Title
Deep Neural Networks for Cardiovascular Magnetic Resonance Imaging

Author
Ghodrati kouzehkonan, vahid

Publication Date
2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Deep Neural Networks for Cardiovascular Magnetic Resonance Imaging

A dissertation submitted in partial satisfaction of the

requirements for degree Doctor of Philosophy

in Physics and Biology in Medicine

by

Vahid Ghodrati Kouzehkonan

2022

ABSTRACT OF THE DISSERTATION

Deep Neural Networks for Cardiovascular Magnetic Resonance Imaging

by

Vahid Ghodrati Kouzehkonan

Doctor of Philosophy in Physics and Biology in Medicine

University of California, Los Angeles, 2022

Professor J. Paul Finn, Chair

Magnetic Resonance Imaging (MRI) is a powerful diagnostic imaging modality known to provide high soft-tissue contrast and spatial resolution. Much of the versatility of MRI stems from the fact that the signal from different tissue types can be weighted differently through manipulation of the sequence in which radiofrequency (RF) and gradient events are played out during the data acquisition phase. However, data acquisition for most MRI measurements is sequential, limiting its speed and increasing its susceptibility to motion artifacts. This is particularly the case for cardiovascular applications, where cardiac and respiratory motion complicate all aspects of the data acquisition and signal processing pathways. Moreover, following data acquisition and image reconstruction, clinically relevant post-processing may require substantial time and effort, increasing the burden on clinical centers and medical staff. Thus, general algorithms should be customized to accelerate image acquisition, image reconstruction and image post-processing with the goal of expanding the speed, scope and reliability of cardiovascular MRI applications. This

dissertation describes several deep learning-based methods applying tailored image reconstruction, respiratory motion correction, blood vessel segmentation, and instant T1 mapping calculation.

The first application is the acceleration of dynamic cardiac MRI. Modern approaches to speeding MR image acquisition involve the use of significantly under-sampled k-space data (with a proportional reduction in acquisition time), such that the Nyquist limit of traditional signal sampling is violated and the missing k-space data are estimated by other means. The missing data are typically recovered either through incorporating independently acquired surface coil spatial sensitivity maps (parallel acquisition) or through iterative reconstruction via optimized approximations that enforce both sparsity in the sampled domain and consistency with the explicitly acquired data (compressed sensing). Although both parallel imaging and compressed sensing (CS) have proved powerful, they manifest hard limits as the degree of undersampling is increased. Moreover, even with fast modern processors and dedicated reconstruction hardware, image reconstruction times can become prohibitive. Deep learning methods have the potential to address several of the limitations noted for current parallel imaging and CS techniques and to expand the scope of clinical applications.

Our first task was to develop a deep Convolutional Neural Network (CNN) to reconstruct the 2D dynamic cine images from the highly undersampled k-space data, e.g., 8X-10X. In our platform, redundant information in the temporal dimension was used, and the data consistency was imposed in the k-space domain. Indeed, we used CNN only to learn the effective Spatio-temporal regularizer from the historical data in our platform. Learnable parameters (weights and biases) of the neural network were optimized during the off-line training process and tested on the unseen data. Testing inference time was ~40ms per frame, while more than 1s is usually required for conventional parallel imaging and compressed sensing combined reconstruction.

Our next task was to correct respiratory motion artifact that was superimposed on the images acquired during the free-breathing 2D cardiac cine scan. Although segmented (multi-shot) cardiac cine is the gold standard in cardiac imaging, it requires breath-holding through the data acquisition, which may not be feasible in all patients. For this reason, in this dissertation, we sought to find a way to study the performance of the deep neural networks in removing the respiratory artifact from effected 2D cardiac cine images. To achieve that, we trained an adversarial autoencoder network using unpaired training data (healthy volunteers and patients who underwent clinically indicated cardiac MRI examinations). We used a U-net structure for the encoder and decoder parts of the autoencoder. We considered an adversarial objective to regularize the code space of the autoencoder. To ensure that the network reduces the respiratory motion artifact without losing accuracy or introducing new spurious features, we first examined its performance on artificially corrupted data with simulated rigid motion. Then, we demonstrated the feasibility of the proposed approach in vivo by training on actual respiratory motion-corrupted images in an unpaired manner and testing on volunteer and patient data. We showed that it is feasible to correct the respiratory motion-related image artifacts without accessing the paired free of the motion artifact target. Quantitatively in this feasibility study, the mean structural similarity indices (SSIM) for the simulated motion-corrupted images and motion-corrected images were 0.76 and 0.93 (out of 1), respectively. Concerning the image sharpness, the proposed method improved the Tenengrad focus measure of the motion-corrupted images by 12% in the simulation study and 7% in the in-vivo study. Subjective image quality assessments showed that the average overall subjective image quality for the motion-corrupted images, motion-corrected images, and breath-hold images were 2.5, 3.5, and 4.1(out of 5.0), respectively. Statistically, there was a significant difference between the image quality scores of the motion-corrupted and breath-held images ($P<0.05$); however, after

respiratory motion correction, there was no significant difference between the image quality scores of motion-corrected and breath-held images.

Our next further application is joint compensation of the respiratory motion artifact and reconstruction of the high-quality 3D images from the undersampled acquisition in the 3D dynamic cardiac cine MRI. Imaging acceleration and respiratory motion compensation remain two significant challenges in MRI, particularly for cardiothoracic, abdominal, and pelvic MRI applications. This dissertation sought to implement a novel 3D generative adversarial network (GAN)-based technique to jointly reconstruct the image and compensate the respiratory motion artifact of 4D (time-resolved 3D) cardiac MRI. We trained the 3D GAN based on combinations of the pixel-wise content loss, adversarial loss, and a novel data-driven temporal aware loss function. Asides from the image reconstruction, the proposed method also compensates for the respiratory motion of the free-breathing scans. We adopted a novel progressive growing-based strategy to achieve a stable and sample-efficient training process for the proposed 3D GAN. We thoroughly evaluated the performance of the proposed method qualitatively and quantitatively based on the relatively large patient populations (3D cardiac cine data from 42 patients). Our radiological assessments showed that the proposed method achieved significantly better scores in general image quality and image artifacts at 10.7X-15.8X acceleration than the self-gated compressed sensing wavelet (SG CS-WV) approach at 3.5X-7.9X acceleration (4.53±0.540 vs. 3.13±0.681 for general image quality, 4.12±0.429 vs. 2.97±0.434 for image artifacts, $p<0.05$ for both). Radiological evaluations approved that the reconstructed images were free of the spurious anatomical structures and concerning the functional analysis was in good agreement with the conventional SG CS-WV approach. We showed promising results for high-resolution (1mm3)

free-breathing 4D cardiac MR data acquisition with simultaneous respiratory motion compensation and fast reconstruction time which might pave the way for future 4D MR researches.

The fourth application is the fast and accurate calculation of the myocardial T1 and T2 values. Modified Look-Locker inversion recovery (MOLLI) pulse sequence is a widely used MR pulse sequence that allows the measurements and mapping of the myocardial T1 and T2 values. Modeling of the signal evolution of the MOLLI sequence is required to compute the accurate relaxometry parameters. Bloch equation simulation with slice profile correction (BLESSPC) algorithm could consider the non-rectangle 2D RF excitation slice profile effects, B1+ errors, and imperfect inversion and T2 preparation pulses. Nonetheless, BLESSPC is computationally expensive, which limits its applicability in practice. We sought to implement a deep neural network for fast and accurate computation of myocardial T1/T2 relaxometry values by training the neural network on the simulated data computed based on the BLESSPC algorithm. We trained two separate neural networks based on simulated radial T1-T2 values. Trained T1-T2 models were evaluated concerning the stability of the different noise levels and compared against the BLESSPC algorithm. Testing and comparison were performed in different levels, including simulation, phantom, and in vivo data acquired by the MOLLI sequence at 1.5 T and radial T1-T2 sequence at 3 T. Trained models in the phantom studies achieved similar accuracy and precision to the BLESSPC algorithm with respect to T1-T2 estimations for both MOLLI and radial T1-T2 ($P > 0.05$). For in vivo, trained models and BLESSPC produced similar myocardial T1/T2 values for radial T1-T2 at 3 T (T1: 1366 ± 31 ms for both methods, $P > 0.05$; T2: 37.4 ms ± 0.9 ms for both methods, $P > .05$), and similar myocardial T1 values for the MOLLI sequence at 1.5 T (1044 ± 20 ms for both methods, $P > .05$). As was expected, our proposed method can compute the T1/T2 map in less than 1 second (CPU-based) with similar accuracy and precision to the BLESSPC as

the computationally expensive but comprehensive algorithm. The developed model in this dissertation offers a fast and promising approach for accurate computation of myocardium T1/T2 values, replacing BLESSPC for both MOLLI and radial T1-T2 sequences.

The fifth application is the automatic peripheral artery, and vein segmentation in the lower extremities based on ferumoxytol enhanced magnetic resonance angiography (FE-MRA). The post-processing of FE-MRA images mainly includes segmentation of the peripheral vasculature and classification of them into arteries and veins, often performed by an experienced radiologist via visual inspection and manual delineations. Due to the large size of the high resolution, volumetric peripheral MRA, e.g., 560 x 940 x 240, manual annotation is a time-consuming and tedious process. Since manual labeling is a subjective process and depends on physician's experience and knowledge, it can potentially introduce high inter-observer variability. To achieve an accurate and reproducible segmentation of peripheral arteries and veins, we sought to develop an automatic platform in this dissertation. Our proposed platform first segmented the high-quality vascular network from FE-MRA volumetric images and then classified them into arteries and veins. For the segmentation, we used a local attention-gated 3D U-Net and trained that by using a deep supervision mechanism based on a linear combination of the focal Tversky loss and region mutual loss. We performed a region-growing algorithm for the classification, starting from the initial arterial seeds obtained by time-resolved images to separate the arteries from the veins. Quantitatively, our platform achieved a competitive F1 = 0.8087 and Recall = 0.8410 for blood vessel segmentation compared with F1 = (0.7604, 0.7573, 0.7651) and Recall = (0.7791, 0.7570, 0.7774) obtained with Volumetric-Net, DeepVesselNet-FCN, and Uception. The proposed method achieved F1 = (0.8274 / 0.7863) in the calf region-the most challenging region in peripheral arteries and veins segmentation for the artery and vein classification stage. The platform described in this

dissertation is fully automatic without requirements for human interaction and able to extract and label the peripheral vessels from FE-MRA volumes in less than 4 minutes. This method improves upon manual segmentation by radiologists, which routinely takes several hours – an endeavor that is often time- and cost-prohibitive.

The dissertation of Vahid Ghodrati Kouzehkonan is approved.

Holden H. Wu

Kim-Lien Nguyen

Michael Albert Thomas

J. Paul Finn, Committee Chair

University of California, Los Angeles

2022

TABLE OF CONTENT

LIST OF ACRONYMS

GENERAL

T	-	Tesla
Mm	-	millimeter
s	-	second
ms	-	millisecond
ECG	-	Electrocardiogram
HR	-	heart rate
Bpm	-	beat per minute

GENERAL MAGNETIC RESONANCE IMAGING

MRI	-	magnetic resonance imaging
2D/3D	-	two/three dimensional
FOV	-	field of view
TE	-	echo time
TR	-	repetition time
SNR	-	signal to noise ratio
RF	-	radiofrequency
PI	-	parallel imaging
CS	-	compressed sensing
GRAPPA	-	generalized autocalibrating partial parallel acquisition
CE-MRA	-	contrast enhanced magnetic resonance angiography
FE-MRA	-	ferumoxytol enhanced magnetic resonance angiography
SG	-	self gating
CS-WV	-	compressed sensing wavelet

SPECIAL TECHNICAL TERMS

CoV	-	coefficient of variation
ROI	-	regions of interest
CNN	-	convolutional neural network
GAN	-	generative adversarial networks
ANN	-	artificial neural network
SGD	-	stochastic gradient decent
TAV-GAN	-	temporally aware volumetric generative adversarial network
DC	-	data consistency
TA	-	temporally aware
AG	-	attention gate
DS	-	deep supervision

LIST OF FIGURES

from the code space data. b) An adversarial autoencoder (AAE) combines a simple autoencoder with an adversarial regularizer called discriminator to the code space. The discriminator is trained with the goal of accurately differentiating between data generated for the code space of the autoencoder and the data from the external data source Y. The adversarial autoencoder is trained with the goal of generating the code space data that resemble the external source data Y. The end result of the AAE network is that the code space data are driven to represent the external data source as much as possible during the adversarial (and competing) training processes between the encoder part of the autoencoder and the discriminator. In the context of our motion correction work using AAE, the encoder and decoder networks are each a convolutional U-net, the input x of the autoencoder is a free-breathing motion-corrupted image, the code space data is the corresponding motion-corrected image of the same dimensions, and the external data source Y is unpaired standard breath-hold motion-free reference images. The code space is driven by the discriminator network to be motion-corrected images such that they resemble the motion-free images from the external source Y. More details about the structure of our network are included in Figure 4-2. . 39

Figure 4-2. The autoencoder part consists of two convolutional U-net. A U-net consists of two paths: (I) the contracting path, which contains 3 down-sampling stages; (II) the expanding path, which includes 3 up-sampling stages. In order to preserve high-level features, it consists of dense connections from the early stages to the later stages of the network. Each convolution layer used in the U-net consists of trainable convolution filters (stride = 1) followed by rectified linear unit (ReLU) as a non-linear activation function except for the last layer. If ReLU was used in all layers as the non-linear activation function, then the network would only be able to learn to map to positive values. Because the input data sets were normalized to the range of [-1,1], it is important for the last layer's activation function to have the capability to pass negative values in the forward

of a slice from the volumetric images, and the second row shows the difference map between two sequential cardiac phases. As can be seen in (a), it has the lowest noise and flickering artifacts through the cardiac phases among the others. The image in the column (b) has relatively higher noise and flickering artifacts through the cardiac phases than the image in column (a). Based on the calculation of the noise inside a 15×15×15 cubic region from the background, images in the datasets B1 (mean of the standard deviation = 0.076) are 2 times noisier than the images in the datasets A (mean of the standard deviation = 0.038). Column (c) presents image that was profoundly affected by noise. Approximately, the noise level for noisy images in datasets B2 (mean of the standard deviation = 0.304) based on the calculation of the noise inside a 15×15×15 cubic region from the background is, on average, 8 times the images in datasets A. Column (d) shows an image from a CHD patient with breathing irregularities scanned under anesthesia. As shown in column (d), image quality is degraded due to the respiratory motion artifacts. The image in column (e) shows an image from a CHD patient scanned under free-breathing without anesthesia. As shown in column (e), the quality of the image is degraded substantially due to the respiratory artifact and breathing irregularities. 79

Figure 5-6. Qualitative comparison between different image reconstruction methods for a male CHD patient from test dataset B1 (6 y.o. and 18 kg weight) who was scanned under anesthesia. Row (a) shows the reconstruction/respiratory motion correction results and rows (b) and (c) show the zoomed view of the cardiac and liver region. Row (d) shows the temporal difference between 5th and 6th cardiac phases. The 2D GAN image has substantial residual artifacts. The 3D U-Net image is blurrier than the GAN based methods (TAV-GAN, Temporal-GAN, and Volumetric-GAN). As shown in (d), reconstruction results from TAV-GAN and Temporal-GAN have the lowest incoherency and flickering artifacts, which implies that the proposed TA loss can

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to sincerely express my appreciation for my advisors, Dr. Paul Finn and Dr. Peng Hu, and my thesis committee, Dr. Holden Wu, Dr. Kim-Lien Nguyen, Dr. Albert Thomas, for all supports that they provided to me throughout my Ph.D. journey. In particular, I would like to thank Peng Hu for his substantial support, especially in the challenging paper revision process.

I wholeheartedly want to thank Dr. Mark Bydder. He provided substantial help in learning the correct way of doing research. He offered insightful pieces of advice on my research and shared his wisdom on life and work. I was fortunate that I had a chance to work closely with him and solidify my knowledge.

I want to thank Mark, Fadil, and Andres for their generous help in my research. I have learned not only research-related skills but also high-level life-related skills. Without their kind support, I could not finish many projects quickly. I also want to thank Heather for providing permanent supports.

I am pleased to work with all the current and past members and staff of the Magnetic Resonance Research Lab (MRRL): Fadil, Jiaxin, Summer, Shams, Chang, Caroline, Hengjie, Take, Jessica, Nyasha, Andres, Tess, Zhaohuan, and more. I want to thank them for willing to be my volunteers for the research MRI scans. I am proud of being a member of the Physics and Biology in Medicine program family. In particular, I would like to thank Dr. Michael McNitt-Gray, Reth Im, and Alondra Correa Bautista for their permanent supports and guidance throughout my Ph.D. life.

Last but not least, I would like to express my gratitude to my friends Amirhossein, Mohammad, Mark, Fadil, Andres, Racheal, Bao, Jie, Qihui, Asyieh. I am fully recharged whenever I see them and talk to them. Lastly, I would like to million times thank my parents, Ahad and Nasrin, my brother, Saeed, and my sister, Parisa, for always being so supportive and giving me the freedom to pursue my goal.

Vita

Education:

- Ph.D. student, Biomedical Physics, University of California, Los Angeles, CA, 2016–2021
- M.Sc., Biomedical Engineering, Amirkabir University of Technology, Iran, 2012-2015
- B.S., Biomedical Engineering & Electrical Engineering, Amirkabir University of Technology, Iran, 2009-2013

First/Co-First Author Peer-Reviewed Publications:

1. Gao C, **Ghodrati V**, Shih SF, et al. Undersampling artifact reduction for free-breathing 3D stack-of-radial MRI based on a deep adversarial learning network. Magn Reson Med. (under revision)
2. **Ghodrati V**, Rivenson Y, Prosper A, et al. Automatic segmentation of peripheral arteries and veins in ferumoxytol-enhanced MR angiography. Magn Reson Med. 2021; 00: 1-15.
3. **Ghodrati V**, Bydder M, Bedayat A, et al. Temporally aware volumetric generative adversarial network-based MR image reconstruction with simultaneous respiratory motion compensation: Initial feasibility in 3D dynamic cine cardiac MRI. Magn Reson Med. 2021; 86: 2666-2683.
4. Gao Y, **Ghodrati V**, Kalbasi A, et al. (2021), Prediction of soft tissue sarcoma response to radiotherapy using longitudinal diffusion MRI and a deep neural network with generative adversarial network-based data augmentation. Med. Phys., 48: 3262-3372.
5. **Ghodrati V**, Bydder M, Ali F, et al. Retrospective respiratory motion correction in cardiac cine MRI reconstruction using adversarial autoencoder and unsupervised learning. NMR in Biomedicine. 2021; 34:e4433.
6. **Ghodrati V**, Shao J, Bydder M, et al. MR image reconstruction using deep learning: evaluation of network structure and loss functions. Quant Imaging Med Surg 2019;9(9):1516-1527.

Co-Author Peer-Reviewed Publications

7. Ali F, Bydder M, Han H, Wang D, **Ghodrati V**, et al. Slice encoding for the reduction of outflow signal artifacts in cine balanced SSFP imaging. Magn Reson Med. 2021; 86: 2034– 2048.

8. Bydder M, Ali F, **Ghodrati V**, Hu P, Yao J, Ellingson BM. Minimizing echo and repetition times in magnetic resonance imaging using a double half-echo k-space acquisition and low-rank reconstruction. NMR in Biomedicine. 2021; 34:e4458.
9. Shao J, **Ghodrati V**, Nguyen K-L, Hu P. Fast and accurate calculation of myocardial T1 and T2 values using deep learning Bloch equation simulations (DeepBLESS). Magn Reson Med. 2020; 84: 2831– 2845.
10. Bydder M, **Ghodrati V**, Gao Y, Robson MD, Yang Y, Hu P. Constraints in estimating the proton density fat fraction. Magn Reson Imaging (2020) 66:1–8.
11. Zhou Z, Han F, **Ghodrati V**, Gao Y, Yin W, Yang Y, Hu P. (2019), Parallel imaging and convolutional neural network combined fast MR image reconstruction: Applications in low-latency accelerated real-time imaging. Med. Phys., 46: 3399-3413.
12. Nguyen K-L, Shao J, **Ghodrati V**, et al. Ferumoxytol-enhanced CMR for vasodilator stress testing: a feasibility study. JACC Cardiovasc Imaging. 2019;12(8):1582-1584.

Chapter 1 Introduction

1.1 Outline

Magnetic Resonance Imaging is a non-invasive imaging tool that can provide the highest soft-tissue contrast among the existing imaging modalities. Sophisticated manipulation of the intrinsic and extrinsic contrast mechanisms could enable us to increase the sensitivity of the MR signal to a variety of physiological behavior and achieve various tissue contrasts. Therefore, MRI is a versatile choice for clinical use to detect pathologies, quantify biological parameters, and reveal functional changes.

MRI is inherently slow despite its vast potential, so it requires a relatively long scan time and could be susceptible to the motion, e.g., bulk motion or cardiorespiratory motion. In addition, acquired images cannot be directly used for diagnosis purposes and required post-processing steps in some MRI applications. Such post-processing steps, which in some applications are usually performed manually by the radiologists- add more burden to the clinical settings- or needed time-consuming computations by the radiologists add more burden to the clinical settings. For example, contrast-enhanced magnetic resonance angiography (CE-MRA) can be acquired from the patients in a reasonably short time, but segmenting the blood vessels, e.g., arteries and veins, and assessing the amount of their blockage is taking several hours. Thus, imaging acceleration techniques and faster post-processing tools are highly in demand. This dissertation sought to develop tools to reduce the scan time and respiratory motion artifact in the 2D/3D cardiac imaging and implement post-processing methods for the instant and accurate T1 and T2 computation and for the peripheral artery and vein segmentation from CE-MRA.

