukur, "Musical mirror-symmetrical movement tasks: comparison of rhythm versus melody-playing," *Neuroreport*, vol. 31, pp. 523–529, May 2020.

[103] T. Çukur, A. G. Huth, S. Nishimoto, and J. L. Gallant, "Functional subdomains within human FFA," *Journal of Neuroscience*, vol. 33, pp. 16748–16766, Oct 2013.

[104] T. Çukur, A. G. Huth, S. Nishimoto, and J. L. Gallant, "Functional Subdomains within Scene-Selective Cortex: Parahippocampal Place Area, Retrosplenial Complex, and Occipital Place Area," *Journal of Neuroscience*, vol. 36, no. 40, pp. 10257–10273, 2016.

[105] T. Çukur, S. Nishimoto, A. G. Huth, and J. L. Gallant, "Attention during natural vision warps semantic representation across the human brain," *Nature Neuroscience*, vol. 16, pp. 763–770, Jun 2013.

[106] T. Zhang, J. S. Gao, T. Çukur, and J. L. Gallant, "Voxel-based state space modeling recovers task-related cognitive states in naturalistic fmri experiments," *Frontiers in Neuroscience*, 2020.

[107] I. Kiremitçi, Ö. Yılmaz, E. Çelik, M. Shahdloo, A. G. Huth, and T. Çukur, "Attentional modulation of hierarchical speech representations in a multitalker environment," *bioRxiv*, 2020.

[108] E. U. Saritas, D. Lee, T. Çukur, A. Shankaranarayanan, and D. G. Nishimura, "Hadamard slice encoding for reduced-FOV diffusion-weighted imaging," *Magnetic Resonance in Medicine*, vol. 72, pp. 1277–1290, Nov 2014.

[109] S. G. Kafali, T. Çukur, and E. U. Saritas, "Phase-correcting non-local means filtering for diffusion-weighted imaging of the spinal cord," *Magnetic Resonance in Medicine*, vol. 80, no. 3, pp. 1020–1035, 2018.

[110] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

[111] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134, 2017.

[112] C. Ouyang, J. Schlemper, C. Biffi, G. Seegoolam, J. Caballero, A. N. Price, J. V. Hajnal, and D. Rueckert, "Generalising Deep Learning MRI Reconstruction Across Different Domains," *arXiv preprint*, 2019.

[113] T. Çukur, J. M. Santos, D. G. Nishimura, and J. M. Pauly, "Varying kernel-extent gridding reconstruction for undersampled variable-density spirals," *Magnetic Resonance in Medicine*, vol. 59, pp. 196–201, Jan 2008.

[114] A. Ozaslan, A. Alacaoglu, O. Demirel, T. Çukur, and E. Saritas, "Fully automated gridding reconstruction for non-cartesian x-space magnetic particle imaging," *Physics in Medicine & Biology*, vol. 64, no. 16, p. 165018, 2019.

[115] M. A. Griswold, P. M. Jakob, R. M. Heidemann, M. Nittka, V. Jellus, J. Wang, B. Kiefer, and A. Haase, "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magnetic Resonance in Medicine*, vol. 47, no. 6, pp. 1202–1210, 2002.

[116] B. B. Thukral, "Problems and preferences in pediatric imaging," *The Indian Journal of Radiology & Imaging*, vol. 25, no. 4, pp. 359–364, 2015.

[117] K. Krupa and M. Bekiesińska-Figatowska, "Artifacts in magnetic resonance imaging," *Polish Journal of Radiology*, vol. 80, pp. 93–106, 2015.

[118] C. M. Stonnington, G. Tan, S. Klöppel, C. Chu, B. Draganski, C. R. Jack, K. Chen, J. Ashburner, R. S. J. Frackowiak, and R. S. Frackowiak, "Interpreting scan data acquired from multiple scanners: a study with Alzheimer's disease.," *NeuroImage*, vol. 39, pp. 1180–1185, Feb 2008.