In a broad sense, imaging duration depends on two main factors: spatial/temporal resolution and signal-to-noise ratio (SNR). For example, extended Fourier encoding steps are required to achieve a higher spatial/temporal resolution image which elongates the scan duration. As another example, to achieve a higher SNR image, multiple averaging as a traditional approach is needed, which again extends the scan duration. One potential remedy to reduce the scan time is to acquire fewer data points instead of filling the whole k-space. However, such an incomplete k-space acquisition would lead to the aliasing artifacts. Two general approaches have been introduced to recover the high-quality images from the incomplete k-space measurements: Parallel Imaging (PI)[1-4], which relies on using channel information to turn the underdetermined set of equations into an overdetermined problem, or Compressed Sensing (CS)[5,6], which takes advantage of the incoherent measurements along with appropriate regularizers to solve the underdetermined MR reconstruction problem.

Although CS can achieve a higher acceleration factor, in other words, it can reconstruct the significantly more undersampled k-space than PI; it has its limitations. For example, regularizer terms in the CS framework have to be decided before reconstruction, or, more importantly, it requires iterative computation to recover the high-quality images through solving the optimization problem. Recently deep neural networks showed promising results in medical imaging and, in particular, in image reconstruction and artifact removal tasks. Deep neural networks consist of several layers in which each layer contains learnable weights and non-linear activation functions. It requires a training process in which the stochastic gradient descent (SGD) based algorithm was usually applied through the chain rules on the objective function to adjust the weights and all trainable parameters of the network. Once the training process is completed, it can apply to the unseen data and fastly produce the results. The capability of the neural networks to learn the

compelling features from the historical data and their fast inference time makes them suitable in MRI applications. For example, fixed regularizer terms in the CS reconstruction can be replaced with neural networks to learn a better regularizer. As another example, neural networks can be trained to approximate the functions, particularly those that require heavy computation. For instance, neural networks can be trained to rapidly segment the blood vessels from MRA images which is one of the most labor-intensive tasks in the post-processing stage of the MRI.

This dissertation describes several application-tailored based on the deep neural networks that aim to accelerate the 2D/3D cardiac MRI and offers retrospective-based methods for the respiratory motion correction. Moreover, this dissertation aims to achieve instant and accurate T1/T2 calculations based on the MOLLI sequence and automate the peripheral artery and vein segmentation process in FE-MRA in the post-processing stages.

1.2 Organization of the thesis

Chapter 2 Background: This chapter first begins with a brief introduction of the concepts behind nuclear magnetic resonance and classical signal processing-based approaches for imaging acceleration and motion artifact correction. Then, we tried to smoothly move from the classical approaches to the deep learning-based approaches. In this chapter, we tried to explain the essential deep learning-based concepts, in particular, deep neural networks, in simple words and their potential in addressing the interesting MRI problems with a primary focus on the imaging acceleration and motion compensation, the post-processing type regression problems, and the image segmentation problems. It is not possible to cover all aspects of the deep neural networks in this introductory chapter, and our main focus will be on the relevant aspects of them to our applications.

Chapter 3 Deep learning based Dynamic Cardiac Magnetic Resonance Image Reconstruction Pipeline: This chapter describes a deep learning pipeline for fast reconstruction of the highly undersampled 2D dynamic cardiac magnetic resonance images. Deep convolutional neural networks are used in this work as a modeling tool to learn an effective Spatio-temporal regularizer. Also, to keep the data consistency, we used a hard replacement scheme to use the already acquired lines to force the data fidelity term in the k-space domain. Quantitative comparisons with the CS-based reconstruction were performed on the patients' data in a retrospective manner. This version of our platform is the extended version of our previously published journal articles[7,8]. The new version described in this chapter has been filed as a patent application.

Chapter 4 Retrospective Respiratory Motion Correction in Cardiac Cine MRI Reconstruction: This chapter demonstrates a retrospective deep neural network-based method to reduce the respiratory motion artifact from the free-breathing 2D segmented cine images. Neural networks in the supervised settings were usually required input-target pairs for the training. Since access to the paired input-target images in the context of the non-rigid motion artifacts is challenging or almost impossible, we considered the deep learning-based approach that can be trainable without needing the paired data. We implemented a deep adversarial autoencoder to remove the respiratory motion artifact in the image domain. The only requirement that the implemented adversarial autoencoder has to meet is the availability of two sets of data: 1) free of the respiratory motion artifact cardiac images and 2) respiratory motion artifacted images. We first examined the network on the simulated data to ensure that the proposed approach is functioning correctly. Then we thoroughly evaluated based on the real data acquired from volunteers and patients. This work has been published as a journal article[9].

Chapter 5 Temporally Aware Volumetric GAN-based 4DMR Image Reconstruction and Respiratory Motion Compensation: This chapter introduces a novel neural network-based platform for simultaneously image reconstruction and respiratory motion compensation of 4D cardiac MRI. Temporally Aware Volumetric GAN technique (TAV-GAN) incorporates a novel temporally aware objective function as an extra regularizer in addition to adversarial loss, L1 and SSIM loss functions to reduce flickering artifacts through the cardiac phases with no explicit need to use the multiple cardiac phases as the inputs for the network. We also described the well-known challenges of training GANs for high-dimensional images. We addressed such challenges by adopting an effective progressive training strategy based on starting the training from the low-resolution volumetric images and gradually increasing the resolution to reach the original volumetric image size. The proposed method was thoroughly evaluated qualitatively and quantitatively based on 3D cardiac cine data from 42 patients. This work has been published as a journal article[10].

Chapter 6 Fast and accurate quantification of myocardial T1 and T2 values using Deep learning Bloch Equation Simulations (DeepBLESS): This chapter demonstrates a deep learning-based methodology to achieve a fast and accurate computation of the cardiac T1 and T2 relaxometry parameters. To prepare the data sets, we took a different approach, and instead of using the real datasets commonly used to train the network, we used simulated datasets for training the neural network. Such a simulated dataset enabled us to produce a large dataset and effectively train the network. Also, we considered the advanced T1 and T2 calculation methods to consider the different factors that can potentially affect the T1 and T2 values. Comprehensive evaluations ranging from the simulations to the in vivo validations were considered to examine the proposed approach. The results of this chapter have been published as a journal article[11].

Chapter 7 Automatic Peripheral Artery and Vein Segmentation: This chapter describes an automated platform for segmentation of the peripheral arteries and veins in the lower extremities based on Ferumoxytol-enhanced MR Angiography (FE-MRA). We demonstrated a deep learning-based pipeline to extract peripheral vasculature from high spatial resolution FE-MRA datasets and label them as arteries and veins via exploiting the time-resolved high temporal resolution volumetric images. For extraction of the blood vessels, an attention-gated 3D U-Net is used and trained based on advanced loss functions and a deep supervision mechanism. We tried to study and analyze the role of the different components of the neural network and their effect on the segmentation results. Also, we quantitatively evaluated the proposed approach thoroughly and compared it against the state-of-the-art approaches. This work has been published as a journal article[12].

Chapter 8 Conclusion: The deep neural network-based applications presented in this dissertation are summarized in this chapter, and potential future directions are briefly explored.

Chapter 2 Background

This chapter introduces background about MRI, T1/T2 mapping, inverse problems in magnetic resonance image reconstruction, deep neural networks, and their role in solving the inverse problems. In addition, brief background information about and blood vessel segmentation are provided. It is important to note that this chapter is not meant to be a comprehensive summary of the topics but to briefly familiarize the readers with the discussed materials in the subsequent chapters.

2.1 History of Magnetic Resonance Imaging

Without a doubt, the development of the MRI is one of the most successful and fantastic events in the history of medical imaging. MRI was built based on the principles of the nuclear magnetic resonance (NMR) phenomenon, which Felix Bloch[13] and Edward M. Purcell[14] have discovered, and they shared the 1952 Nobel Prize in Physics independently in 1946. Although the NMR phenomenon was discovered in 1946, the first imaging experiments were conducted in the 1970s by Lauterbur[15] and Damadian[16]. Damadian in 1971 showed that NMR could contrast the behavior of water in benign and malignant tissues. At the same time, Lauterbur imaged a cross-sectional part of the two water tubes. The researchers have rapidly realized the importance of the mentioned inventions and developed the first human whole-body NMR imaging system in 1977. Lauterbur ultimately was awarded the Nobel Prize in physiology and medicine for expanding the spatial encoding gradients and Mansfield[17-19] for his mathematical description of MR imaging physics. As noted above, MRI researches gained several Nobel Prizes, which shows its importance and complexities. In this section, we reviewed the critical MR physics concepts from the

macroscopic point of view, and we used the system model with Fourier encodings to describe the image generation. For more detail, we encourage the reader to refer to these papers[20,21].

2.2 NMR Physics

Nuclei with an odd mass number, such as the hydrogen atom's nucleus, possess an angular momentum, i.e., spin. At the thermal equilibrium state, spins are distributed randomly, which results in zero net magnetization. Applying an external magnetic field $\boldsymbol{B_0}$ polarized the spins and forced them to precessed around the specific directions. For example, for a spin-1/2 system, e.g., hydrogen (^{1}H) spins would take two directions parallel and anti-parallel to the applied magnetic field. Magnetization vector $\boldsymbol{M} = (M_x, M_y, M_z)$ is commonly used to describe the behavior of the spin system at the macroscopic level. In the absence of the external magnetic field $\boldsymbol{B_0}$, magnetization vector $\boldsymbol{M} = \boldsymbol{0}$. In the presence of $\boldsymbol{B_0}$, the magnitude of $\boldsymbol{M}$ will be directly proportional to the magnitude of $\boldsymbol{B_0}$ and the total number of spins. Also, in the presence of $\boldsymbol{B_0}$, the precession frequency of the spins, i.e., Larmor frequency ω_0, is proportional to $\boldsymbol{B_0}$:

$$\omega_0 = \gamma \boldsymbol{B_0} \tag{2-1}$$

Where γ is the atom-specific constant and called gyromagnetic ratio. For the sake of clarity, we should state that in the context of MRI, $\boldsymbol{B_0}$ represents the strong static magnetic field, e.g., 1.5T. MRI machine also has a time-varying RF filed, which is commonly denoted by $B1(t)$. In contrast to strong $\boldsymbol{B_0}$, which is always on, $B1(t)$ can be easily turned on or off, and its magnitude is significantly lower than the static magnetic field. Suppose a spin system is exposed to both $\boldsymbol{B_0}$ and $B1(t)$ since the magnitude of $B1(t)$ is negligible compared to $\boldsymbol{B_0}$, so it seems that applying an extra $B1(t)$ has no effect. However, it is not the whole story, and if $B1(t)$ with a Larmor frequency is applied perpendicular to $\boldsymbol{B_0}$, it can create a resonance condition in

which $\boldsymbol{M_0}$ can be tipped from the direction of the applied $\boldsymbol{B_0}$, say $\boldsymbol{z}$-direction, to the direction of the applied RF field say transversal direction. Once $B1(t)$ is removed, the spin system will relax to its initial state, i.e., alignment with $\boldsymbol{B_0}$, and throughout the relaxation, it will release a radio-frequency signal which an RF receiver coil can detect. The Bloch equations explicitly describe the time-dependent behavior of the magnetization $\boldsymbol{M} = (M_x, M_y, M_z)$ in the presence of the magnetic field $\boldsymbol{B}$. Equation (2-2) shows the simplified Bloch equation:

$$\frac{d\boldsymbol{M}}{dt} = \gamma \boldsymbol{M} \times \boldsymbol{B} - \frac{M_x \boldsymbol{i} + M_y \boldsymbol{j}}{T_2} - \frac{(M_z - M_z^0)\boldsymbol{k}}{T_1} \tag{2-2}$$

Where M_z^0 is the initial magnetization in the presence of $\boldsymbol{B_0}$ only, T1 and T2 are the relaxation parameters and controls the recovery time of the longitudinal component of $\boldsymbol{M}$ (M_z) decaying time of the transverse component of $\boldsymbol{M}$ ($M_x \boldsymbol{i} + M_y \boldsymbol{j}$). Since the relaxation times are sample-specific and related to the tissue characteristics so they can be used to create the contrast between different tissues.

2.3 Spatial Localization

As illustrated in subsection 2.2, once the applied $B1(t)$ is turned off, the bulk magnetization will relax to the initial magnetization state, which releases the RF signal, and the RF receiver coils can record that signal. This RF signal contains the information from the whole sample, and in order to localize such information, we need an extra 3D spatially variant longitudinal magnetic field known as the gradient field $\boldsymbol{G} = (G_x, G_y, G_z)$. The role of the gradient field is to make the effective magnetic field, and therefore the relative precession frequency of the magnetization, a linear function of spatial coordinates along the respective axes. In other words, it encodes the spatial information in the received RF signal. The received RF signal or, in a better term, MR signal from

a volume of interest $m(\vec{x})$ in the presence of the spatially-dependent field can be described as Equation (2-3):

$$s(t) = \int m(\vec{x}) \exp\left(-i\omega(\vec{x}, t)\right) d\vec{x} \quad (2\text{-}3)$$

Where spatially varying phase $\omega(\vec{x}, t)$ can be calculated as follows:

$$\omega(\vec{x}, t) = \int_{t1}^{t2} \gamma G(\tau).\vec{x} d\tau = 2\pi(k_x(t)x + k_y(t)y + k_z(t)z) \quad (2\text{-}4)$$

Where $k_x(t)$, $k_y(t)$, and $k_z(t)$ are the time integrals of the orthogonal components of the gradient waveform and formulated as the following equations:

$$k_x(t) = \frac{\gamma}{2\pi} \int_{t1}^{t2} G_x(\tau) d\tau \quad (2\text{-}5)$$

$$k_y(t) = \frac{\gamma}{2\pi} \int_{t1}^{t2} G_y(\tau) d\tau \quad (2\text{-}6)$$

$$k_z(t) = \frac{\gamma}{2\pi} \int_{t1}^{t2} G_z(\tau) d\tau \quad (2\text{-}7)$$

Where $G_x(t)$, $G_y(t)$, and $G_z(t)$ are the time-varying gradient field and t1 and t2 represents the start and end time points of the applied gradient field. By substituting $\vec{\boldsymbol{k}}(t) = (\boldsymbol{k}_x(t), \boldsymbol{k}_y(t), \boldsymbol{k}_z(t))$ in Equations (2-3) and (2-4), the acquired signal as a function of time can be formulated as:

$$s(t) = \int m(\vec{x}) \exp\left(-i2\pi\vec{\boldsymbol{k}}(t).\vec{\boldsymbol{x}}\right) d\vec{x} \quad (2\text{-}8)$$

The most crucial fact summarized by Equation (2-8) is that the acquisition signal is the Fourier transform of the target volume of interest $m(\vec{x})$. It means that the MR acquisition process can be seen as the sampling in the spatial-frequency space with the trajectory controlled by $\vec{\boldsymbol{k}}(t)$, the time integral ofthe gradient. The spatial-frequency domain so defined is called k-

space in the context of MRI. Using Equations (2-5), (2-6), and (2-7), the excited spins can be localized to any arbitrary point in 3D space by applying appropriate gradients.

2.4 Cartesian Sampling and Image Reconstruction

Cartesian sampling is the most common way to sample the k-space. Figure 2-1 shows the Cartesian sampling for 2D and 3D imaging.

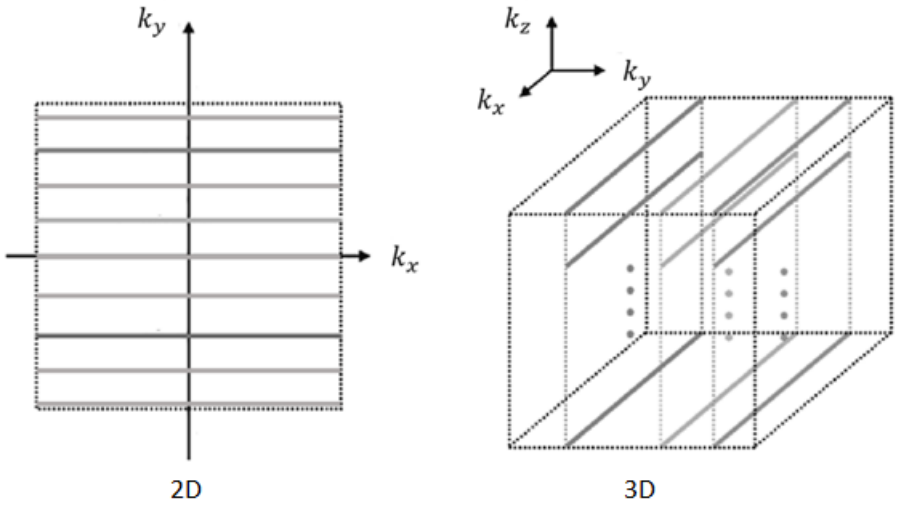

Figure 2-1. 2D and 3D Cartesian sampling trajectories

Fast Fourier Transform (FFT) algorithm is the most efficient digital implementation for transformation between k-space and image space, which are inverse transforms of each other. For Cartesian sampling, k-space lines have to be equally spaced, and samples have to fall onto a Cartesian grid. Cartesian sampling is the most robust sampling strategy to deal with the several sources of system imperfections, e.g., off-resonance and eddy currents[22,23]. However, it is sensitive

to movement of the target volume of the interest during the acquisition process, which in many instances can last for several minutes.

2.5 Undersampling in Cartesian k-space

In the context of the Cartesian sampling, Image reconstruction can be expressed as a set of linear equations as the following:

$$y = \boldsymbol{F}x \tag{2-9}$$

$\boldsymbol{F}$ is the Fourier operation, x is the underlying image, and y is the measurement lines (k-space lines). If we access the fully sampled k-space, the desired x can be found by multiplying the inverse Fourier operation $\boldsymbol{F^{-1}}$ to both sides of Equation (2-9). If the measurement lines are free of motion artifact, it will result in a clean and artifact-free image. To accelerate the imaging process, one potential approach is acquiring fewer lines and filling the k-space partially. So, in the setting of Cartesian undersampling, Equation (2-9) will change to:

$$y = U\boldsymbol{F}x \tag{2-10}$$

Where U is the undersampling binary mask which includes the index of the sampled k-space lines. Although an undersampling scheme will decrease the required acquisition time, it may induce aliasing artifact in the image. From the signal and system point of view, the image reconstruction problem from the fully or undersampled k-space can be viewed as an inverse problem in which the input x is imported to a system described by the forward operation and the output is the measurements y. The forward operation is $U\boldsymbol{F}$ and $\boldsymbol{F}$ for the undersampled and fully sampled acquisition, respectively, in the image reconstruction. It is important to note that the forward operation is wholly known for the pure image reconstruction problem. In the MR reconstruction

problem, we are interested in outputting an image y of the object x, in Equation (2-10), free of any aliasing artifact. We will review two classical techniques in the broad sense in the next section.

2.6 Conventional Reconstruction Techniques

As noted in the previous section, the MR image reconstruction problem can be formulated as an inverse problem $y = \boldsymbol{F}x$, where x is the complex-valued image series formatted as an $N = N_x \times N_y \times N_t$ column vector, $\boldsymbol{F}$ is the Fourier encoding matrix, and y is the measurement k-space vector. It is worth noting that the measurements in this section, for the sake of generality, are assumed as the time series of 2D acquisitions (2D + t). By applying undersampling schemes, the inverse problem formulation would change to $y = A_u x$, where forward operation A_u is a composite operator with size $M \times N$ includes sensitivity maps S, Fourier encoding matrix $\boldsymbol{F}$, and binary undersampling mask U. In MR undersampled inverse problems, usually, the number of measurements M is significantly less than the number of unknown N ($M << N$); thus the direct solution is not possible because of the underdetermined nature of the problem. So, in order to solve this problem, two general approaches have been introduced: Parallel Imaging (PI)[1-4], which relies on using channel information to turn the underdetermined set of equations into an overdetermined problem, or Compressed Sensing (CS)[5,6], which takes advantage of the compact representation of the data in some transform domains to solve the underdetermined MR reconstruction problem.

2.7 Motion Artifact

Up to this point, we only considered that motion artifacts did not contaminate our measurements. Here is the proper place to discuss the effect of the movement in the imaging target of interest on the acquired k-space lines in a simplified setting. As mentioned earlier in this chapter, MR scanners excite the spins of an object and encode the received signal to Fourier coefficients

along a pre-defined path from the gradient pulse's shape. We formulated the signal coming from a moving object in MRI and its effects on the acquired signal in Cartesian schemes. For simplicity, we formulated the rigid body motion in 2D Cartesian acquisition.

Let us assume $\widehat{M}_{\theta,t}$ represents the object motion in matrix format, which includes the translation and rotation. Also, $\boldsymbol{u}$ is the unknown sharp object, $\overline{\boldsymbol{F}} = \boldsymbol{AF}$ is the product of Fourier matrix and affected acquired lines, i.e., phase encodes lines, and e is the additive noise. For the sake of clarity, affected acquired lines are the lines affected by the motion artifact and contain incorrect phase information. Therefore, the acquired signal in the k-space formulation can be described as Equation (2-11):

$$y = \sum_0^T \overline{\boldsymbol{F}}\, \widehat{M}_{\theta,t} dt \boldsymbol{u} + e \tag{2-11}$$

$\widehat{M}_{\theta,t}$ is the operational matrix in the image domain, representing the rigid motion information as a function of time (t) and the object's rotation (θ). Since translation and rotation of the object in the image domain are equal to the phase shift and rotation in the frequency domain, we can translate the mentioned matrix to the frequency domain and reach Equation (2-12):

$$y = \sum_0^T \widehat{K}_{\theta,t}\, dt \overline{\boldsymbol{F}} \boldsymbol{u} + e \tag{2-12}$$

If we compare the motion correction and image reconstruction problems in MRI from the general point of view, we can realize that both problems are types of inverse problems. In the motion correction inverse problem, the forward operation is unknown, while the forward operation is known in the pure MR image reconstruction problem.