[119] J. E. Iglesias, E. Konukoglu, D. Zikic, B. Glocker, K. Van Leemput, and B. Fischl, "Is synthesizing MRI contrast useful for inter-modality analysis?," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 631–638, 2013.

[120] M. I. Miller, G. E. Christensen, Y. Amit, and U. Grenander, "Mathematical textbook of deformable neuroanatomies," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 90, no. 24, pp. 11944–11948, 1993.

[121] N. Burgos, M. J. Cardoso, K. Thielemans, M. Modat, S. Pedemonte, J. Dickson, A. Barnes, R. Ahmed, C. J. Mahoney, J. M. Schott, J. S. Duncan, D. Atkinson, S. R. Arridge, B. F. Hutton, and S. Ourselin, "Attenuation correction synthesis for hybrid PET-MR scanners: Application to brain studies," *IEEE Transactions on Medical Imaging*, vol. 33, pp. 2332–2341, Dec 2014.

[122] J. Lee, A. Carass, A. Jog, C. Zhao, and J. L. Prince, "Multi-atlas-based CT synthesis from conventional MRI with patch-based refinement for MRI-based radiotherapy planning," *Proceedings of SPIE–the International Society for Optical Engineering*, vol. 10133, p. 101331I, Feb 2017.

[123] A. Jog, A. Carass, S. Roy, D. L. Pham, and J. L. Prince, "MR image synthesis by contrast learning on neighborhood ensembles," *Medical Image Analysis*, vol. 24, no. 1, pp. 63–76, 2015.

[124] A. Jog, S. Roy, A. Carass, and J. L. Prince, "Magnetic resonance image synthesis through patch regression," in *IEEE International Symposium on Biomedical Imaging*, pp. 350–353, 2013.

[125] S. Roy, A. Jog, A. Carass, and J. L. Prince, "Atlas based intensity transformation of brain MR images," in *Multimodal Brain Image Analysis*, pp. 51–62, 2013.

[126] Y. Huang, L. Shao, and A. F. Frangi, "Simultaneous super-resolution and cross-modality synthesis of 3D medical images using weakly-supervised joint convolutional sparse coding," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5787–5796, 2017.

[127] V. Sevetlidis, M. V. Giuffrida, and S. A. Tsaftaris, "Whole image synthesis using a deep encoder-decoder network," in *Simulation and Synthesis in Medical Imaging*, pp. 127–137, 2016.

[128] R. Vemulapalli, H. V. Nguyen, and S. K. Zhou, "Unsupervised cross-modal synthesis of subject-specific scans," in *IEEE International Conference on Computer Vision*, pp. 630–638, 2015.

[129] D. H. Ye, D. Zikic, B. Glocker, A. Criminisi, and E. Konukoglu, "Modality propagation: Coherent synthesis of subject-specific scans with data-driven regularization," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 606–613, 2013.

[130] S. Roy, A. Carass, N. Shiee, D. L. Pham, and J. L. Prince, "MR contrast synthesis for lesion segmentation," in *IEEE International Symposium on Biomedical Imaging*, pp. 932–935, 2010.

[131] N. Cordier, H. Delingette, M. Le, and N. Ayache, "Extended modality propagation: Image synthesis of pathological cases," *IEEE Transactions on Medical Imaging*, vol. 35, no. 12, pp. 2598–2608, 2016.

[132] A. Jog, A. Carass, S. Roy, D. L. Pham, and J. L. Prince, "Random forest regression for magnetic resonance image synthesis," *Medical Image Analysis*, vol. 35, pp. 475–488, 2017.

[133] C. Zhao, A. Carass, J. Lee, Y. He, and J. L. Prince, "Whole brain segmentation and labeling from CT using synthetic MR images," in *Machine Learning in Medical Imaging*, pp. 291–298, Springer, Cham, 2017.

[134] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision – ECCV 2016*, pp. 694–711, 2016.

[135] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 105–114, 2017.

[136] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances in Neural Information Processing Systems*, pp. 658–666, 2016.