2.8 **Artificial Neural Networks**

In this section, we introduced the main components of neural networks. It is important to note that we did not provide a comprehensive review of the neural networks because of the limited space. The interested readers are referred to these publications[24,25]. Neural Networks are the flexible and powerful class of nonlinear function approximations. Figure 2-2 shows a simple feedforward fully connected neural network with three layers. Circles in each layer represent nodes in the network, and each line between nodes represents connections between the nodes. When the data is fed to the input layer, it is sent through the connections to the next layer, where some computations were performed on the data and sent to the next layer. This process continues until the last layer produces the final output. Because of the forward movement of the data through the layers, such neural networks are also called feedforward neural networks. In other words, the output of the layers in the feedforward neural networks do not influence its input. The feedforward pass of the neural networks can be formulated mathematically.

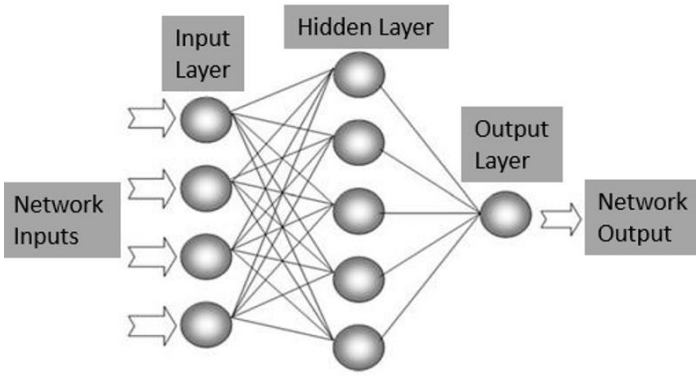

Figure 2-2. A three-layer feedforward fully connected neural network

Since the feedforward neural network is the sequential chain of the layers, it is sufficient to only formulate one arbitrary layer's output for the sake of the mathematical description. Also, for the sake of simplicity, a simple, fully connected neural network is considered. Let us assume that the input signal is a column vector with n components $\boldsymbol{s} = [s_1, s_2, .., s_n]$, $\boldsymbol{W}_{\mathrm{n\times m}}$ is the weight matrix, and $\boldsymbol{b}$ is a row vector with m components. n is the number of the nodes in the previous layer that the connections come from, and m represents the number of the nodes in the current layer. Also, b is the bias term that is added to each node. The output of the nodes in the current layer can be presented as a row vector $\boldsymbol{h}$ with m components:

$$\boldsymbol{h} = f(\boldsymbol{sW}),\ f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \tag{2-13}$$

Where $f(.)$ is the non-linear activation function. In Equation (2-13), one of the most influential and standard activation functions called rectified linear unit (ReLU)[26] is used. It is worth noting that there are several other activation functions such as Tanh and sigmoid. Of the sigmoid functions, the logistic function is the most widely used in machine learning today. To determine the proper weights and bias terms in the neural networks, a training process is required. In the training process, the network takes data as the input and target, and in the forward pass, the network's output is calculated and compared with the ground truth target data. Differences between the output and the target constitute errors. To calculate the error term, comparisons are usually performed based on quantitative metrics such as L1 and L2 norms, Dice, cross-entropy, adversarial loss, and others. We should note that selection of the metrics or objective functions is directly related to the task. After calculating the error following the first feedforward pass, the process of iterative optimization begins. The first backward pass is started, and the process of stochastic gradient descent[27] (SGD) is applied. Gradient descent is the process whereby changes in weights

and biases are implemented in the way that decreases the loss term most rapidly. Gradient descent is carried out with respect to the trainable parameters such as bias and weights to update these parameters and minimize the loss terms. It is worth noting that the backpropagation algorithm is commonly used to calculate the gradient descent efficiently. As the last point in this section, we need to clarify that the term "deep," usually seen in current research, means that the network has many hidden layers, beyond that which connects to the input layer. In recent years, impressive progress in neural networks, such as introducing residual connections[28] and batch normalization[29], facilitates the training of deep neural networks.

2.9 Convolutional Neural Networks

The deep convolutional neural network, a subclass of the deep artificial neural networks, has been proven successful in many applications, particularly in tasks dealing with multi-dimensional data such as images and tensors. Figure 2-3 shows a simple CNN which is designed to perform a classification task. As shown in Figure 2-3, similar to the fully connected ANNs, CNN also contains several layers and trainable weights, i.e., convolutional kernels and non-linear activation functions.

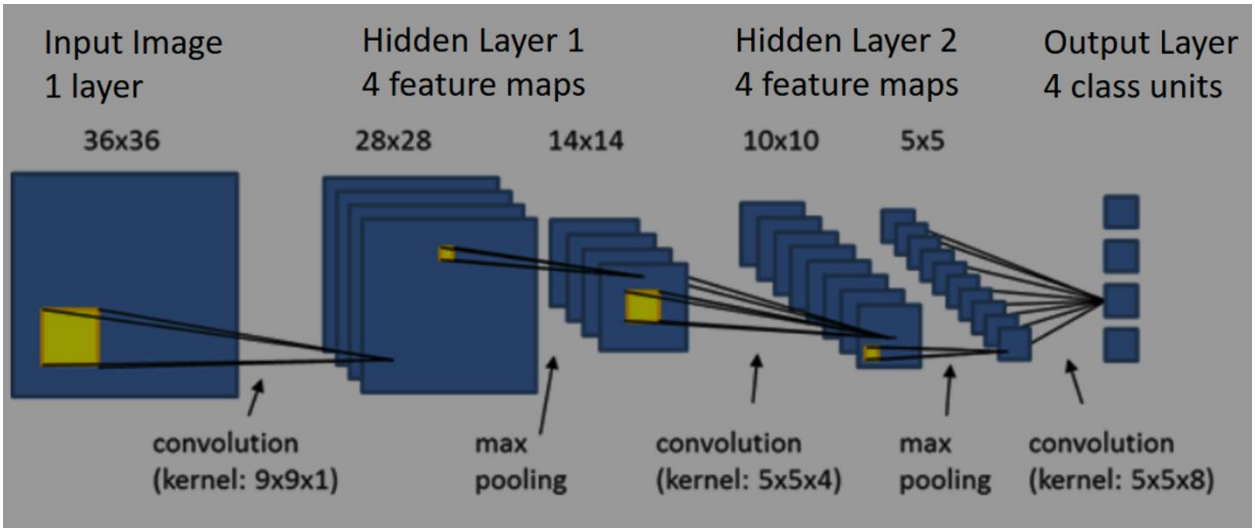

Figure 2-3. A simple CNN architecture

The CNN structure in Figure 2-3 contains a sequential chain of convolutional layers + subsampling layers, in this case max pooling. Each convolutional layer generates a set of feature maps. The number of the produced feature maps depends on the number of the convolutional kernels in each layer. For instance, in Figure 2-3, 4 convolutional kernels with size 9×9 are applied to the input image and generate 4 feature maps as the input to the subsequent layer, i.e. hidden layer 1. Each of the mentioned 4 feature maps is specified by $9 \times 9 = 81$ adjustable weights. Subsampling layers of CNN reduce the spatial resolution of the feature maps. The role of the subsampling layers is not only to reduce the dimensionality but also to help the network's responses be spatially invariant. Ultimately, the last layer outputs a vector with 4 components which can be used to classify the four categories of the input images. The task here was one of categorization. Like fully connected neural networks, trainable weights and biases in CNNs are optimized by SGD based algorithms through the backpropagation technique.

2.10 **Function Approximation by Artificial Neural Networks**

According to the universal approximation properties ANNs[30], a feedforward neural network with a single hidden layer can represent any continuous function on a closed and bounded subset of R^n. However, the number of the hidden nodes in the hidden layer may be impracticably large and, more importantly, may fail to learn and generalize correctly when exposed to new input data. The rationale behind using deep neural networks is that they need fewer nodes in the hidden layers and decrease the generalization error.

Given that the universal approximation theorem is correct, why do we need to use deep ANNs in the context of MRI? This section aims to provide reasons and clarify where and how deep ANNs can help us increase efficiency in the applications presented in this dissertation.

Several state-of-the-art classical approaches have been proposed in dynamic cardiac image reconstruction[31], such as k-t BLAST or k-t Sense, which are all based on the CS and PI concepts. So, it seems that in the well-explored reconstruction area, researchers had already developed mathematically based optimization methods to solve the inverse MR reconstruction problems. Why do we need to use deep ANNs, which are all data-dependent, to learn how to reconstruct the images from the undersampled data? One answer to this question is that the deep ANNs-based image reconstruction inference time is significantly lower than the classical approaches; in other words reconstruction speed is greatly increased. This is particularly important when accessing the reconstructed images is crucial to guide the subsequent scans. Another reason is that the deep ANNs can learn more effective regularizers and priors than the fixed regularizer used in the conventional methods. For instance, since the forward operation is already known, one can only use CNN to learn the effective Spatio-temporal regularizer in image reconstruction.

Deep ANNs have advantages over the classical methods in respiratory motion correction, particularly in free-breathing cardiac cine imaging. Because the nature of the respiratory motion in cardiac imaging is non-linear and despite the assumption of rigid motion, there is no well-defined relationship between the respiratory motion and the k-space measurements. In other words, using ANNs can help us learn the non-linear function that can map data corrupted by respiratory motion artifact to data free of the respiratory motion artifact. Another advantage is the inference time of the ANNs compared to the classical iterative-based algorithms is significantly lower.

Several algorithms have been proposed to estimate the T1/T2 relaxometry values in cardiac imaging. There is a direct relationship between the accuracy of the proposed conventional methods and their computational complexity. For example, Bloch-equation-simulation-based parameters estimation approaches[32] take more detail of the sequence into account, such as considering the effect of the non-rectangle 2D RF excitation slice profile, B1+ errors, and the imperfect inversion T2 preparation to improve the accuracy, but it is time-consuming. Therefore, in this case, using ANNs can substantially decrease the computation time while achieving a similar accuracy of Bloch-equation-simulation-based parameters estimation approaches in cardiac T1/T2 calculations.

The post-processing of CE-MRA images mainly includes segmentation of the peripheral vasculature, which an experienced radiologist often performs via visual inspection and manual delineations. Due to the large size of the high resolution, volumetric peripheral MRA, e.g., 560 x 940 x 240, manual annotation is a time-consuming and tedious process. Since manual labeling is a subjective process and depends on physician's experience and knowledge, it can potentially introduce high inter-observer variability. ANNs can adaptively find highly representative features from historical data through a training process and decrease the required computational time in

contrast to the conventional methods, which solely rely on a priori knowledge and hand-crafted features and require considerable time to complete the segmentation.

Chapter 3 Deep Learning based Dynamic Cardiac Magnetic Resonance Image Reconstruction Pipeline

The purpose of this work was to develop and evaluate a deep learning based dynamic cardiac magnetic resonance image reconstruction pipeline for low-latency and high quality accelerated MR imaging. A 3D CNN used to learn the spatiotemporal regularizer from the historical data. Standard conventional compressed sensing reconstruction k-t FOCUS[33] is compared in terms of reconstruction quality and speed. Quantitative evaluations confirmed that the proposed network was able to images with a lower noise level and reduced aliasing artifacts in comparison with k-t FOCUS. Using the proposed method, each frame can be reconstructed in less than 40ms, suggesting its clinical compatibility. In conclusion, the proposed deep learning based framework is a promising technique that allows low-latency and high quality cardiac MR imaging. A version of this chapter has been initialized as the Siemens Patents. Also, this chapter is the improved version of our previous studies[7,8] which have been published two years ago in the medical physics and Quantitative Imaging in Medicine and Surgery:

1. Zhou, Z., Han, F., Ghodrati, V., Gao, Y., Yin, W., Yang, Y. and Hu, P. (2019), Parallel imaging and convolutional neural network combined fast MR image reconstruction: Applications in low-latency accelerated real-time imaging. Med. Phys., 46: 3399-3413.

2. Ghodrati V, Shao J, Bydder M, Zhou Z, Yin W, Nguyen KL, Yang Y, Hu P. MR image reconstruction using deep learning: evaluation of network structure and loss functions. Quant Imaging Med Surg 2019;9(9):1516-1527.

3.1 Introduction

MRI acceleration methods are widely used to shorten image acquisition time by under-sampling k-space. Parallel imaging methods such as GRAPPA[4] and compressed sensing (CS)[6] are state-of-the-art approaches that are routinely used. GRAPPA uses a fully-sampled k-space center region to train convolution kernels which are subsequently used to fill in missing k-space samples. However, a potential challenge of parallel imaging is that at high acceleration factors, the g-factor could result in significant noise amplification. The CS method takes advantage of the intrinsic sparsity of the data in a specific transform domain and random k-space sampling (incoherent point spread function) to remove noise-like image artifacts in the image. CS-MRI typically uses predefined and fixed sparsifying transforms, e.g., total variation (TV), discrete cosine transforms and discrete wavelet transforms[5]. This can be extended to more flexible sparse representations learned directly from data using dictionary learning[34]. However, CS-MRI is associated with challenges in finding appropriate regularizers for specific applications and manually tuning the hyperparameters, a time-consuming process that is difficult to standardize. In addition, the optimization process involves non-convex terms, so there is no guarantee of achieving a global minimum or even converging to a solution.

Recent advances in deep neural networks open a new possibility to solve the inverse problem of MR image reconstruction in an efficient manner. Deep learning-based approaches are well-developed in computer vision tasks such as image super-resolution[35-38], denoising and inpainting[39-42], while their application to medical imaging is still at a relatively early stage. For MR image reconstruction, these approaches typically learn the proper transformation between the input (zero-filled under-sampled k-space) and the target (the fully-sampled k-space) by minimizing a specific loss-function through a training process. Recently, a few different networks have been used to

automate medical image reconstruction[43-51]. Jin et al.[43] focused on CT reconstruction and proposed a Filter Back Projection Convnet (FBPConv) to reconstruct the CT data 1,000 faster than classic methods while preserving the image quality. Sandino et al.[46] trained a Unet architecture on 3D cardiac datasets and compared the results based on pixel-wise loss functions. Hammernik et al.[48] proposed variational network to learn the effective priors to accelerate the knee imaging and shorten the acquisition and reconstruction time. Schlemper et al.[49] proposed a novel deep cascade network for dynamic image reconstruction and showed superior performance of their network to CS-MRI. They used the data sharing layer to learn the spatiotemporal correlation of dynamic cardiac imaging data, which substantially improved the performance of their network. Hyun et al.[50] used a simplified Unet and proposed a k-space correction method to improve the performance of their network in MR reconstruction. Zhu et al.[51] used fully connected layers followed by a convolutional autoencoder to directly map the k-space data to the image domain. Deep neural networks have also been used to explore much more effective image priors and sparsifying transforms from a given datasets and combined with conventional CS methods. As proposed in[52], the ADMM algorithm is used to solve the inverse problems such as CS-MRI. Another interesting technique[48] was recently reported using a variational autoencoder for learning the effective priors to reconstruct knee datasets. Generative adversarial networks (GANs) have been proposed to achieve a higher perceptual quality in inverse problems such as super resolution[53-56]. Additional new techniques have been proposed to increase the sharpness and preserve the texture information in MR reconstruction tasks[57,58]. Transfer learning has also been explored as an effective image reconstruction method [59,60].

In this work, we sought to develop a deep learning based pipeline and apply it to 2D dynamic cardiac imaging for low-latency online reconstruction. We used a 3D CNN to learn the effective

spatio-temporal regularizer and forced the data consistency by hard replacement in the k-space domain. We demonstrate the capability of our framework on dynamic cardiac imaging and compared its performance quantitatively against the k-t FOCUS[33].

3.2 Methods

This section briefly described the methods, including the general compressed sensing model, network structure, data preparation, and the training process. We highly recommended visiting our previous publications in MR dynamic cardiac image accelerations, which focused on the loss function and combined PI and CNN[7,8]. The crucial difference of this improved version with our prior works is including the temporal information to achieve higher accelerations (8X-10X) which is two times more than our prior PI combined with CNN work.

3.2.1 General Compressed Sensing Model

In general, the MR image reconstruction problem can be formulated as an inverse problem $Fx = y$, where x is the complex-valued image series formatted as an $N = N_x \times N_y \times N_t$ column vector, F is the Fourier encoding matrix, and y is the measurement k-space vector. Undersampling schemes are applied in practice to accelerate the acquisition process. Because of the applied undersampling, the inverse problem formulation would change to $F_u x = y_u$, where F_u is a composite operator with size M×N includes sensitivity maps S, Fourier encoding matrix F, and binary undersampling mask U. In MR undersampled inverse problems, usually, the number of measurements M is significantly less than the number of unknown N ($M << N$), thus the direct solution is not possible because of the underdetermined nature of the problem. So, in order to solve this problem, two general approaches have been introduced: Parallel Imaging (PI), which relies on using channel information to turn the underdetermined set of equations into an overdetermined

problem, or Compressed Sensing (CS), which takes advantage of the compact representation of the data in some transform domains to solve the underdetermined MR reconstruction problem. Our focus in this work is on the second category, i.e., CS. Conventional CS algorithms estimate the reconstructed image x by minimizing the unconstrained optimization problem:

$$\min_{x} \{ \frac{\mu}{2} \sum_{c=1}^{nc} \| UFS_i x - y_i \|_2^2 + R(x) \} \quad (3\text{-}1)$$

In the data consistency term (first term in Eq. (3-1)), U is the undersampling mask, F is the Fourier transform, and Si denotes the sensitivity map of the ith channel. μ controls the weight of the data consistency term, and the number of channels is presented by nc. The regularization term (second term in Eq.(3-1)) generally is a sparse regularizer such as total-variation or the first norm of the wavelet transform of x. There are two points about Equation 1 which deserved to note: 1) the regularizer term is a fixed operation and before optimization has to be decided, 2) solving such optimization problem requires an iterative algorithm which could increase the reconstruction time. In order to accelerate the reconstruction process and design a more powerful regularizer based on the historical data, we propose to use a convolutional neural network. In this work, we translated the CS reconstruction problem in Equation (3-1) into a deep neural network to accelerate the reconstruction process and learn a more powerful Spatio-temporal regularizer.

3.2.2 Network Structure

Figure 3-1 shows the network structure, which consists of 8 sequential stages in which each stage contains two parallel paths—weighted outputs with a learnable parameter of both paths combined through the training process. One path (DC) that does not include any learnable weights is the data consistency and another path (CNN3D), a learnable part, is a convolutional network, and its goal is to learn a proper Spatio-temporal regularizer. To design the data consistency, we

employed a hard replacement scheme. A hard replacement scheme is a simple and efficient way to force the data fidelity, and its function is based on filling the phase encoding lines of the k-space that have already been acquired by the original measurements. To implement a data-driven-based Spatio-temporal regularizer, we used a relatively shallow 3D convolutional network to use spatial and temporal redundant information of the dynamic series of images to learn an effective regularizer. We used five cascaded convolutional layers in each stage rather than a very deep convolutional network to minimize the overfitting issues.

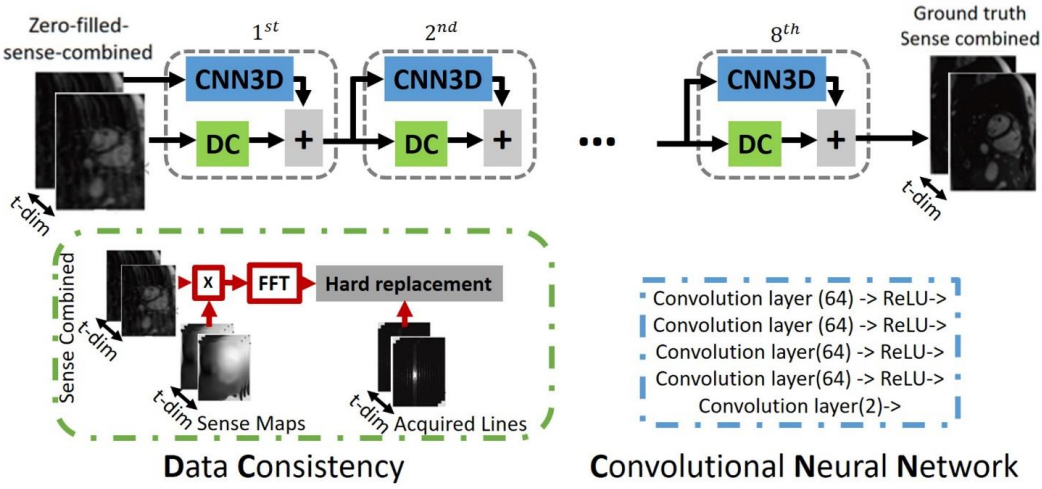

Figure 3-1. Network structure: eight sequential stages were considered in our end-to-end reconstruction pipeline. Each stage contains two parallel paths 1) CNN3D and 2) DC. CNN3D is a shallow five-layered convolutional neural network that aims to learn a Spatio-temporal regularizer. DC is the data consistency term that replaced the reconstructed phase-encoding lines with the real acquired phase-encoding lines. The output of the data consistency path in the image space is combined by the output of the CNN3D in the learnable fashion in each stage.

3.2.3 Data Preparation and Training

To train and evaluate the network, we used retrospectively acquired clinical breath-held 2D multi-slice, ECG-triggered, GRAPPA 2X, bSSFP cardiac cine MR images in the short-axis, horizontal long-axis, and vertical long-axis views from 42 patients. We divided this data into 25 patients' data (583 dynamic images) to train the network and 17 patient cases (272 dynamic images)

for testing the network. Since the retrospectively acquired data was based on the GRAPPA 2X, a proper ICE function was employed to reconstruct the free of the aliasing single-channel complex image for the data. Besides, a particular ICE function was used to extract the sensitivity maps for each dynamic set of images. The main reason behind using the ICE function to calculate the sensitivity maps was training the data on more realistic images, thus improving the performance of the network and its compatibility with the Siemens scanner.

For training, the network, five sets of data including sensitivity maps, multi-channel undersampled raw data (k-space), undersampling masks (binary masks), coil combined single-channel complex zero-filled dynamic images, and coil combined single-channel complex aliasing free dynamic images (target) are required. Figure 3-2 graphically summarized the required data for training the network. For the sake of clarity, data preparation was shown in Figure 3-2 for a single image; for dynamic images, this process has to be iterated through the dynamic frames. As illustrated in Figure 3-2, coil combination was achieved by summing up the images in an element-wise manner in the channel dimension.

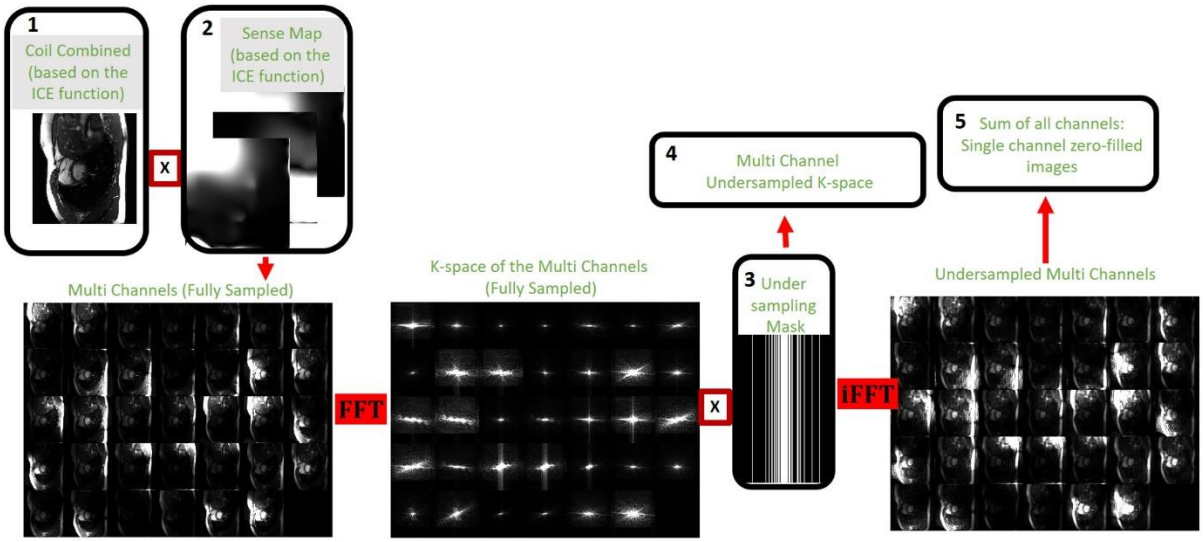

Figure 3-2. Data preparation pipeline for training the network. For training, the network five sets of data are required. These sets of data and the process of generation of them were shown inside the five black rectangles.

For testing the network, only four sets of the data, including sensitivity maps, multi-channel undersampled raw data (k-space), undersampling masks (binary masks), and coil combined single-channel complex zero-filled dynamic images are required. Three points are considered in designing the undersampling mask:

1. Initial sampling lines on (discretized) golden steps were calculated.

2. The initial calculated lines were repositioned based on a predefined density distribution.

3. Lines were sorted in a way that they zigzag across time to minimize the gradient jump.

Figure 3-3 shows an exemplary undersampling mask for 8X and 10X acceleration factors through the cardiac phases. For training the network, the Adam optimizer was used with the momentum parameter $\beta = 0.9$, mini-batch size= 1, and an initial learning rate of 0.0001 to minimize the L1 norm between the reconstructed dynamic images and the corresponding dynamic targets. Weights for the network were initiated with random normal distributions with a variance of $\sigma = 0.01$ and mean $\mu = 0$. The network was trained for five epochs, i.e., 5×8750 iterations in an end-to-end fashion based on the five sequential cardiac frames extracted from the dynamic training data. The training was performed with the Pytorch interface on a commercially available graphics processing unit (GPU) (NVIDIA Titan XP, 12 GB RAM). Once the network was trained, it was tested based on the full-sized dynamic images rather than five sequential dynamic frames.

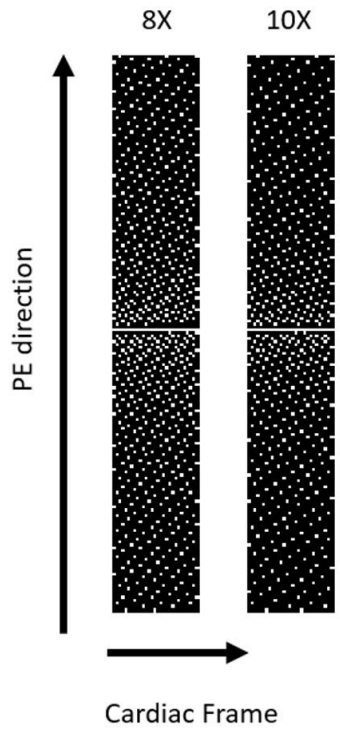

Figure 3-3. Undersampling binary mask for 8X and 10X acceleration factors. White points show the position of the phase encoding lines through the cardiac frames.

3.3 Result

In the following, qualitative and quantitative results were presented for 8X and 10X acceleration factors. Figure 3-4 shows the arbitrarily selected cardiac frame reconstruction results for the horizontal long axis (HLA) view of the cardiac image.

Figure 3-5 shows the exemplary reconstruction results for a short-axis view of the cardiac image. As evident in Figures 3-4 and 3-5, the proposed pipeline can recover the inter and intracardiac structure without tangible quality loss. In addition, it seems that the cardiac structure in the reconstructed image is not deformed and is similar to the ground truth image.

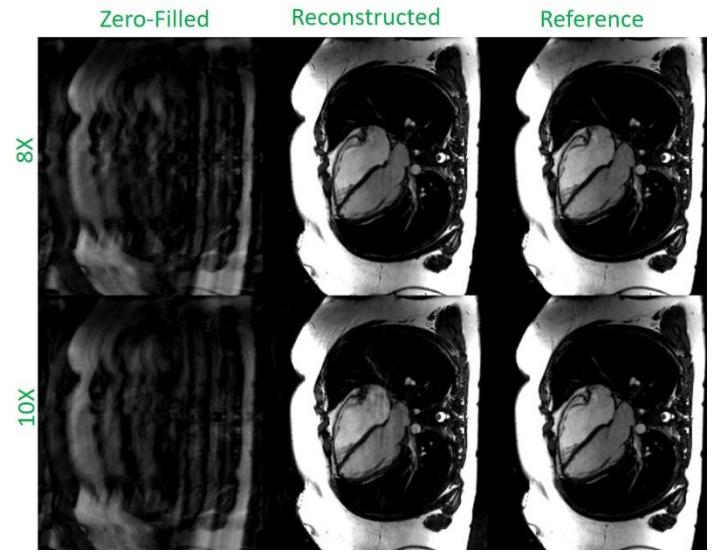

Figure 3-4. Qualitative reconstruction results of arbitrarily selected test data for the HLA cardiac view for 8X and 10X acceleration factors.

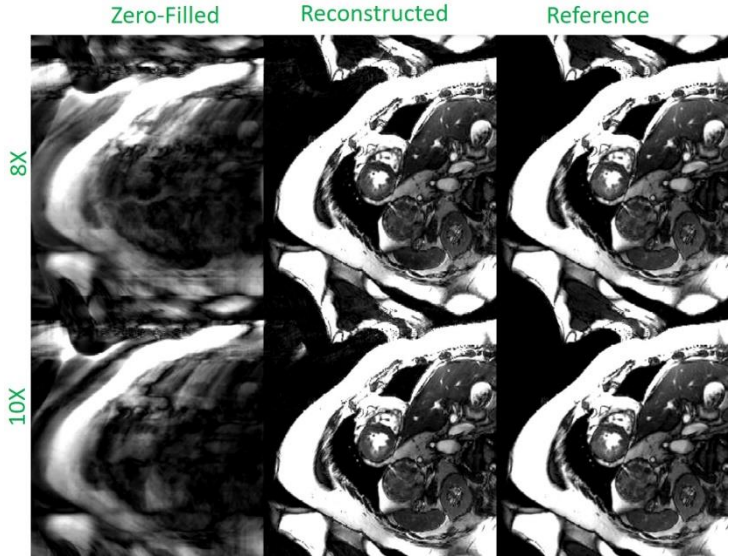

Figure 3-5. Qualitative reconstruction results of arbitrarily selected test data for the SAX cardiac view for 8X and 10X acceleration factors.

Table 3-1 shows the quantitative results for different methods and different acceleration factors.

Table 3-1. Quantitative comparisons: Our proposed pipeline achieved significantly higher SSIM and lower MSE than our pipeline without incorporating temporal information and the classic state-of-the-art k-t FOCUS.

	Mean SSIM $x10^{-2}$		Mean MSE $x10^{-3}$	
	8X	10X	8X	10X
Our pipeline	91*	89*	0.3*	0.4*
Our pipeline w.o. temporal information	78	74	0.9	1.2
k-t FOCUS	80	77	0.7	1.1

* There was a statistically significant difference ($P<0.05$) between the proposed method and other methods concerning the quantitative metrics SSIM and MSE.

3.4 Discussion

In this work, we proposed a deep learning-based pipeline to reconstruct the cardiac cine MR images from the undersampled measurements. We considered the redundancy in the temporal dimension to achieve a higher acceleration factor. Based on Table 3-1, it seems that the pipeline which includes the temporal information has significantly better performance than the same pipeline without temporal information. Also, compared to the state-of-the-art CS-based method (k-t FOCUS), our pipeline achieved statistically better quantitative scores, i.e., MSE and SSIM. Using a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz), the reconstruction time was approximately 30 ms/cardiac phase for the proposed pipeline and 300 ms/cardiac phase for k-t FOCUS.

Two significant concerns exist in the use of deep neural networks in image reconstruction tasks in medical imaging. First, whether these networks were able to preserve the pathologies in reconstructing the highly undersampled and respiratory motion corrupted images; and Second, whether these networks introduced new spurious anatomical features in the images. We included the L1 loss to constraint the network's output in our proposed method in the image domain to address the first concern. Besides, data consistency is applied in the k-space domain in a hard

replacement scheme; in other words, forward operation in the reconstruction formulation is incorporated in our pipeline. To address the second concern, we asked two radiologists to carefully examine the reconstructed results of the proposed method to evaluate the images concerning the newly spurious anatomical features. Based on the radiological assessments on all reconstructed images of the test dataset, there were no new introduced spurious features in the reconstructed images.

For future studies, expanding the pipeline to the multi-task-based platform could potentially increase the efficiency of cardiac imaging. For instance, one could add more output nodes to get the required post-processed stage of the cardiac imaging, e.g., ejection fraction, etc. in order to change the single task platform to a multi-task platform proper loss functions and balancing between them are required and will be considered in the future studies.

3.5 Conclusion

Deep learning-based image reconstruction helped us to achieve the 8X-10X acceleration in 2D cardiac imaging. Such a high acceleration is achieved by taking advantage of the redundancy in the temporal dimension of the data. Also, the designed platform outperformed the state-of-the-art classic k-t FOCUS in terms of the quantitative MSE and SSIM scores.

Chapter 4 Retrospective Respiratory Motion Correction in Cardiac Cine MRI Reconstruction

This work aimed to develop a deep neural network for respiratory motion correction without accessing the paired data in free-breathing cine MRI and evaluate its performance. To achieve that, we trained an adversarial autoencoder based on the unpaired data from the healthy volunteers and patients who underwent clinically indicated cardiac MRI examinations. The autoencoder learns the identity map for the free-breathing motion-corrupted images and preserves the structural content of the images, while the discriminator, which interacts with the output of the encoder, forces the encoder to remove motion artifacts. We used a U-Net structure for the network's encoder and decoder parts and regularized the code space by an adversarial objective. We performed two separate evaluations. First, we evaluated the network based on the data that were artificially corrupted with simulated rigid motion concerning motion correction accuracy and the presence of any artificially created structures. Second, we evaluated the network on the real cases, including the patient and the volunteers whose images were corrupted by the respiratory motion artifact. A version of this chapter has been published[9] in the NMR in Biomedicine:

1. Ghodrati V, Bydder M, Ali F, Gao C, Prosper A, Nguyen KL, Hu P. Retrospective respiratory motion correction in cardiac cine MRI reconstruction using adversarial autoencoder and unsupervised learning. NMR Biomed. 2021 Feb; 34(2):e4433.doi: 10.1002/nbm.4433.

4.1 Introduction

In current clinical practice of thoracic and abdominal MRI, images are commonly acquired during a breath-hold to compensate for respiratory motion. Physiological limitations of breath-holding constrain the data acquisition window to approximately 15-20 seconds in relatively healthy patient populations. In clinical practice, many patients undergoing MRI have impaired breath-holding abilities, further limiting the acquisition window. As a result, 3D acquisitions are not routinely performed during a single breath-hold, despite previous efforts[61-64]. Many approaches have been proposed to enable free-breathing thoracic and abdominal MRI, including real-time single-shot cine imaging[65-67] and the use of non-Cartesian sampling (e.g. radial), which tends to be less sensitive to respiratory motion[68-70]. However, the use of these approaches is not without compromise. For example, single-shot imaging approaches are generally of inferior image quality, signal-to-noise ratio or resolution when compared to their corresponding k-space segmented techniques. Non-Cartesian sampling, although relatively immune to motion, is prone to various other types of image artifacts, including streaking, off-resonance blurring and issues related to gradient delays. Alternative methods for respiratory motion compensation include respiratory bellows gating[71-73], diaphragm navigators[74], and MR self-gating[75]. These techniques, as a whole, result in prolonged scan time and reduced scanning efficiency, as a significant portion of the data is discarded. In addition to longer acquisition times, these techniques each suffer from their own respective drawbacks. Respiratory bellows rely on air pressure signal, which may not always have a well-defined correlation with the respiratory position of various anatomical structures. Diaphragmatic navigators and MR self-gating navigators have enabled high quality imaging of the coronary arteries[76,77]; however, their adoption in routine clinical imaging remains limited, in part because irregular and abrupt breathing pattern changes often reduce image quality and reliability.

Multiple methods have been proposed for motion correction[78-80], where motion is corrected in k-space using well-known relationships between affine motion and the corresponding k-space. However, these corrections are often inadequate because of significant non-rigid and deformable motion, which does not have well-defined k-space correction methods.

In this work, we sought to investigate the use of deep neural networks (DNNs) for respiratory motion compensation in MRI to alleviate some of the aforementioned problems. DNNs, particularly convolutional DNNs, have presented new possibilities for tackling a wide range of inverse problems including image inpainting, super resolution[35-38], denoising and deblurring[39-41, 81-84] in an efficient manner. The main advantage of DNNs over classical data processing approaches is that it learns the effective features and priors in a data-driven fashion. To date, few studies have implemented DNNs for motion compensation[84-88]. Recent studies have shown that DNN can correct rigid-motion artifacts in brain imaging[85,88]. They mainly trained convolutional neural networks with pixel-wise objective functions in a supervised manner. Haskell et al. combined a deep convolutional network with model-based motion estimation approach in an iterative manner to reduce the rigid motion artifacts from the 2D T2-weighted rapid acquisition with refocused echoes (RARE) brain images[88]. Their algorithm is of iterative nature, and in each iteration, the output of the convolutional neural network (CNN) was used to estimate the motion parameters and to correct the image k-space. They used time series registration information from fMRI scans to create the realistic motion trajectories. Then, they modified motion-free raw k-space brain data to synthesize realistic rigid-motion-corrupted images, and subsequently they estimated the motion parameters and forced the data consistency.

However, for DNN-based respiratory motion compensation in cardiac and abdominal imaging, supervised learning approaches are generally not feasible because the ground truth non-rigid

motion data, which is needed for training the network, is either extremely challenging to obtain or simply not available. Kustner et al. reported a feasibility study to correct rigid and non-rigid motion artifacts by implementing a conditional generative adversarial network (GAN) (MedGAN), in which the generative network consists of eight cascaded U-nets[84]. The network was trained using a combination of adversarial, style transferring, and perceptual[89] loss functions. Among the loss functions used by Kustner et al.[84], the perceptual loss function requires paired data, which is challenging to obtain, especially for non-rigid motion correction tasks. In addition, there are $>10^8$ trainable parameters in the network architecture used by Küstner et al.[84] Armanious et al. used a cycle consistency approach to extend the MedGAN in a way that can be trained in unsupervised manner[90]. They incorporated an attention module in their generator network to capture long-range dependencies. They mainly focused on reducing rigid simulated artifacts from brain datasets and achieved promising results.

The goal of this study was to develop and validate a DNN-based platform to remove respiratory motion artifacts in free-breathing imaging. We chose 2D cardiac cine imaging as an exemplary target application to validate our technique. In particular, based on the numerous challenges associated with obtaining the ground truth non-rigid motion data, we aimed to develop a network that can be trained in an unsupervised manner. In particular, our DNN is based on an adversarial autoencoder[91-93] network structure to take advantage of its ability to be trained in a self-supervised manner without access to paired training data or the ground truth motion data. In our work, the encoder and decoder part of the adversarial autoencoder are both convolutional U-nets. The autoencoder's code space is regularized with an adversarial loss network. The autoencoder preserves anatomical accuracy and consistency during the motion correction process while the adversarial network regularizes the encoder and drives the code space to be as close as possible to

a motion artifact-free image. By leveraging the intrinsic competition between these two networks during the training process, we expect motion-corrected, artifact-free images to preserve their fidelity with regard to the overall anatomical structure and consistency.

4.2 Methods

4.2.1 Theory

An autoencoder is a neural network that reconstructs an output that is almost identical to its input with the goal of learning useful representations of the input data[94]. Figure 4-1(a) shows a general architecture of an autoencoder. It consists of two parts, the encoder and the decoder. The encoder and decoder can be expressed as $En_\theta(z|x)$, $De_\upsilon(\hat{x}|z)$: where, z represents the code space and θ, υ are the learnable parameters of the encoder and decoder networks, respectively. Equation (4-1) formulates the objective function of the autoencoder network as an L1-norm minimization problem:

$$min_{\theta,\upsilon} E_{x\epsilon X}(|x - De_\upsilon(En_\theta(x))|_1) , \tag{4-1}$$

where, x is a batch of the data selected from dataset X. In general, putting constraints on the autoencoder, such as limiting the dimension of code space or adding regularization to the code space prevents them from learning a trivial identity mapping. In our problem, the code space has the same dimension as the input data. Therefore, a proper approach is to add a special regularizer to the code space to produce the motion-corrected images. Equation (4-2) describes the objective function of the regularized autoencoder:

$$min_{\theta,\upsilon} E_{x\epsilon X}\left(\left|x - De_\upsilon\left(En_\theta(x)\right)\right|_1 + \beta R(En_\theta(x))\right), \tag{4-2}$$

where β is the tuning parameter for the regularizer R. Such a regularizer needs to be capable of assessing the presence and extent of significant motion artifacts in the image and the regularizer needs to be differentiable.