[137] J. M. Wolterink, A. M. Dinkla, M. H. F. Savenije, P. R. Seevinck, C. A. T. van den Berg, and I. Isgum, "Deep MR to CT synthesis using unpaired data," in *Simulation and Synthesis in Medical Imaging*, pp. 14–23, 2017.

[138] A. Ben-Cohen, E. Klang, S. P. Raskin, M. M. Amitai, and H. Greenspan, "Virtual PET images from CT data using deep convolutional networks: Initial results," in *Simulation and Synthesis in Medical Imaging*, pp. 49–57, 2017.

[139] F. Mahmood, R. Chen, and N. J. Durr, "Unsupervised reverse aomain adaptation for synthetic medical images via adversarial training," *IEEE Transactions on Medical Imaging*, vol. 37, pp. 2572–2581, Nov 2018.

[140] H. Huang, P. S. Yu, and C. Wang, "An introduction to image synthesis with generative adversarial nets," *arXiv preprint*, Mar 2018.

[141] Y. Hu, E. Gibson, L.-L. Lee, W. Xie, D. C. Barratt, T. Vercauteren, and J. A. Noble, "Freehand ultrasound image simulation with spatially-conditioned generative adversarial networks," in *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*, pp. 105–115, Springer, Cham, 2017.

[142] Y. Hiasa, Y. Otake, M. Takao, T. Matsuoka, K. Takashima, J. L. Prince, N. Sugano, and Y. Sato, "Cross-modality image synthesis from unpaired data using CycleGAN: Effects of gradient consistency loss and training data size," in *Simulation and Synthesis in Medical Imaging*, pp. 31–41, Mar 2018.

[143] J. T. Guibas, T. S. Virdi, and P. S. Li, "Synthetic medical images from dual generative adversarial networks," *arXiv preprint*, Sep 2017.

[144] P. Costa, A. Galdran, M. I. Meyer, M. Niemeijer, M. Abramoff, A. M. Mendonca, and A. Campilho, "End-to-end adversarial retinal image synthesis," *IEEE Transactions on Medical Imaging*, vol. 37, pp. 781–791, Mar 2018.

[145] A. Chartsias, T. Joyce, R. Dharmakumar, and S. A. Tsaftaris, "Adversarial image synthesis for unpaired multi-modal cardiac data," in *Simulation and Synthesis in Medical Imaging*, pp. 3–13, 2017.

[146] F. Calimeri, A. Marzullo, C. Stamile, and G. Terracina, "Biomedical data augmentation using generative adversarial neural networks," in *Artificial Neural Networks and Machine Learning*, pp. 626–634, 2017.

[147] J. M. Wolterink, A. M. Dinkla, M. H. F. Savenije, P. R. Seevinck, and C. A. T. Van Den Berg, "MR-to-CT synthesis using cycle-consistent generative adversarial networks," in *MED-NIPS*, 2017.

[148] Y. Huo, Z. Xu, S. Bao, A. Assad, R. G. Abramson, and B. A. Landman, "Adversarial synthesis learning enables segmentation without target modality ground truth," in *IEEE 15th International Symposium on Biomedical Imaging*, pp. 1217–1220, Dec 2018.

[149] L. Bi, J. Kim, A. Kumar, D. Feng, and M. Fulham, "Synthesis of positron emission tomography (PET) images via multi-channel generative adversarial networks (GANs)," in *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*, pp. 43–51, 2017.

[150] H. Zhao, H. Li, S. Maurer-Stroh, and L. Cheng, "Synthesizing retinal and neuronal images with generative adversarial nets," *Medical Image Analysis*, vol. 49, pp. 14–26, Oct 2018.

[151] D. Nie, R. Trullo, C. Petitjean, S. Ruan, and D. Shen, "Medical image synthesis with context-aware generative adversarial networks," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 417–425, 2017.

[152] O. Shitrit and T. Riklin Raviv, "Accelerated magnetic resonance imaging by adversarial neural network," in *Cardoso M. et al. (eds) Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support.*, vol. 10553, pp. 30–38, 2017.