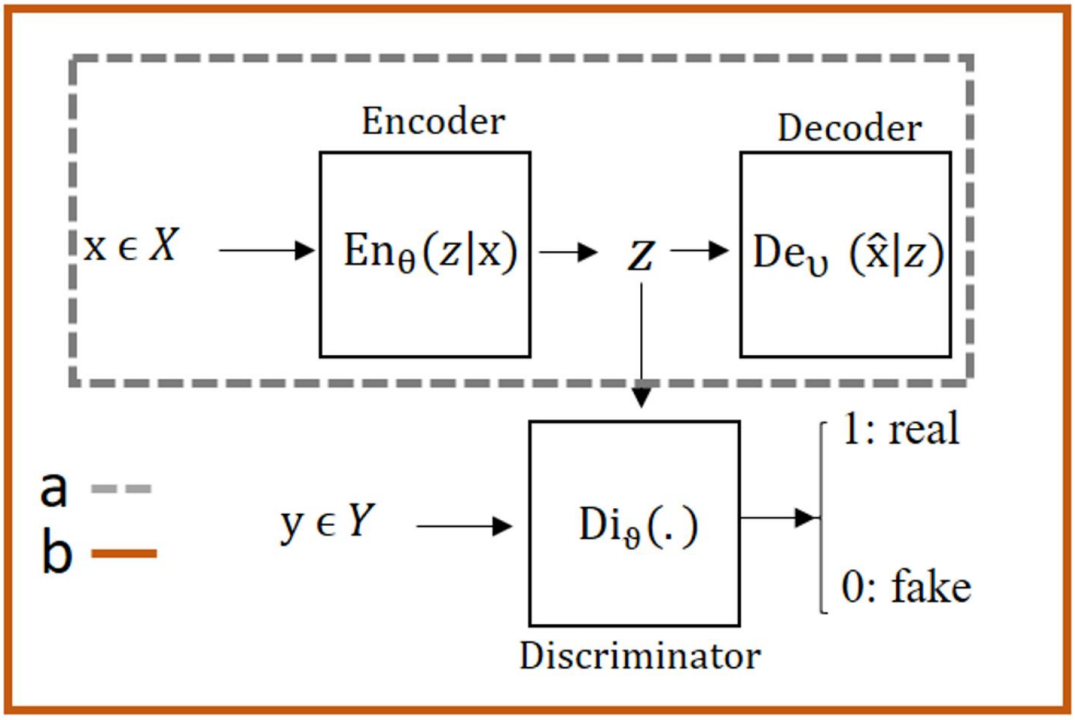

Figure 4-1. Various structures of the autoencoder: a) A simple autoencoder which encodes the high dimensional inputs into the code space data, which is usually of substantially reduced dimensions, by applying a series of convolutional layers. The decoder recovers the same input data from the code space data. b) An adversarial autoencoder (AAE) combines a simple autoencoder with an adversarial regularizer called discriminator to the code space. The discriminator is trained with the goal of accurately differentiating between data generated for the code space of the autoencoder and the data from the external data source Y. The adversarial autoencoder is trained with the goal of generating the code space data that resemble the external source data Y. The end result of the AAE network is that the code space data are driven to represent the external data source as much as possible during the adversarial (and competing) training processes between the encoder part of the autoencoder and the discriminator. In the context of our motion correction work using AAE, the encoder and decoder networks are each a convolutional U-net, the input x of the autoencoder is a free-breathing motion-corrupted image, the code space data is the corresponding motion-corrected image of the same dimensions, and the external data source Y is unpaired standard breath-hold motion-free reference images. The code space is driven by the discriminator network to be motion-corrected images such that they resemble the motion-free images from the external source Y. More details about the structure of our network are included in Figure 4-2.

Although without access to paired data, an explicit form of the metrics to be used for such assessment may not exist, it can be learnt via the neural network. Such a neural network uses an

adversarial loss to force the code space to be similar to the motion-free images. Figure 4-1(b) shows a graphical view of the proposed platform. As can be seen in the Figure 4-1(b), an adversarial objective is added to the conventional autoencoder structure to regularize the output of the encoder. The input of the encoder $x\epsilon X$ is the motion-corrupted image acquired during free breathing without any means of motion compensation. The output of the encoder z is one of the inputs of the discriminator network during network training. In addition, the discriminator has access to motion artifact-free images Y that are not necessarily paired with the input X for the encoder. Through the training process, the discriminator drives the encoder to correct motion artifact in such a way that the discriminator network is not able to distinguish between unpaired high-quality images acquired during breath-holds that are free of motion artifacts and the motion-corrected images generated by the encoder. Equation (4-3) shows the adversarial regularizer of the network:

$$min_{\theta_{En}} max_{\vartheta_{Di}} \; E_{y\epsilon \mathrm{Y}}[logDi_{\vartheta}(y)] + E_{x\epsilon \mathrm{X}}[log(1 - Di_{\vartheta}(En_{\theta}(x))] \,, \tag{4-3}$$

where ϑ and θ are the trainable parameters of the discriminator and encoder networks, respectively. Equation (4-4) shows the full objective function of the entire proposed network with regularizer weight $\beta = 1$:

$$min_{\theta,\upsilon} E_{x\epsilon X}(|x - De_{\upsilon}(En_{\theta}(x))|_1) + min_{\theta_{En}} max_{\vartheta_{Di}} \; E_{y\epsilon \mathrm{Y}}[logDi_{\vartheta}(y)] + E_{x\epsilon \mathrm{X}}[log(1 - Di_{\vartheta}(En_{\theta}(x))] \tag{4-4}$$

The first term in the Equation (4-4) represents the reconstruction objective and it preserved the overall accuracy of the motion-corrected images with regard to anatomical structure and image content. The second term in Equation (4-4) is the regularizer and its role is to force the encoder part to produce the images with similar appearance as motion-free images. The detailed network

structure, including the layers and number of kernels, are shown in the Figure 4-2. In our cardiac cine imaging validation, the input for the encoder was motion-corrupted free-breathing cardiac cine data acquired with a conventional k-space segmented cardiac cine imaging sequence during free breathing. The high-quality imaging data were obtained using standard clinical breath-held cardiac cine data from patients who underwent clinically indicated cardiac MRI. Because the proposed platform does not require paired data for training and can be trained in a self-supervised fashion, the high-quality breath-held data could be obtained from a cohort of subjects separate from the motion-corrupted data.

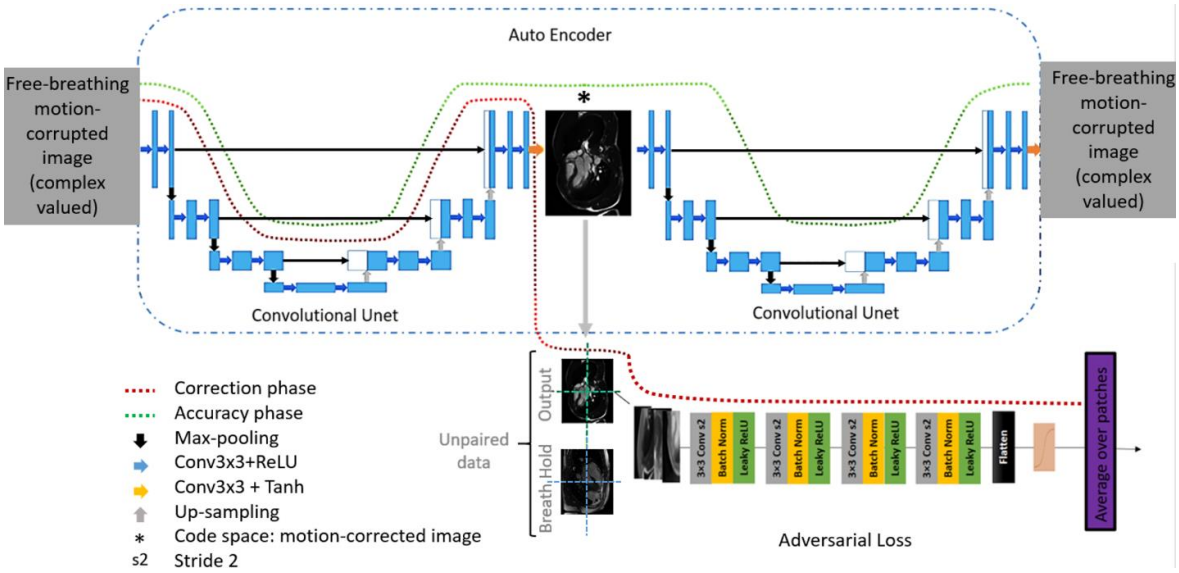

Figure 4-2. The autoencoder part consists of two convolutional U-net. A U-net consists of two paths: (I) the contracting path, which contains 3 down-sampling stages; (II) the expanding path, which includes 3 up-sampling stages. In order to preserve high-level features, it consists of dense connections from the early stages to the later stages of the network. Each convolution layer used in the U-net consists of trainable convolution filters (stride = 1) followed by rectified linear unit (ReLU) as a non-linear activation function except for the last layer. If ReLU was used in all layers as the non-linear activation function, then the network would only be able to learn to map to positive values. Because the input data sets were normalized to the range of [-1,1], it is important for the last layer's activation function to have the capability to pass negative values in the forward propagation process. Therefore, in the last layer, we used the hyperbolic tangent (Tanh) function as the activation function for both U-Nets, which makes mapping the input to the range of [-1, 1] possible. The discriminator part is a regularizer of the code space and it consists of 4 main convolutional layers. Each layer in this network included trainable convolution filters (stride = 2) followed by batch normalization and Leaky ReLU. The starting number of channels used in the discriminator was

64, which was doubled after each strided-layer. At the end of the network we used a flattened layer which vectorized the extracted features from the last convolution layer and passed it to nonlinear sigmoid function followed by averaging function. All the convolutional kernels used in either the U-net or the discriminator had 3x3 size. In total, each U-net in our platform approximately has 5.2×10^6 trainable parameters and the discriminator has 1.7×10^6 trainable parameters.

4.2.2 Training Procedures

In the training phase, the autoencoder and the discriminator network were trained with Stochastic Gradient Descent (SGD) in two phases – the accuracy phase and the correction phase. In the accuracy phase, the autoencoder updates its encoder U-net and the decoder U-net to minimize the reconstruction error of the input. In the correction phase, the discriminator and the encoder U-net were trained in an adversarial manner, where the discriminator first updates its structure to distinguish between the high-quality cardiac cine data and the samples from the output of the encoder; subsequently, the encoder U-net was updated to produce images that are as similar as possible to the high-quality cardiac cine data. Both networks, autoencoder and the discriminator, were trained in an end-to-end fashion and updated in the training phase sequentially mini-batch after mini-batch. Our training algorithm is summarized in Table 4-1.

Table 4-1. Training algorithm: De(.), En(.), and Di(.) stands for the Decoder, Encoder, and Discriminator, respectively. First two lines belong to the accuracy phase of the training process and the remaining lines belong to the correction phase.

Algorithm 1 Minibatch stochastic gradient descent training of adversarial autoencoder network.
For number of training iterations do: • Sample minibatch of m motion corrupted examples {x1, x2… xm} from motion-corrupted set X. • Update the Encoder and the Decoder by ascending its stochastic gradient: $\nabla_{\theta_{En},\nu_{De}} \frac{1}{m}\sum_{i=1}^{m}\left\|x^i - De(En(x^i))\right\|_1$ • Sample minibatch of m motion corrupted examples {x1, x2… xm} from motion-corrupted set X. • Sample minibatch of m breath-hold examples {y1, y2… ym} from motion-free set Y. • Update the discriminator by ascending its stochastic gradient: $\nabla_{\vartheta_{Di}} \frac{1}{m}\sum_{i=1}^{m}\left[logDi(y^i) + \log\left(1 - Di\left(En(x^i)\right)\right)\right]$ • Sample minibatch of m motion corrupted examples {x1, x2… xm} from motion-corrupted set X. • Update the Encoder by descending its stochastic gradient: $\nabla_{\theta_{En}} \frac{1}{m}\sum_{i=1}^{m}\left[\log\left(1 - Di\left(En(x^i)\right)\right)\right]$

Relatively large image patch size of 128×128 was used as the input to the autoencoder network. Previous studies show that generation of large size image in an adversarial manner is difficult compared to smaller size i.e. 64×64, because larger image patch size generally makes it easier for the discriminator to differentiate between the images provided by the generator and the high-quality data[95,96]. Most stable adversarial training methods were based on 64×64 patch size[97]. In order to stabilize the adversarial training process, a Markovian-patch-based approach was used to train the correction phase network[98]. During the training process, the output of the encoder for

each epoch was divided into 4 patches of size 64×64 and the discriminator either accepts or rejects the decision based on the average probability calculated for the 4 patches.

To update the weights of the correction (encoder + discriminator) and accuracy network (autoencoder), the Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 64, and initial learning rate 0.0001 that is halved every 15,000 iterations. All the weights were initiated with random normal distributions with a variance of σ = 0.01 and mean μ=0. The first iteration was started by updating the accuracy network and for the second iteration, the decoder part was kept frozen with no updates while the correction network was updated. This process was continued and, in each epoch, we produced the test results to make decision for stopping criteria. The training was performed with the Tensorflow interface on a commercially available graphics processing unit (GPU) (NVIDIA Titan XP, 12GB RAM). We allowed 125 epochs that took approximately 11 hours for training.

4.2.3 Data Acquisitions

To evaluate the performance of the proposed neural network and demonstrate its utility, we tested our strategy for cardiac cine imaging. Our institutional review board approved the study, and each subject provided written informed consent. The datasets used to train and test our network consisted of three groups:

1) Free breathing 2D multi-slice, retrospective ECG-triggered, balanced steady state free precession (bSSFP) cardiac cine MR images in the short- and long-axis views from 20 healthy volunteers (Avanto Fit, Siemens Healthineers). The sequence parameters included TR (repetition time)/TE (echo time)=2.44/1.19 ms, FOV(field of view)=271 x 300 mm^2, resolution= 1.74 x 1.92 mm^2, 25 cardiac phases, slice thickness=6 mm, 3-10 slices, acquisition time per slice =8-12s. As

the data were acquired using standard clinical cardiac cine imaging sequences but during free breathing, they were contaminated by respiratory motion artifacts. For comparison purposes, the same sequence was repeated during breath-hold for each healthy volunteer. The free-breathing acquisition time was similar to breath-hold.

2) Standard clinical breath-held 2D multi-slice, retrospective electrocardiogram (ECG)-triggered, bSSFP cardiac cine MR images in the short-axis, horizontal long-axis, and vertical long-axis views from 162 patients. These images were acquired as part of clinically indicated cardiac MRI scans and were collected retrospectively. These images were acquired during breath-holds, had 10-14 slices (one slice per breath-hold of 8-12s with 5-10s pause time between breath-holds), and were used as the high-quality imaging data for the adversarial network.

3) Standard ECG-triggered, bSSFP breath-held cardiac cine images in the short-axis and horizontal long-axis views were acquired from 10 additional patients as part of their clinically indicated cardiac MRI examination. In addition, in each of the 10 patients, we performed the same cardiac cine imaging sequence during free-breathing.

Before the network was trained and tested during our in-vivo study, we performed a simulation study based on data from Group 2 with simulated rigid motion. The goal of the simulation study was to confirm our technique's motion correction accuracy by commonly used metrics such as peak signal to noise ratio (PSNR) and structural similarity index (SSIM), which would not be possible for the in vivo study due to lack of ground truth data. More details of the simulation study are in the Evaluations section.

For the in vivo network training and validation, we used 15 out of the 20 volunteers' data from Group 1 and all of the breath-held cardiac cine imaging data from the 162 patients in Group 2.

Images in the Cine data were treated as independent samples, i.e., the temporal correlation was not considered during the network training and image reconstruction.

Our network training process was performed in an un-paired fashion. All the breath-held data from Groups 1&2 were shuffled randomly in each training batch before they were used as input data for the discriminator network. All the free-breathing data from Group 1 were randomly shuffled as well before they were used as input data for the autoencoder network. The anatomical orientation (short axis or long axis) was matched between the input data for the autoencoder and the input data for the discriminator network for each training batch. Our network testing was based on the remaining 5 volunteers' data in Group 1 and all the data from Group 3.

To increase the flexibility of the network in correcting the motion artifact for arbitrary image sizes, our network was trained and validated based on 128 × 128 patches extracted from the datasets. Each data set was reconstructed by applying adaptive coil combination to a single complex image and normalized to -1 to 1. Each single complex image was formatted as a real tensor with real and imaginary channels. In total, 125000 patches were used to train the network and 25300 patches were used to validate the network. Once the network was trained, the network testing was performed using full-sized images rather than image patches. As our encoder network has 3 down-sampling stages, we simply padded the input images to the next size that is divisible by 8 before they were input to the network.

4.2.4 Evaluations

Evaluation of the network performance consisted of four main parts:

a) Motion Correction Accuracy: One major concern is whether the reconstructed image is consistent with the breath-held reference. Due to the generative nature of the adversarial

autoencoders, it is important to ensure motion accuracy with regard to structural and anatomical content. To evaluate the motion correction accuracy and confirm that the proposed platform is capable of correcting the motion-corrupted datasets in un-paired training process, a simulation study was conducted. 1D translational respiratory motion of the diaphragm with variable displacements ranging from 10 to 20 pixels was introduced to corrupt the k-space data from Group 2 using a well-known relationship between k-space and the image space as shown in Equation (4-5), where the simulated translation vector is (x_0, y_0), M_1 and M_2 are the k-space data before and after motion corruption, respectively.

$$M_2(k_x, k_y) = e^{-j2\pi(k_x x_0 + k_y y_0)} \times M_1(k_x, k_y) \quad (4\text{-}5)$$

In our simulation, k-space of the Group 2 was divided into 16 segments. To find the diaphragm position for each segment, the respiratory signal was divided to 800-ms cardiac cycles and each cycle was divided into 20 cardiac segments. Inferior-superior diaphragm position was expressed as a function of time[99] in Equation (4-6)

$$y(t) = y_0 - b[\cos\left(\frac{\pi t}{T} - \varphi\right)]^{2n}, \quad (4\text{-}6)$$

where y_0, $y_0 - b$ are the position of diaphragm during end-exhalation and end inhalation, T and φ are the period and initial phase of the respiratory cycle, and n controls the shape of the simulated motion curve. For average diaphragm motion as described in[99], $\varphi = 0$, $n = 3$, and $T = 4sec$ were selected. Our simulated motion assumed that each pixel had an isotropic size of 1 mm^2, $y_0 = 5$ and b varied from 10 to 20 pixels.

Figure 4-3(a) shows details of simulation process for an image with 256 phase-encoding k-space lines. The respiratory cycle was divided into 5 cardiac cycles, where each cycle was further divided into 20 cardiac phases.

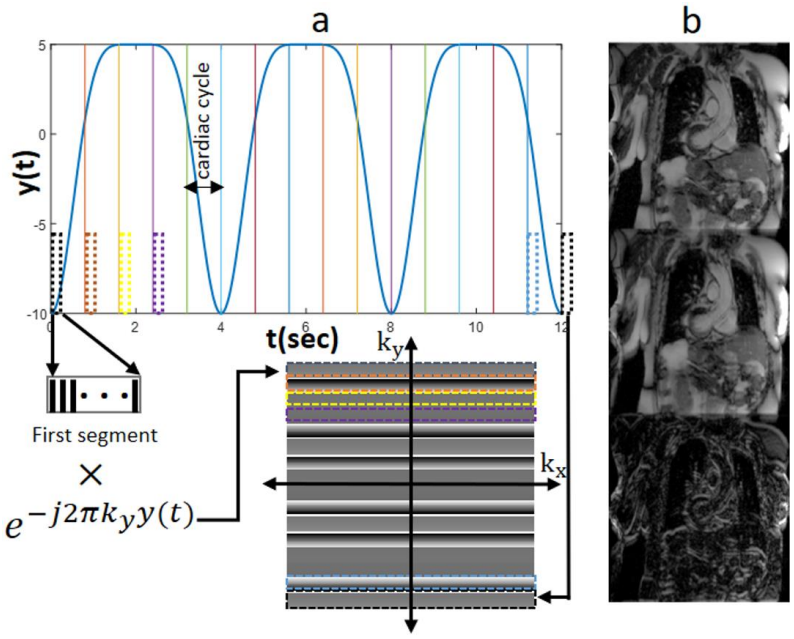

Figure 4-3. Motion simulation process of the Simulation Study. a) respiratory motion pattern and corruption process. Each k-space line is intentionally corrupted by adding a signal phase term that corresponds to the simulated motion distance for the line. b) (From top to bottom) Sample of the original motion-free image, the synthesized respiratory motion-corrupted image, and the error map between them.

In Figure 4-3(a), only the first cardiac phase in each cardiac cycle is shown as a dashed-rectangle. To simulate the motion-corrupted image, each data were divided to 16 k-space segments and each segment was multiplied by the phase term corresponding to its motion on the simulated motion curve shown in Figure 4-3(a) according to Equation (4-5). In our simulation study, all clinical breath-held cine data in Group 2 were used to synthesize the motion corrupted datasets. Out of the 162 synthesized motion-corrupted data sets, 20 were randomly chosen as testing data and were excluded from the training process. The remaining 142 data sets were used to train the network in an unpaired manner. Figure 4-3 (b) shows an example of the synthesized images and

artifacts. These images with synthesized artifact also enabled us to partially prove that our network does not produce extra structures. Assessment of motion correction accuracy was performed by calculating Tenengrad focus measure, PSNR and SSIM for the simulated test data on the image level.

b) Quantitative Sharpness: To quantify the sharpness of an image, the Tenengrad focus measure was used[100,101]. To calculate the Tenengrad focus measure, the image is convolved with a Sobel operator and the square of all the magnitudes greater than a threshold is reported as a focus measure. Equation 4-7 formulates the Tenengrad measure:

$$F_{Tenengrad} = \sum_{i,j}[I(i,j) ** S]^2 + [I(i,j) ** S^T]^2 , \quad (4\text{-}7)$$

where $I(i,j)$ shows the image and S is the Sobel operator: $S = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -2 \end{bmatrix}$. Because of difference in the size of the test cases in both simulation and in-vivo studies, the mean of the Tenengrad measure without threshold was calculated and normalized based on the breath-hold value.

c) Subjective Image Quality Scoring: The motion-corrupted test data from the 5 testing volunteer data sets in Group 1 and the 10 patient testing data sets in Group 3, their corresponding motion-corrected images after our network, and the corresponding breath-held reference cardiac cine images were randomized and presented to an experienced reader with >5 years of experience in reading clinical cardiac MRI who was blinded to the acquisition technique or patient information. The reader scored each of the images, which were presented as cine movies, with regard to image quality with an emphasis on motion artifacts according to the criteria in Table 4-2 [102,103].