[153] Y. Wang, B. Yu, L. Wang, C. Zu, D. S. Lalush, W. Lin, X. Wu, J. Zhou, D. Shen, and L. Zhou, "3D conditional generative adversarial networks for high-quality PET image estimation at low dose," *NeuroImage*, vol. 174, pp. 550–562, Jul 2018.

[154] D. Nie, R. Trullo, J. Lian, L. Wang, C. Petitjean, S. Ruan, Q. Wang, and D. Shen, "Medical image synthesis with deep convolutional adversarial networks," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 12, pp. 2720 – 2730, 2018.

[155] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017.

[156] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *IEEE International Conference on Computer Vision*, pp. 2813–2821, 2017.

[157] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.

[158] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2Photo: Internet image montage," *ACM Transactions on Graphics*, vol. 28, no. 5, pp. 124:1–124:10, 2009.

[159] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint*, 2014.

[160] M. Jenkinson, P. Bannister, M. Brady, and S. Smith, "Improved optimization for the robust and accurate linear registration and motion correction of brain images," *NeuroImage*, vol. 17, no. 2, pp. 825–841, 2002.

[161] M. Jenkinson and S. Smith, "A global optimisation method for robust affine registration of brain images," *Medical Image Analysis*, vol. 5, no. 2, pp. 143–156, 2001.

[162] P. Murugan, "Hyperparameters optimization in deep convolutional neural network / Bayesian approach with Gaussian process prior," *arXiv preprint*, Dec 2017.

[163] T. Hinz, N. Navarro-Guerrero, S. Magg, and S. Wermter, "Speeding up the hyperparameter optimization of deep convolutional neural networks," *International Journal of Computational Intelligence and Applications*, vol. 17, p. 1850008, Jun 2018.

[164] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint*, 2016.

[165] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, pp. 600–612, Apr 2004.

[166] T. H. Kim and J. P. Haldar, "The Fourier radial error spectrum plot: A more nuanced quantitative evaluation of image reconstruction quality," in *IEEE 15th International Symposium on Biomedical Imaging*, pp. 61–64, IEEE, Apr 2018.

[167] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," 2017.

[168] F. Chollet and Others, "Keras." https://github.com/keras-team/keras, 2015.

[169] Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv preprint*, 2016.

[170] M. A. Bernstein, K. F. King, and X. J. Zhou, *Handbook of MRI pulse sequences*. Academic Press, 2004.

[171] O. B. Demirel, T. Kilic, T. Çukur, and E. U. Saritas, "Anatomical measurements correlate with individual magnetostimulation thresholds for kHz-range homogeneous magnetic fields," *Medical Physics*, vol. 47, pp. 1836–1844, Apr 2020.

[172] T. Çukur and D. G. Nishimura, "Multiple repetition time balanced steady-state free precession imaging," *Magnetic Resonance in Medicine*, vol. 62, pp. 193–204, Jul 2009.

[173] T. Çukur, J. H. Lee, N. K. Bangerter, B. A. Hargreaves, and D. G. Nishimura, "Non-contrast-enhanced flow-independent peripheral MR angiography with balanced SSFP," *Magnetic Resonance in Medicine*, vol. 61, pp. 1533–1539, Jun 2009.

[174] T. Çukur and D. G. Nishimura, "Fat-water separation with alternating repetition time balanced SSFP," *Magnetic Resonance in Medicine*, vol. 60, pp. 479–484, Aug 2008.

[175] T. Çukur, N. K. Bangerter, and D. G. Nishimura, "Enhanced spectral shaping in steady-state free precession imaging," *Magnetic Resonance in Medicine*, vol. 58, pp. 1216–1223, Dec 2007.

[176] L. Xiang, Q. Wang, D. Nie, L. Zhang, X. Jin, Y. Qiao, and D. Shen, "Deep embedding convolutional neural network for synthesizing CT image from T1-Weighted MR image," *Medical Image Analysis*, vol. 47, pp. 31–44, Jul 2018.

[177] D. Jiang, W. Dou, L. Vosters, X. Xu, Y. Sun, and T. Tan, "Denoising of 3D magnetic resonance images with multi-channel residual learning of convolutional neural network," *Japanese Journal of Radiology*, vol. 36, pp. 566–574, Dec 2018.

[178] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama, "GAN-based synthetic brain MR image generation," in *IEEE 15th International Symposium on Biomedical Imaging*, pp. 734–738, 2018.

[179] X. Yang, R. Kwitt, M. Styner, and M. Niethammer, "Quicksilver: Fast predictive image registration A deep learning approach," *NeuroImage*, vol. 158, pp. 378–396, Sep 2017.

[180] S. Qi, V. D. Calhoun, T. G. van Erp, J. Bustillo, E. Damaraju, J. A. Turner, Y. Du, J. Yang, J. Chen, Q. Yu, D. H. Mathalon, J. M. Ford, J. Voyvodic, B. A. Mueller, A. Belger, S. McEwen, S. G. Potkin, A. Preda, T. Jiang, and J. Sui, "Multimodal fusion with reference: Searching for

joint neuromarkers of working memory deficits in schizophrenia," *IEEE Transactions on Medical Imaging*, vol. 37, no. 1, pp. 93–105, 2018.

[181] B. Gözcü, R. K. Mahabadi, Y.-H. Li, E. Ilcak, T. C. . Ukur, J. Scarlett, and V. Cevher, "Learning-Based Compressive MRI," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1394–1406, 2018.

[182] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[183] H. Salehinejad, S. Valaee, T. Dowdell, E. Colak, and J. Barfett, "Generalization of deep neural networks for chest pathology classification in x-rays using generative adversarial networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 990–994, IEEE ICASSP, 2018.

[184] S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, and T. Çukur, "Synergistic reconstruction and synthesis via generative adversarial networks for accelerated multi-contrast mri," *arXiv preprint*, 2018.

[185] S. U. H. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, B. Tınaz, and T. Çukur, "Joint recovery of variably accelerated multi-contrast MRI acquisitions via generative adversarial networks," in *27th Annual Meeting of ISMRM*, p. 0666, 2019.

[186] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: sensitivity encoding for fast MRI.," *Magnetic Resonance in Medicine*, vol. 42, no. 5, pp. 952–62, 1999.

[187] M. Yurt, S. U. H. Dar, B. Tınaz, M. Özbey, and T. Çukur, "Semi-supervised learning of mutually accelerated multi-contrast mri synthesis without fully-sampled ground-truths," *arXiv preprint*, 2020.

[188] L. Sun, Z. Fan, X. Fu, Y. Huang, X. Ding, and J. Paisley, "A Deep Information Sharing Network for Multi-Contrast Compressed Sensing MRI

Reconstruction," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 6141–6153, 2019.

[189] D. Polak, S. Cauley, B. Bilgic, E. Gong, P. Bachert, E. Adalsteinsson, and K. Setsompop, "Joint multi-contrast variational network reconstruction (jvn) with application to rapid 2d and 3d imaging," *Magnetic resonance in medicine*, vol. 84, no. 3, pp. 1456–1469, 2020.

[190] E. Gong, G. Zaharchuk, and J. Pauly, "Improving the pi+ cs reconstruction for highly undersampled multi-contrast mri using local deep network," in *25th Annual Meeting of the ISMRM*, Lecture Notes in Computer Science, (Honolulu), p. 5663, Springer, 2017.

[191] L. Xiang, Y. Chen, W. Chang, Y. Zhan, W. Lin, Q. Wang, and D. Shen, "Deep-learning-based multi-modal fusion for fast mr reconstruction," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 7, pp. 2105–2114, 2019.

[192] A. Falvo, D. Comminiello, S. Scardapane, G. Finesi, M. Scarpiniti, and A. Uncini, "A multimodal deep network for the reconstruction of t2w mr images," *arXiv preprint*, 2019.

[193] A. Falvo, D. Comminiello, S. Scardapane, M. Scarpiniti, and A. Uncini, "A multimodal dense u-net for accelerating multiple sclerosis mri," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2019.