Table 4-2. Subjective Image Quality Scoring Criteria

Score	Criteria
1	poor image quality; non-diagnostic
2	fair image quality; diagnostic image, but very blurry endocardial borders without clear definition of fine intra-cardiac structures
3	good image quality; diagnostic image, with less blurry endocardial borders and without clear definition of fine intra-cardiac structures
4	good image quality; diagnostic image, with sharp endocardial borders and without clear definition of fine intra-cardiac structures
5	excellent image quality; diagnostic image, with well-defined endocardial borders and clear definition of fine intra-cardiac structures

d) Cardiac Function Analysis: Motion-corrected images were further evaluated with regard to indices of cardiac function measurements, including left ventricular end-diastolic volume (EDV), end systolic volume (ESV), stroke volume (SV), and ejection fraction (LVEF). These indices were measured from automatic segmentations of epicardial and endocardial left ventricular borders using a commercial tool (Arterys Cardio DL, Arterys Inc, San Francisco, CA). The cardiac function analysis was based on 5 of the test cases, which had full stack of short-axis-view images covering from the apex to the base. The same cardiac function measurements were repeated for the clinical standard breath-hold cardiac cine images acquired on the same 5 subjects.

4.2.5 Statistical Analysis

Statistical analysis was performed using R (version 3.5.3). Statistical tests were applied on the subjective image quality scores to answer two main questions: 1) was there any statistical difference between the motion-corrected, breath-held, and motion-corrupted images? 2) If yes, among the mentioned groups, which pairs had statistically significant difference? To answer these

questions, Friedman's two-way analysis[104] and non-parametric paired comparison tests were applied. Significance level for all statistical test was assumed at $\alpha = 0.05$.

4.3 Result

4.3.1 Simulation Study

Figure 4-4 shows representative examples of artifact-free images, artificially motion-corrupted images, and motion-corrected images. Based on the absolute error map, the proposed network was able to sharpen the edge and remove the ghosting artifact without generating extra structures.

Figure 4-5 shows the frequency plot of SSIM and PSNR scores for the simulated test datasets. Mean value (green circles) and 95% confidence interval (black lines) were also added to the top of each chart. Based on the SSIM scores, the proposed network produced images that were structurally similar to the ground truth and increase the SSIM 22% in comparison to motion-corrupted images. Also, the PSNR results show that our motion correction network was able to reduce the residual errors and improve PSNR by 25% in comparison to motion-corrupted images. The normalized Tenengrad focus measure was 0.82±0.06 for the motion-corrupted images and 0.92±0.04 for the motion-corrected images, representing a 12% increase using the proposed technique.

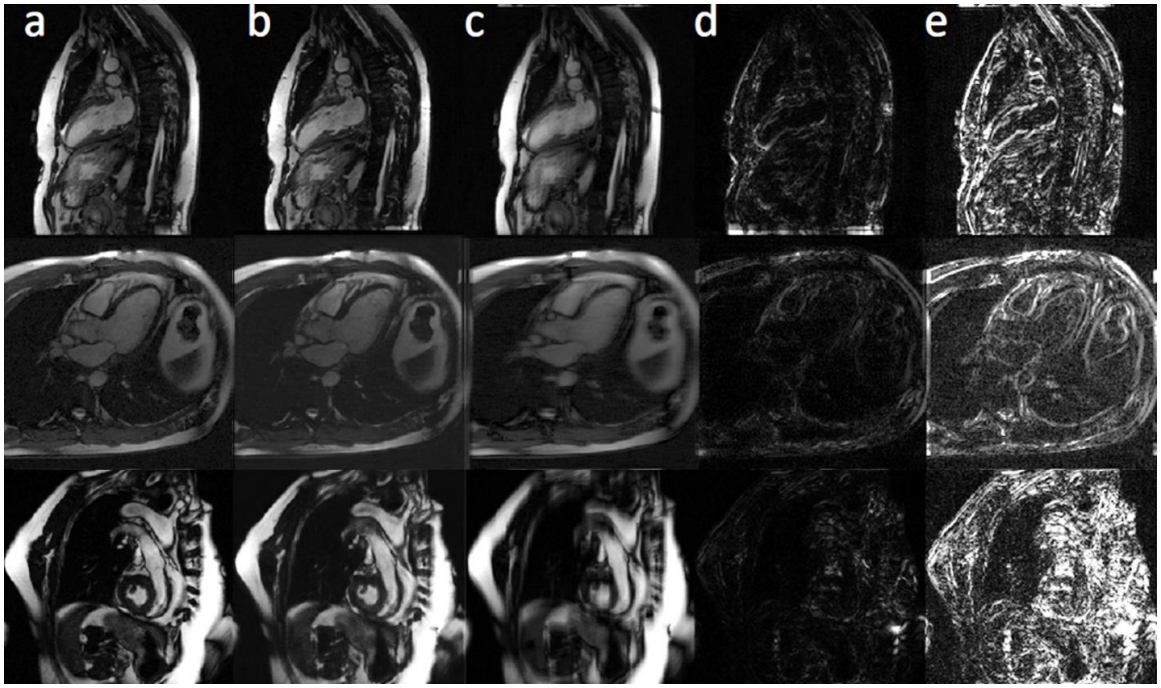

Figure 4-4. Motion accuracy simulation study results. Columns a, b, and c are the ground truth, motion-corrected, and synthetically motion-corrupted images. Absolute error map between ground truth and the motion-corrected/motion-corrupted images are shown in columns d and e, respectively. The first row shows an example for the vertical long-axis view, the second row presents a horizontal long-axis view, and the third row represents the short-axis view.

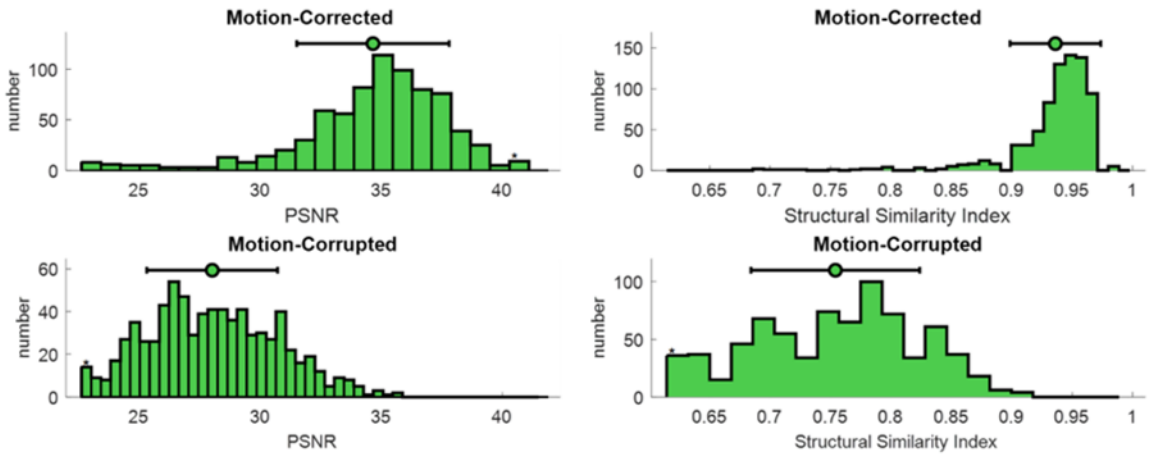

Figure 4-5. Quantitative simulation analysis. SSIM and PSNR, common metrics for image evaluation, were calculated for the simulated motion-corrupted data sets (bottom row) and motion-corrected images (top row). Both scores were reported by frequency plot and 95% of confidence interval. Mean values are shown with green circles; 95% of confidence intervals are depicted by black lines.

4.3.2 In vivo Study

After validating the proposed method's performance in correcting synthesized motion, the network was trained and tested based on in vivo motion-corrupted datasets. Throughout the training process, intermediate output in each epoch was exported to monitor the training process. Figure 4-6 shows improvement in quality of the output image through the training process. After 5 epochs, the outputs were blurry and had substantial artifacts, which would easily enable the discriminator network to classify unequivocally as fake images. However, as the training went on, the image quality of the encoder output was progressively improved. After 125 epochs, the quality of the images was sufficiently high for the discriminator to label them as real images.

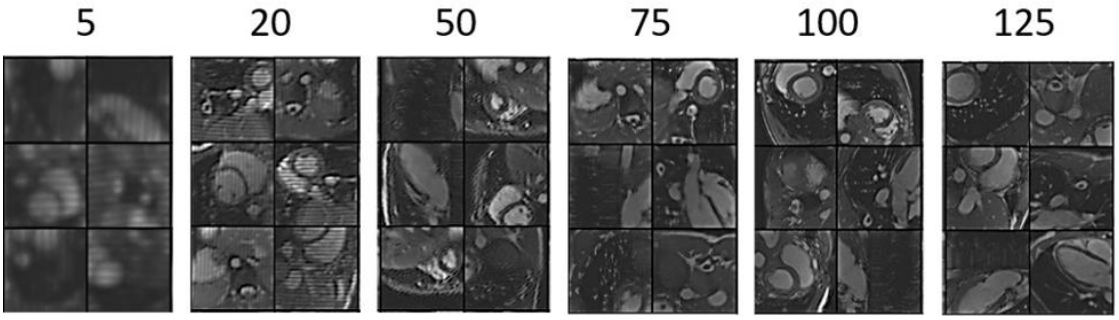

Figure 4-6. Image quality of the encoder output image with respect to the number of training epochs. As the training progresses, the image quality increases steadily.

Figure 4-7 shows representative images from two test volunteers' data. The motion-corrected image (column b) reduced the motion artifact from the motion-corrupted image (column c) and provided visually sharper images at the interventricular septum and better visualization of the heart and adjacent structures. Residual minor blurring still exists for smaller structures, however.

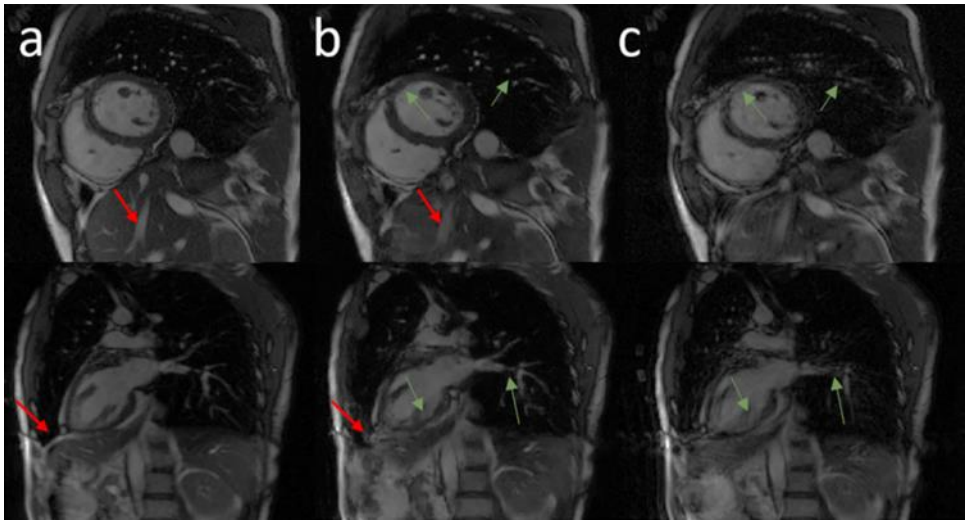

Figure 4-7. Representative images in the short-axis view, and vertical long-axis view from two volunteer subjects. Columns a, b, and c show the breath-held cine, motion-corrected free-breathing cine, and motion-corrupted free-breathing cine images, respectively. Green arrows highlight structures that were recovered completely by the network. Red arrows point to regions of residual blurring.

Figure 4-8 shows representative examples of breath-held cine (a), motion-corrected free-breathing cine (b), and motion-corrupted free-breathing (c) images from a patient who underwent a clinically indicated cardiac MRI exam. The network was able to eliminate the motion artifact seen at the left ventricular myocardium and the right ventricle. The motion-corrected images overall resemble the breath-held cine images.

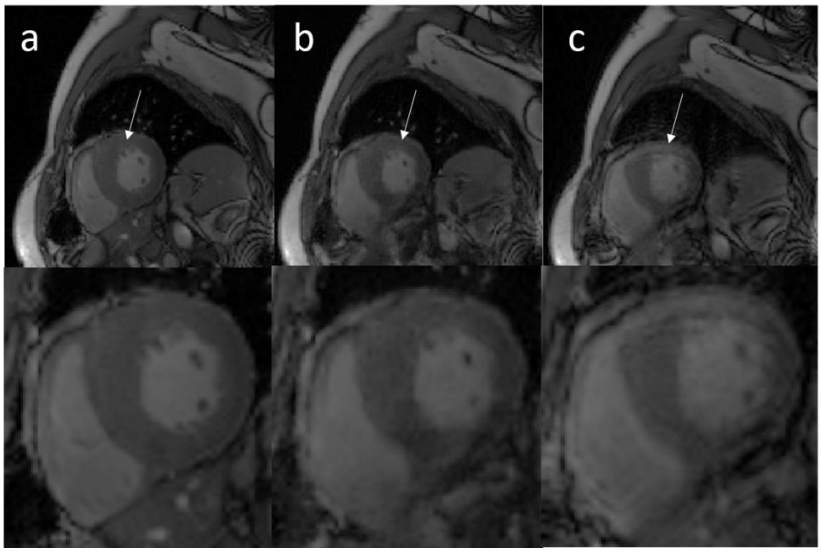

Figure 4-8. Representative cardiac cine images from a testing data acquired on a patient. Columns a, b, and c show standard clinical breath-held cine, the motion-corrected cine based on free-breathing data, and motion-corrupted cine data, respectively. White arrows show that the left ventricle region is significantly affected by motion artifacts and these artifacts were removed by the proposed network.

The normalized Tenengrad focus measure was 0.86±0.13 for the motion-corrupted images and 0.92±0.11 for the motion-corrected images, which represent a 7% increase using the proposed motion correction network.

Figure 4-9 shows Bland-Altman plots of the left ventricular SV, ESV, EDV, and LVEF for the cardiac functional analysis. The cardiac function parameters calculated based on our motion-corrected images were in good agreement with standard breath-hold images.

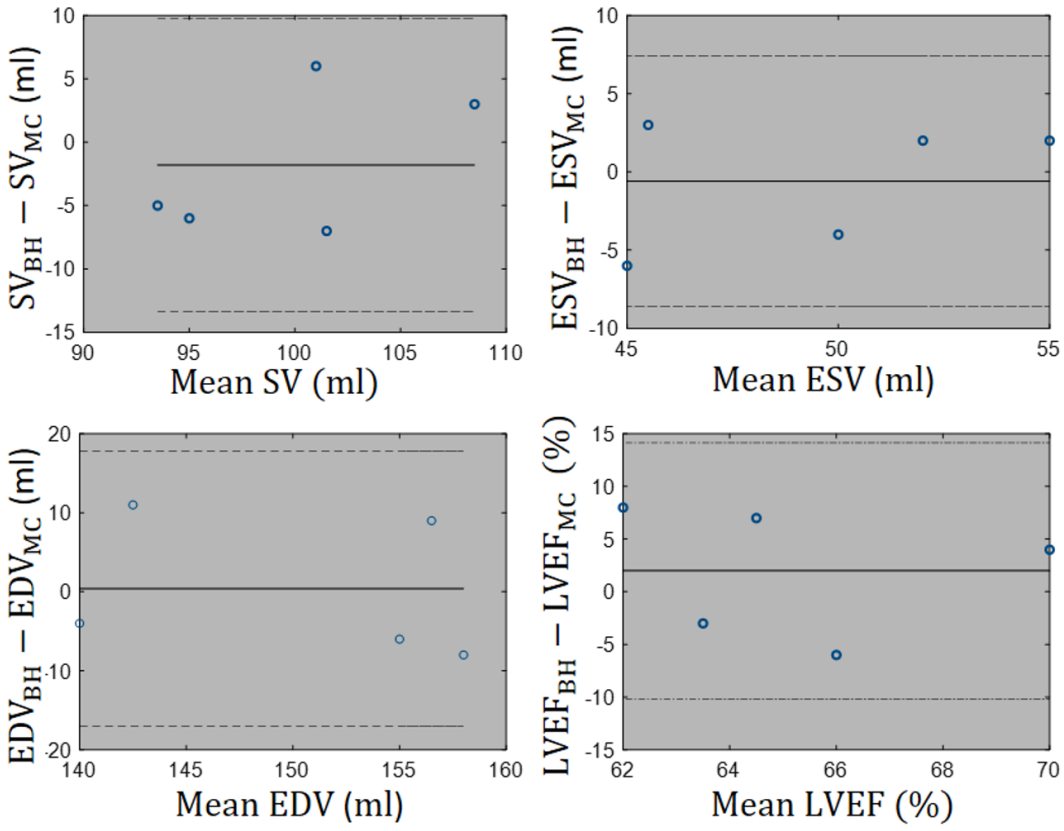

Figure 4-9. Functional analysis: Left ventricular endocardial borders are automatically segmented to compute stroke volume (SV), end-systolic volume (ESV), end-diastolic volume (EDV), and ejection fraction (LVEF) for 5 test cases. Bland-Altman plots confirm that there is agreement with 95% confidential

level between functional metrics measured from breath-hold free of the motion images and motion-corrected images.

Figure 4-10(a) summarizes image quality scores. To identify any statistically significant difference in the overall image quality of breath-held cine, motion-corrected, and motion-corrupted groups, the null hypothesis assumed that the rank distribution of groups is the same. The null hypothesis was rejected significantly ($P<0.05$) by applying Friedman's two-way analysis on the rank scores of different groups. Figure 4-10(b) reports paired comparisons between the mentioned groups. As can be seen, there was no statistically significant difference between the qualitative scores of motion-corrected and breath-held groups. Due to the statistically significant difference between the scores of motion-corrected and motion corrupted groups as highlighted by yellow edge, we conclude that the proposed method, increases the overall image quality of the motion-corrupted images

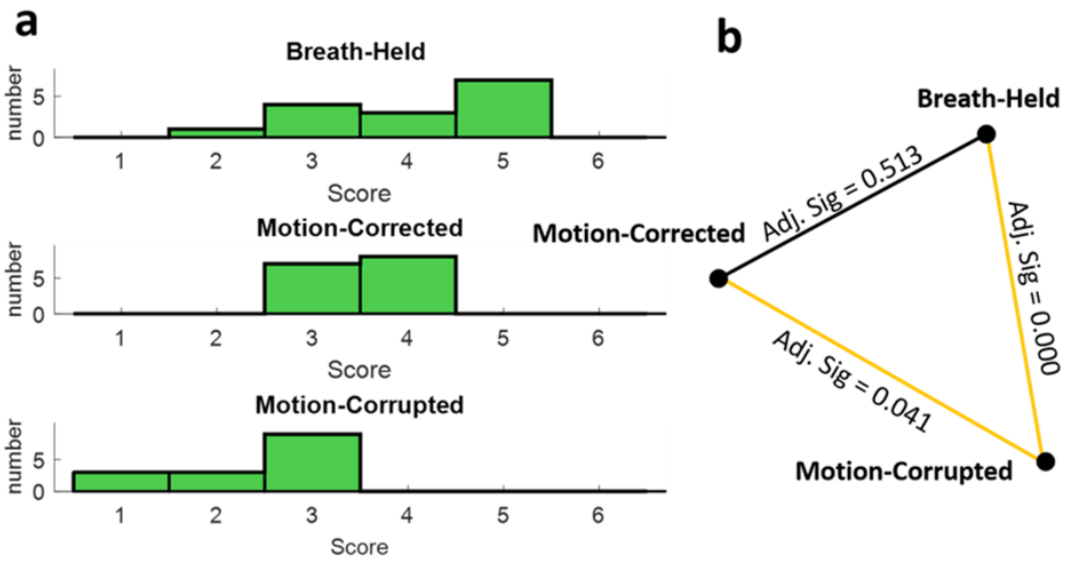

Figure 4-10. Blinded overall image quality reading and non-parametric paired comparison. (a) Frequency of overall image quality scores for each group. (b) Results from non-parametric paired comparisons. Statistically significant differences between pairs are highlighted by yellow lines.

4.4 Discussion

We proposed a deep learning-based method to reduce respiratory motion artifacts and tested it for free-breathing cardiac cine imaging. The proposed method was evaluated in terms of SSIM, PSNR, image sharpness score and subjective image quality. We showed that our adversarial autoencoder technique can effectively reduce or eliminate blurring and ghosting artifacts associated with respiratory motion while enabling free-breathing scanning. Using an adversarial autoencoder neural network in the proposed scheme has several advantages over the conditional generative adversarial networks for motion correction. First, the network does not require paired data because network training can be performed in a self-supervised manner. Second, image data consistency and anatomical accuracy was enforced implicitly in training process using an autoencoder network to ensure the motion-corrected image retains the important anatomical and structural content of the image. It is worth noting that the data consistency and anatomical accuracy may be enforced explicitly if the process of non-rigid motion-corruption is well-defined mathematically or if we had access to strictly paired motion-corrupted/motion-free data. An added benefit of our technique is its shorter total scan time than conventional breath-hold cine imaging. Conventional breath-hold cine imaging needs pauses between breath-holds, which increases the overall acquisition time. In our proposed method, the same cine imaging sequence can be run sequentially without any pauses. Therefore, the total cine scan time for our proposed method is less than the conventional breath-hold cine imaging.

In medical imaging applications, acquiring large amounts of paired data for motion correction tasks can be highly challenging and time consuming. Other approaches, such as conditional GANs, usually use L1 or perceptual loss functions for the generator network, which requires paired data to stabilize the training process. In our adversarial autoencoder, the autoencoder path preserves the

overall structural content accuracy, which is mandatory for medical imaging applications; while the adversarial path forces the encoder network to correct the motion artifacts in the images.

Typical motion-induced effects in MRI include blurring, ghosting, regional signal loss, and appearance of other unphysical signals[105]. Based on the quantitative sharpness analysis, the proposed method was able to increase the sharpness score in the simulation study by 12% and in the in-vivo study by 7%. It seems that there is a drop in the improvement of the sharpness score from the simulation study to the in-vivo study. It may be explained by considering the difference between the simulation and the in-vivo study. Realistically, motion corruption for Cartesian cine images under free breathing tends to cause more ghosting effect than blurring. Therefore, baseline normalized Tenengrad focus measure is expected to be higher for the real motion affected images than the simulated motion affected images, which was predominantly blurred by the simulated motion.

Two important concerns for our type of technique are: 1) whether the pathologies were preserved in the proposed motion correction network; 2) whether our adversarial-based network introduced new spurious anatomical features in the images. Based on our expert radiologist's evaluation of 3 test set images, we did not find any cases where any of these two scenarios occurred. However, we caution that larger scale evaluations in future studies are clearly warranted before a definitive conclusion can be made.

One of the innovations of this work with regard to the network architecture is that we used convolutional U-Nets for the Encoder and Decoder. In other commonly used autoencoders, the code space is often of smaller dimension/size compared to the input. However, for our application, the code space is the motion-corrected image and needs to have exactly the same size as the input

images. Therefore, both the encoder and decoder parts of the autoencoder need to be networks that produce an output that is of the same size as the input. Convolutional U-Net has this desirable property. We note that there are many other potential network structures that also have this property (input size = output size), residual networks and dense networks being just two examples. However, several nice characteristics of U-Net made it a suitable choice: 1) It covers a large receptive field without increasing the depth of the network. 2) It is able to extract the features in multi-scale levels of the resolution. 3) The dense connections between the different levels of the U-Net make its training process very stable and effective.

Several further enhancements of the network may help improve its performance. First, we did not exploit all available information in MR data. Exploiting the spatio-temporal correlations, multi-channel data as well as acquisition parameter details could increase the capability of the network to correct respiratory motion artifacts. Several conventional motion correction methods identify k-space data that are corrupted by motion, often by leveraging redundant k-space signal afforded by multiple receiver coils[106,107]. The proposed technique operates more in the image space. The input data are motion-corrupted images that have already been reconstructed from motion-corrupted k-space. Therefore, our approach is fundamentally different from the aforementioned methods. As is with many other types of deep neural networks, our technique cannot be mathematically fully understood in analytical forms. We speculate that our network relies on learning and recognizing the underlying patterns of motion artifacts that are typically present in a free-breathing scan in order to improve the image quality and remove motion contamination. We expect our approach can be fine-tuned to be applied in tandem after conventional motion correction methods are finished to remove any residual motion artifacts. Second, we focused on correcting motion under normal free breathing condition. The performance of our technique in the presence

of deeper than normal breathing remains to be evaluated. Using prior information about the characteristics of the motion may constrain the degrees of motion and correct the motion more effectively. For example, incorporating the respiratory bellows signal could afford us extra information about the motion-corrupted k-space lines. This extra information could enable us to incorporate the explicit data consistency term in the network, which could further improve the performance of our technique. Third, our platform is a 2D network, which performs correction in-plane. For through-plane motion, implementing a 3D adversarial autoencoder may be considered. Fourth, we did not take into account the differences in image FOV between the training data and the testing data when training the network. Therefore, motion correction capability of our network for arbitrary FOV should be investigated. Fifth, we did not compare our method with other free-breathing imaging methods, such as self-gated and real-time imaging. Such a comparison is clearly warranted in future studies.

Our study has limitations. It is possible that our technique might not entirely remove any motion-related artifacts in our image. Inspecting the supplementary videos S2-1 (available online as a supporting file of our published article[9]) showed that the endocardium in the end-systolic state is not as sharp as the breath-hold acquisition, indicating residual respiratory motion if the assumption of consistent cardiac motion between acquisitions held. However, these artifacts are minor, and motion artifacts are significantly reduced compared to the motion-corrupted images. Radiological assessments also aligned with this observation, where the respiratory-motion corrected images by our proposed method got a statistically non-significant lower score than the breath-hold cine images. A more extensive study with more clinical validation is needed to examine the proposed method on the patient cases' large cohort. For future work, two general directions may be considered. First, taking advantage of the redundant information in the channel

and temporal dimensions could further improve the current network's performance. Second, focusing on data augmentation may be very beneficial if we could realistically simulate in vivo motion patterns and their associated MR signal, which is currently challenging. In the absence of this, an alternate approach is to use unpaired high-quality data to train an adversarial autoencoder network.

4.5 Conclusion

In this study, we proposed an approach to reduce the respiratory motion artifact in cardiac imaging. Our approach enabled the free-breathing scan for cardiac cine imaging. We also showed that the quality of the images from the radiological point of view is acceptable for diagnosis purposes.

Chapter 5 Temporally Aware Volumetric GAN-based 4DMR Image Reconstruction and Respiratory Motion Compensation

This work aims to develop a 3D generative adversarial network to simultaneously compensate the respiratory motion and reconstruct the highly undersampled 3D dynamic cine images. In particular, our goal is to achieve high acceleration factors 10.7X-15.8X and maintaining robust and diagnostic image quality superior to state-of-the-art self-gating (SG) compressed sensing wavelet (CS-WV) reconstruction at lower acceleration factors 3.5X-7.9X. We trained a 3D GAN based on pixel-wise content losses, adversarial loss, and a novel data-driven temporal aware loss to preserve the anatomical accuracy and the temporal coherency. We proposed a novel training strategy that extends the progressive-growing training technique to make the training possible for our proposed GAN structure. We developed the proposed GAN and qualitatively and quantitatively evaluated its performance based on 3D cardiac cine data acquired from 42 patients with congenital heart disease. A version of this chapter has been published[10] in the Magnetic Resonance in Medicine:

1. Ghodrati V, Bydder M, Bedayat A, Prosper A, Yoshida T, Nguyen KL, Finn JP, Hu P. Temporally aware volumetric generative adversarial network-based MR image reconstruction with simultaneous respiratory motion compensation: Initial feasibility in 3D dynamic cine cardiac MRI. Magn Reson Med. 2021 Jul 13. doi: 10.1002/mrm.28912.

5.1 Introduction

Imaging acceleration and respiratory motion compensation remain two major challenges in MRI, particularly for cardiothoracic[108], abdominal[109] and pelvic MRI[110] applications. For image acceleration, parallel imaging[3,4,111] and compressed sensing (CS)[5] have enabled routine clinical MRI scans[112-115] from head to toe with robust acceleration rates of 4-6-fold (4X-6X). For respiratory motion compensation, numerous strategies have been extensively studied, including diaphragm navigators and various types of MR motion self-gating[112,116-120] based on repetitively acquired k-space center. However, variations and irregularities in each patient's breathing pattern [121] could compromise the accuracy and performance of these motion compensation methods; it often remains elusive why the same type of navigator or MR self-gating works on some patients but not on some others. In the past few years, motion-regularized methods have been proposed to reconstruct images for multiple motion states in a single optimization process[112,117,118,122,123]. These approaches exploited CS to incorporate prior information about the inherently low dimensional nature of the moving images using appropriate sparsity regularization in a transform domain such as finite difference, and Wavelet (WV). Although these state-of-the-art methods could reconstruct motion resolved images from significantly undersampled k-space data, they are computationally intensive. Moreover, motion-regularized methods rely on sparsity assumptions, which may not be able to pick up dataset-specific inherent latent structures[34,57].

Deep neural networks, particularly convolutional neural networks (CNNs) and generative adversarial networks (GANs), have shown promises for MRI image reconstruction[7,8,47-49,51,57,58,60,124-142] and motion correction[9,84,88,143,144]. 3D CNNs or 2D convolutional recurrent neural networks (CRNNs) have been proposed to exploit the spatiotemporal information in 2D dynamic MRI[49,126,134,136]. Qin et al. proposed a novel 2D-CRNN framework to reconstruct high quality 2D

cardiac MR images from undersampled k-space data (6X-11X) by exploiting the temporal redundancy and unrolling the traditional optimization algorithms[136]. Hauptmann et al. used a 3D convolutional U-Net to suppress the spatiotemporal artifacts in radial 2D dynamic imaging from undersampled data (13X)[134]. These methods effectively address flickering artifacts between temporal frames in 2D time-series images and provide improved reconstruction quality over conventional CS-based approaches. However, these methods are mainly trained based on pixel-wise objective functions, which is insensitive to the images' high-spatial-frequency texture details[8,57,145]. As the field moves toward high dimensional imaging, i.e. 4D (3D spatial + time) MRI acquisitions, the extension of these methods to 4D imaging is not straightforward[146], as it require volumetric CRNN or 4D convolutional neural networks, which may present substantial challenges in network training strategy and convergence.

GANs have been used to reconstruct images that provide similar or better visual image quality as standard reconstruction methods[57,58]. Mardani et al. showed impressive results of using 2D GAN in image reconstruction from under-sampled MRI datasets (5X-10X)[57]. In particular, they showed the 2D GAN-based reconstruction outperformed the CS-Wavelet (CS-WV) approach in their overall image quality evaluation of their volumetric abdominal images. However, 2D GAN cannot fully leverage the redundancy within volumetric images; hence, only limited acceleration factors can be achieved. Furthermore, 2D slice-by-slice approaches cannot preserve the through-slice coherence, such that flickering artifacts might be introduced to 3D images. Although the extension of 2D GAN to a 3D volumetric GAN can address the issue mentioned earlier, training a volumetric GAN represents a challenge and requires a sophisticated training process.

We propose a 3D GAN-based deep neural network and apply it to 4D cardiac MR image reconstruction and motion compensation. Our goal was to enable a high acceleration factors

10.7X-15.8X while maintaining robust and highly diagnostic image quality that is superior to state-of-the-art CS reconstructions at lower acceleration factors. Furthermore, respiratory motion compensation is achieved simultaneously in the image reconstruction pipeline, potentially enabling fully free-breathing 4D MR data acquisition and fast automated reconstruction of the data within minutes. To achieve our goals, we incorporated a specialized data-driven objective function that we dubbed temporally aware (TA) loss to regularize the output of the generator network in the volumetric GAN and to maintain coherence in the temporal dimension. Our network training procedure was inspired by the progressive growing strategy proposed by Karras et al.[96] to generate high-resolution images from noise vectors. We extended their strategy in a way that is applicable to the task of network-based image reconstruction from aliased, respiratory motion-corrupted images.

5.2 Theory

Figure 5-1 shows the overall view of the proposed temporally aware volumetric GAN (TAV-GAN). The TAV-GAN is a volumetric GAN trained based on the adversarial loss, pixel-wise content losses, and a novel TA loss. The TA loss was obtained from a separately trained ancillary temporal GAN. To train the TAV-GAN, we used paired 3D image patches from 3D highly-aliased (10.7X-14.2X acceleration) and respiratory motion-corrupted input images and from high-quality self-gated CS-WV reference images (2.8X-4.7X acceleration).

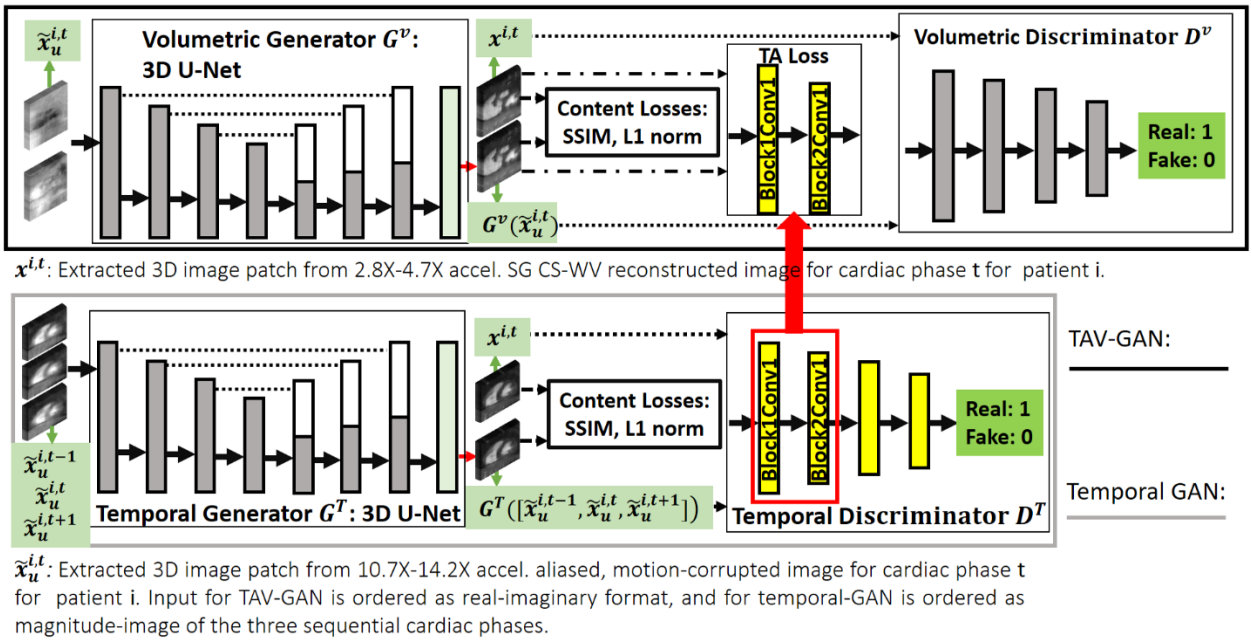

Figure 5-1. Overview of the proposed temporally aware volumetric GAN (TAV-GAN). The main component is a volumetric GAN (top). An ancillary temporal GAN (bottom), which is pre-trained, provides the temporally aware (TA) loss for the volumetric GAN training. Three objective functions, including content losses (SSIM, and L1), adversarial loss, and TA loss, are used to train the volumetric GAN. The role of the content loss is to compel the volumetric generator to produce anatomically correct images, and the role of the TA loss is to compel the volumetric generator to produce temporally coherent image. The TA loss is calculated based on L2 distance between features in two intermediate layers (Block 1 Conv 1 and Block 2 Conv 1) of the pre-trained temporal discriminator DT when the output of the volumetric generator Gv and the ground truth image volumes are separately input to DT. The temporal generator and discriminator take as input accelerated, aliased, and respiratory motion-corrupted magnitude 3D image patches from three consecutive temporal frames (t-1, t, and t+1), and produce an un-aliased, and respiratory motion-corrected 3D image patch for frame t.

As shown in Figure 5-1, both the volumetric and temporal GANs, which are two major components of the TAV-GAN, are both 3D networks. The difference between them is that the volumetric GAN was trained based on paired complex 3D image patches $\tilde{\boldsymbol{x}}_{\boldsymbol{u}}^{\boldsymbol{i,t}}$(input) and $\boldsymbol{x}^{\boldsymbol{i,t}}$ (target) extracted from the input and reference 3D images, while the temporal GAN was trained using three sequential magnitude-based concatenated 3D image patches $\tilde{\boldsymbol{x}}_{\boldsymbol{u}}^{\boldsymbol{i,t-1}}, \tilde{\boldsymbol{x}}_{\boldsymbol{u}}^{\boldsymbol{i,t}}, \tilde{\boldsymbol{x}}_{\boldsymbol{u}}^{\boldsymbol{i,t+1}}$ as the input.

Therefore, the temporal GAN is a 3D GAN-based network that exploits the spatiotemporal information in the dataset.

5.2.1 Volumetric GAN

A GAN is comprised of two neural networks, a generator network G, and a discriminator network D that are trained jointly in an adversarial manner. In the context of image reconstruction, the generator G attempts to generate images from a source data distribution and minimize the distance between the data distribution of the generated images and a reference data distribution. In contrast, the discriminator network D aims to estimate as accurately as possible the probability that a sample came from the reference data distribution[147].

In our technique, the detailed network architecture for the volumetric generator G^v and volumetric discriminator D^v is shown in Figure 5-2. Suppose $\tilde{X}_u = \{\tilde{x}_u^{i,t} \mid 1 \leq i \leq P,\ 1 \leq t \leq C\}$ is a set of highly-accelerated and respiratory motion-corrupted dynamic 3D image patches, and $X = \{x^{i,t} \mid 1 \leq i \leq P,\ 1 \leq t \leq C\}$ is a set of "clean" reference 3D image patches without respiratory motion artifacts or aliasing from k-space under-sampling, where P and C represent the number of patients and cardiac frames, respectively. We omitted the location index from the 3D image patches for clarity; for the rest of the manuscript, we consider $x^{i,t}$ and $\tilde{x}_u^{i,t}$ as the paired 3D image patches.

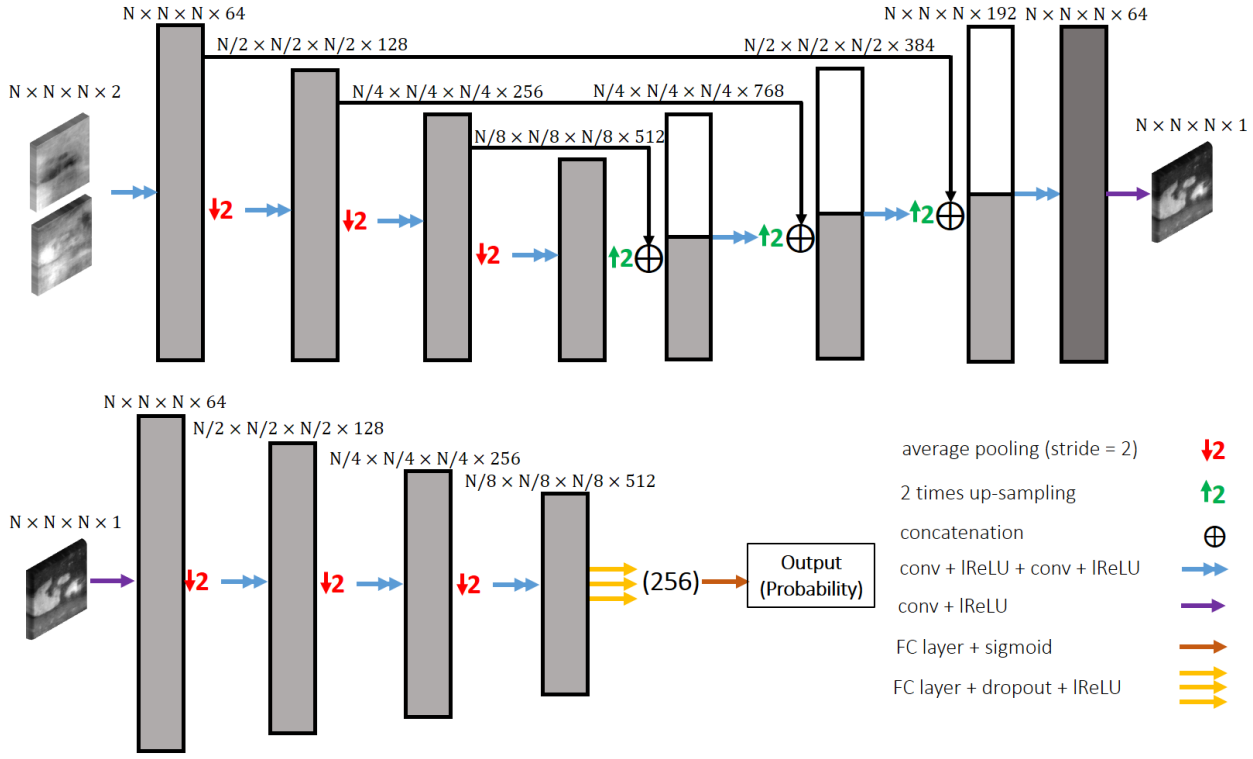

Figure 5-2. Detailed network structure for the volumetric generator and discriminator used in TAV-GAN. The generator is a 3D U-Net which consists of two paths: (I) the encoder path, which contains three downsampling blocks; (II) the decoder path, which includes three up-sampling blocks. Each block contains two convolutional layers, with each layer containing learnable convolution filters followed by Leaky ReLU (LReLU). Convolutional layers in the first block of the network contain 64 convolutional kernels, and the number of kernels doubles in each deeper block. Down-sampling and up-sampling blocks in the encoder and decoder paths are connected via average polling (strides = 2) and up-sampling (strides = 2). A skip connection is used to pass the data between each pair of same-sized up-sampling and down-sampling blocks. The discriminator is a binary classifier that contains three down-sampling operations followed by two convolutional layers in which each convolutional layer contains convolutional kernels followed by LReLU. The last two layers are the fully connected layer followed by dropout and LReLU, and a single decision fully connected layer with a sigmoid activation function. Discriminator takes the magnitude of the generated images to decide whether it is "generated" or "clean" images. The input and output of the generator for the Volumetric-GAN and temporally aware volumetric GAN (TAV-GAN) in the training phase are complexed-valued 3D image patches with size N×N×N×2 (real and imaginary), and magnitude-valued 3D image patches with size N×N×N×1, respectively. The input and output of the generator for the Temporal-GAN in the training phase are magnitude-valued 3D image patches with size N×N×N×3 (three sequential cardiac phases) and a magnitude-valued 3D image patch with size N×N×N×1, respectively. Due to the limitation of the GPU memory, N=64 is used in this work.