[194] K. H. Kim, W. Do, and S. Park, "Improving resolution of MR images with an adversarial network incorporating images with different contrast," *Medical Physics*, vol. 45, no. 7, pp. 3120–3131, 2018.

[195] E. Biyik, K. Keskin, S. Uh Dar, A. Koc, and T. Çukur, "Factorized sensitivity estimation for artifact suppression in phase-cycled bSSFP MRI," *NMR in Biomedicine*, vol. 33, p. e4228, Apr 2020.

[196] T. Çukur, M. Lustig, and D. G. Nishimura, "Multiple-profile homogeneous image combination: application to phase-cycled SSFP and multicoil imaging," *Magnetic Resonance in Medicine*, vol. 60, pp. 732–738, Sep 2008.

[197] J. Huang, C. Chen, and L. Axel, "Fast multi-contrast MRI reconstruction," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7510 LNCS, pp. 281–288, 2012.

[198] B. Bilgic, V. K. Goyal, and E. Adalsteinsson, "Multi-contrast reconstruction with Bayesian compressed sensing," *Magnetic Resonance in Medicine*, vol. 66, no. 6, pp. 1601–1615, 2011.

[199] M. J. Ehrhardt and M. M. Betcke, "Multicontrast MRI Reconstruction with Structure-Guided Total Variation," *SIAM Journal on Imaging Sciences*, vol. 9, no. 3, pp. 1084–1106, 2016.

[200] O. Yilmaz, E. U. Saritas, and T. Çukur, "Enhanced phase-sensitive SSFP reconstruction for fat-water separation in phased-array acquisitions," *Journal of Magnetic Resonance Imaging*, vol. 44, no. 1, pp. 148–157, 2016.

[201] T. Çukur, "Spectrally selective imaging with wideband balanced steady-state free precession MRI," *Magnetic Resonance in Medicine*, vol. 75, pp. 1132–1141, Mar 2016.

[202] B. Quist, B. A. Hargreaves, T. Çukur, G. R. Morrell, G. E. Gold, and N. K. Bangerter, "Simultaneous fat suppression and band reduction with large-angle multiple-acquisition balanced steady-state free precession," *Magnetic Resonance in Medicine*, vol. 67, pp. 1004–1012, Apr 2012.

[203] R. R. Ingle, T. Çukur, and D. G. Nishimura, "The central signal singularity phenomenon in balanced SSFP and its application to positive-contrast imaging," *Magnetic Resonance in Medicine*, vol. 67, pp. 1673–1683, Jun 2012.

[204] T. Çukur, M. Yamada, W. R. Overall, P. Yang, and D. G. Nishimura, "Positive contrast with alternating repetition time SSFP (PARTS): a fast imaging technique for SPIO-labeled cells," *Magnetic Resonance in Medicine*, vol. 63, pp. 427–437, Feb 2010.

[205] K. L. Mills and C. K. Tamnes, "Methods and considerations for longitudinal structural brain imaging analysis across development," *Developmental cognitive neuroscience*, vol. 9, pp. 172–190, 2014.

[206] Z. Y. Shan, R. Kwiatek, R. Burnet, P. Del Fante, D. R. Staines, S. M. Marshall-Gradisnik, and L. R. Barnden, "Progressive brain changes in patients with chronic fatigue syndrome: a longitudinal mri study," *Journal of Magnetic Resonance Imaging*, vol. 44, no. 5, pp. 1301–1311, 2016.

[207] B. A. Duffy, W. Zhang, H. Tang, L. Zhao, M. Law, A. W. Toga, and H. Kim, "Retrospective correction of motion artifact affected structural mri images using deep learning of simulated motion," 2018.

[208] M. Aksoy, C. Forman, M. Straka, T. Çukur, J. Hornegger, and R. Bammer, "Hybrid prospective and retrospective head motion correction to mitigate cross-calibration errors," *Magnetic Resonance in Medicine*, vol. 67, pp. 1237–1251, May 2012.