D^v is trained to distinguish between samples from the clean reference data set X and samples generated by G^v. The adversarial loss function for training D^v can be expressed as a sigmoid cross-entropy between an image x drawn from the reference set X and the generated image $G^v(\tilde{x}_u)$ where the image $\tilde{x}_u$ is drawn from set $\tilde{X}_u$:

$$\min_{\theta_{d^v}} L^a_{D^v}\left(D^v(x;\theta_{d^v}), G^v\left(\tilde{x}_u;\theta_{g^v}\right)\right) = \min_{\theta_{d^v}} \left[-\log D^v(x;\theta_{d^v})\right] + \left[-\log(1 - D^v\left(G^v\left(\tilde{x}_u;\theta_{g^v}\right);\theta_{d^v}\right)\right] \quad (5\text{-}1)$$

θ_{d^v}, θ_{g^v}, and $L^a_{D^v}(.)$ indicate the trainable parameters of D^v,G^v, and adversarial loss for the discriminator, respectively. On the contrary, G^v is trained to maximize the likelihood of the images generated by it being classified as a sample from the reference data set X. In theory, the negated discriminator loss $-L^a_D$ could be a proper loss function for training the generator G^v; however, from a practical standpoint, this approach suffers from a diminishing gradient issue. Hence, the adversarial loss for the generator network $L^a_{G^v}(.)$ is typically expressed as:

$$\min_{\theta_{g^v}} L^a_{G^v}\left(D^v(x;\theta_{d^v}), G^v\left(\tilde{x}_u;\theta_{g^v}\right)\right) = \min_{\theta_{g^v}} \left[-\log(D^v\left(G^v\left(\tilde{x}_u;\theta_{g^v}\right);\theta_{d^v}\right)\right] \quad (5\text{-}2)$$

Based on the loss functions described in Equations (5-1) and (5-2), training a GAN does not require paired data; the only requirement is the availability of two data sets: a reference data set X and an artifacted data set $\tilde{X}_u$ with large enough cardinality. Under a successful training process, the GAN would produce images with data distribution similar to the distribution of the reference image data set. However, there is no guarantee that the generated image $G^v(\tilde{x}_u^{i,t})$ will be matched anatomically with its corresponding clean reference image from the same patient $x^{i,t}$ [131]. To constrain the generator output, we added extra pixel-wise content loss functions to the objective function of G^v in addition to the adversarial loss specified in Equation (5-2). As shown in Equation

(5-3), the content loss is a linear combination of 3D structural similarity (SSIM) and normalized L1 norm in order to preserve local structural similarity and promote image spatial sparsity:

$$\min_{\theta_{g^v}} \lambda\left[\frac{1}{N}\left\|x^{i,t} - G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right\|_1\right] - \zeta\left[SSIM_{3D}\left(x^{i,t}, G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right)\right] \quad (5\text{-}3)$$

λ and ζ are the hyperparameters that control the amount of spatial sparsity and local patch wise similarity. N is the normalization factor and is equal to the number of the voxels in $x^{i,t}$. Detailed SSIM equation is provided in Appendix I. Even though the described objective functions in Equations (5-1), (5-2), and (5-3) can transform the under-sampled and respiratory motion-corrupted volumetric image data to clean aliasing-free and respiratory motion-artifact-free images, it cannot preserve the coherence in the temporal dimension, which in our experience often resulted in flickering artifacts between cardiac frames. Therefore, we propose to add a novel temporally aware (TA) loss function to the generator G^v to further improve performance.

5.2.2 Temporal GAN and TA loss

In TAV-GAN, an ancillary temporal GAN network is pre-trained such that its discriminator D^T can be used to achieve the TA loss for training the volumetric GAN. As shown in Figure 5-1, three sequential magnitude-only volumetric image patches $\tilde{x}_u^{i,t-1}$, $\tilde{x}_u^{i,t}$, and $\tilde{x}_u^{i,t+1}$ are stacked and input to the temporal generator G^T. Upon successful training, G^T produces an image $G^T([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}])$ that can be acceptable to the temporal discriminator D^T as a clean alias-free and respiratory motion-free image and has minimal pixel-wise content loss relative to its corresponding clean image $x^{i,t}$. The detailed network architecture for the temporal generator and temporal discriminator is similar to the network architectures shown in Figure 5-2, except the temporal generator trained based on the three sequential aliased, respiratory motion-corrupted 3D image patches as the input and the corresponding paired un-aliased, respiratory motion-corrected

3D image patch for the middle frame as the target. Equations 5-4 and 5-5 summarize the total loss function for the temporal generator and discriminator, respectively.

$$\min_{\theta_{g^T}} L_{G^T}^{Total}\left(x^{i,t},\ G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right) =$$

$$\min_{\theta_{g^T}} \gamma\left[\ L_{G^T}^{a}\left(D^T\left(x^{i,t}; \theta_{d^T}\right),\ G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right)\right] +\ \lambda\left[\frac{1}{N}\left\|x^{i,t} -\right.\right.$$

$$\left.\left. G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right\|_1\right] - \zeta\left[SSIM_{3D}\left(x^{i,t}, G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right)\right] \quad (5\text{-}4)$$

$$\min_{\theta_{d^T}} L_{D^T}^{Total}\left(D^T\left(x; \theta_{d^T}\right),\ G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right) =$$

$$\min_{\theta_{d^T}} \gamma\left[\ L_{D^T}^{a}\left(D^T\left(x; \theta_{d^T}\right),\ G^T\left([\tilde{x}_u^{i,t-1}, \tilde{x}_u^{i,t}, \tilde{x}_u^{i,t+1}]; \theta_{g^T}\right)\right)\right] \quad (5\text{-}5)$$

γ, λ and ζ are the weights of the adversarial loss, normalized L1 loss, and SSIM loss, respectively. Once the temporal GAN is trained, the temporal discriminator is detached and its intermediate feature space is used to calculate the TA loss for training the volumetric GAN as follows:

$$L_{G^v}^{TA}\left(x^{i,t},\ G^v\left(\tilde{x}_u^{i,t}; \theta_{g^v}\right)\right) = \frac{1}{N}\left[f_{b,c}\left\|D_{b,c}^T(x^{i,t}) - D_{b,c}^T\left(\ G^v\left(\tilde{x}_u^{i,t}; \theta_{g^v}\right)\right)\right\|_2^2\right] \quad (5\text{-}6)$$

where $L_G^{TA}(.)$ computes the normalized squared of the Euclidian distance in the feature space between its two given inputs. $D_{b,c}^T(x^{i,t})$ denotes extracted features from the cth convolution layer in the bth block of the temporal discriminator D^T. bth block includes all the layers in the discriminator after (b-1)th pooling operation and before bth pooling operation. $f_{b,c}$ weighs the squared of the normalized L2 norm of the features extracted from block number b and convolution number c. N is the normalization factor which is equal to the number of the voxels in the calculated volumetric features. Equations (5-7) and (5-8) summarize the total loss for the generator and the discriminator networks in the TAV-GAN.

$$\min_{\theta_{g^v}} L_{G^v}^{Total}\left(x^{i,t},\ G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right) = \min_{\theta_{g^v}} \gamma\left[L_{G^v}^{a}\left(D^v\left(x^{i,t};\theta_{d^v}\right),\ G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right)\right] +$$

$$\upsilon\left[L_{G^v}^{TA}\left(x^{i,t},\ G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right)\right] + \lambda\left[\frac{1}{N}\left\|x^{i,t} - G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right\|_1\right] -$$

$$\zeta\left[SSIM_{3D}\left(x^{i,t}, G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right)\right] \tag{5-7}$$

$$\min_{\theta_{d^v}} L_{D^v}^{Total}\left(D^v\left(x^{i,t};\theta_{d^v}\right),\ G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right) = \min_{\theta_{d^v}} \gamma\left[L_{D^v}^{a}\left(D^v\left(x^{i,t};\theta_{d^v}\right),\ G^v\left(\tilde{x}_u^{i,t};\theta_{g^v}\right)\right)\right] \tag{5-8}$$

γ , υ , λ and ζ are the weights of the adversarial loss, TA loss, normalized L1 loss, and SSIM loss, respectively.

5.3 Methods

5.3.1 Progressive TAV-GAN Training Strategy

Training GANs are inherently challenging in particular for high-dimensional images. In the absence of substantial overlap between the training data distribution (clean) and the generated data distribution (output of the generator), the gradients calculated in the back propagation process can point to more or less random directions, which may present substantial challenges in the training process[95-97]. A variety of strategies have been proposed to stabilize the training process of GANs for the image-to-image translation tasks, such as the Markovian-patch-based approach[98], Wasserstein Generative Adversarial Networks (wGANs)[148], and least-squares GANs (LS-GANs)[149]. LS-GANs and wGANs are popular training approaches and are effective in training GANs based on medium sized 2D-images of 128×128 or 64×64. In practice, the extension of them to higher dimensions or larger image sizes is not straightforward and requires some ad-hoc methods such as initialization of the generator's trainable weights before the training process. For instance, Mardani et al. stabilized the LS-GAN for MR image with size 320×256 by using pure L1 norm at the beginning of the training and gradually switch to the adversarial loss[57]. A more recent

approach proposed by Karras et al. showed surprising results in generating the high-resolution image from noise vectors[96]. Their training methodology is based on starting the training with generating the low-resolution images and progressively increasing the resolutions by adding layers to the network. In this work, we adopted such a progressive training method when training the TAV-GAN.

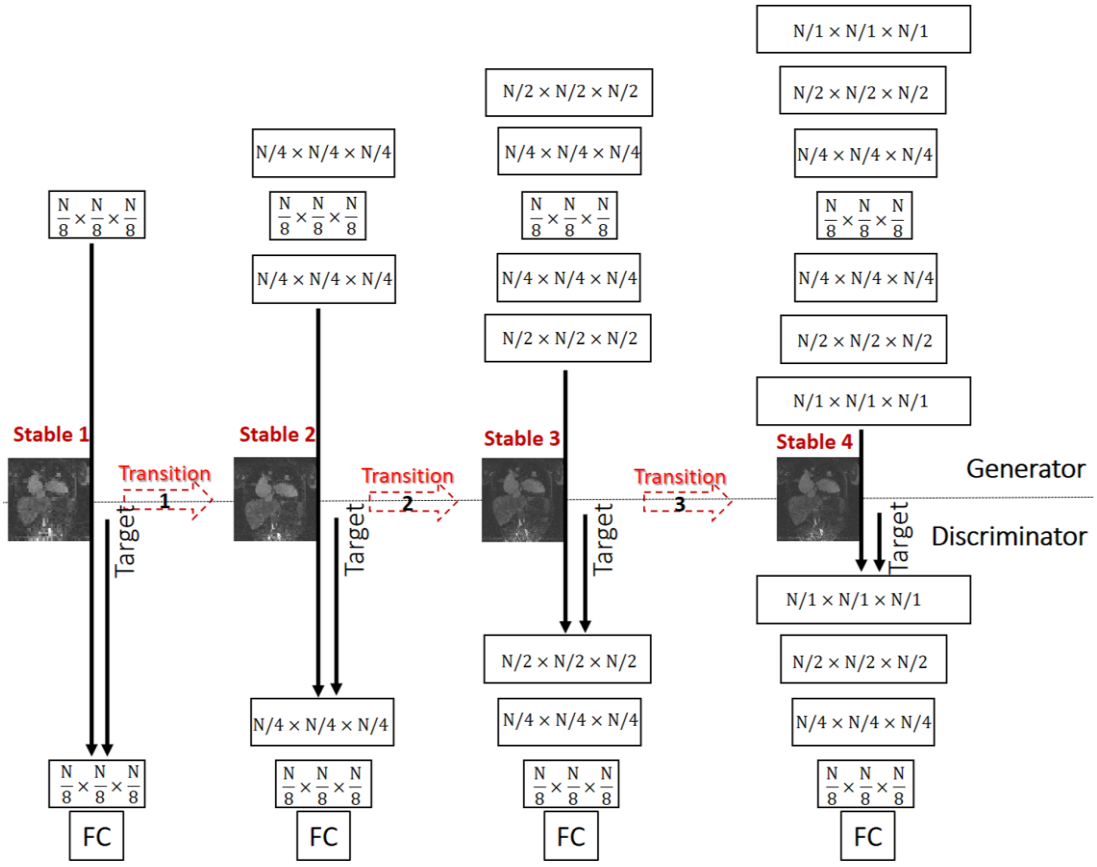

Figure 5-3. Progressive training strategy for the TAV-GAN. As training of GAN for low-resolution images is in general easier than high-resolution images, in our progressive training strategy, we initiate the training with the low-resolution layer of the generator and discriminator networks that handles N/8×N/8×N/8 image volume size, and gradually expand the network to reach the higher-resolution layers. For the sake of clarity, only the first three dimensions (spatial dimensions) of the features for the network layers are shown and skip connections in the generator network are not shown. The progressive training process consists of a chain of stable and transition phases. The first stable phase (Stable 1) is started by training the lowest-resolution layers, and in the transition phase, new layers are added and gradually mixed with old layers to reach the second stable phase where the resolution of the layers is doubled in each spatial dimension. This process is continued until the main resolution (N=64, 64×64×64) is reached. This training strategy enables us to have a stable GAN training process for high dimensional image reconstruction tasks.

Our proposed strategy for training the TAV-GAN is shown in Figure 5-3. The training process consisted of four stable phases and three transition phases that were interleaved. The training started with the first stable phase in which only layers with the lowest resolution level are built and trained for an epoch. Subsequently, in each of the transition phases, the new layers (with weight 1-α) were added to the existing layers (with weight α) of the generator and discriminator. The parameter α was linearly decreased from 1 to 0 through the iterations of an epoch. For instance, from the beginning of the transition phase (α=1), the newly added layers had zero weight, and as α decreases, the new layers had more weight until the part of the existing layers were faded (α=0). Once α reached 0, the transition phase was finished, and the next stable phase was started. These stable and transition phases were alternated while more layers were added progressively until the stable phase 4 was finished, which concluded the training process. Figure 5-4 shows more details of the first stable and transition phases for the TAV-GAN. The number of required training epochs was decided based on the quality of the test outputs and equilibrium state of the adversarial loss for the generator and the discriminator. The training process for the TAV-GAN in the first three stable and transition phases used the loss functions in Equations (5-7) and (5-8) with parameters $\gamma = 1, \upsilon = 0, \lambda = 0.5, \zeta = 0.3$. It means in the first three stable and transition phases, TA loss was not considered in the training process. In the last stable phase, TA loss was turned on with $\upsilon = 0.5$ ($f_{1,1} = 0.7, f_{2,1} = 0.3$).

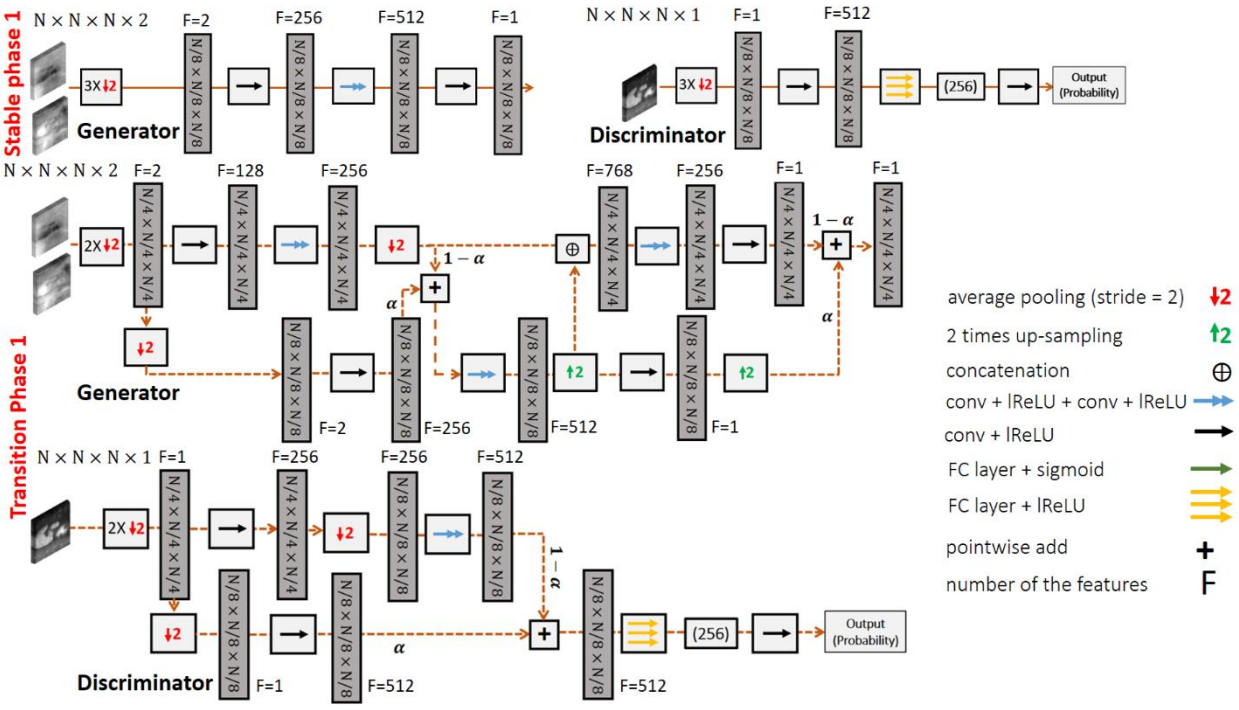

Figure 5-4. An example of the stable and transition phases of TAV-GAN training: In stable phase 1, the generator and the discriminator are built for the lowest resolution. The input for the network is down-sampled three times to match the lower resolution, and subsequently, it is entered into a convolution layer to increase its features from 2 to 256. Those features are then entered into two sequential convolutional layers that are the main layers of the 3D U-Net for the lowest resolution. Afterwards, the output is entered into another convolution to combine the 512 features to 1 feature. The role of the first and the last convolutional layers is to create proper number of features. The Discriminator also has fewer layers, similar to the generator in the first stable phase. Low-resolution image volume is entered into a convolutional layer to increase the number of features to match the required input size for the fully connected layers. After an epoch of training the first stable phase, the network is grown gradually through a transition phase. As seen in the first transition phase, some convolutional layers with doubled-resolution are added to the generator from the left and right sides. Besides, some convolutional layers also added to the discriminator from its left side. This addition is a pairwise gradual addition, which is controlled by parameter α, which linearly decreases from 1 to 0 through the total number of mini-batch iterations of an epoch. The first transition phase is started by α=1 (stable phase 1), and once α reached 0, the second stable phase is started. The growth process will continue until reaching the main resolution and building the main network structure shown in Figure 5-2. In our work N=64 was used.

5.3.2 Comparison Study

The proposed TAV-GAN was compared with four other networks, including 2D GAN, 3D U-Net, Volumetric-GAN only and Temporal-GAN only. For the 2D GAN, we used a 2D U-Net with four down-sampling and four up-sampling blocks for the 2D generator and a simple 2D binary

classifier with four down-sampling blocks as the discriminator. Appendix II includes further details of the network structure for the 2D GAN. The 3D U-Net approach is essentially the volumetric generator portion of the TAV-GAN shown in Figure 5-2, which has three down-sampling and three up-sampling blocks. In addition, to demonstrate the benefits of the TAV-GAN, we also compared our TAV-GAN with the Volumetric-GAN alone (Fig. 5-1, top panel without TA loss) and the Temporal-GAN alone (Fig. 5-1, bottom panel). The aforementioned progressive training strategy for the TAV-GAN was applied to training the Temporal-GAN and the Volumetric-GAN as well. A similar training strategy was adjusted for the 2D GAN, detailed in Appendix II.

For both the Volumetric-GAN and the Temporal-GAN approaches, we used a combination of two loss functions including the content loss L_G^c ($\lambda = 0.5, \zeta = 0.3$), and adversarial loss L_G^a ($\gamma = 1$). For the 3D U-Net, only content loss L_G^c ($\lambda = 1, \zeta = 0.1$) was used. The loss function for the 2D GAN is detailed in Appendix II. Weights of the loss functions were determined empirically with a limited number of searches.

For the TAV-GAN, Temporal-GAN, and Volumetric-GAN, the Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 16, an initial learning rate 0.0001 for the generator, and an initial learning rate 0.00001 for the discriminator. For the 3D U-Net, the Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 16, an initial learning rate 0.0001. Weights for all networks were initiated with random normal distributions with a variance of σ = 0.01 and mean μ=0. Optimizer parameters for the 2D GAN is reported in Appendix II. The training was performed with the Pytorch interface on a commercially available graphics processing unit (GPU) (NVIDIA Titan RTX, 24GB RAM).

Once the 3D networks were trained, they were tested based on the full-sized 3D image rather than 3D image patches. As the 3D U-Net and the generator part of the 3D GANs, including Temporal-GAN, Volumetric-GAN, and TAV-GAN, have 3 down-sampling stages, we padded the test input volume to the next size divisible by 8 before they were input to the network. For the 2D GAN, testing was performed based on the full-sized 2D image. As the generator part of the 2D GAN has 4 down-sampling stages, the size of the padded full-sized 2D image was divisible by 16 before inputting to the network. The testing was performed based on a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz).

5.3.3 Datasets

We used 3D cine cardiac MR data acquired previously for training, validating and testing the TAV-GAN technique, as well as for comparing with the other four networks. These data were acquired either as part of a separate research study or as part of the patient's clinically indicated MRI. The data were acquired on a 3T scanner (Magnetom TIM Trio, Siemens Medical Solutions) on 42 separate patients (age range 2 days-60 years, 84% were pediatric congenital heart disease (CHD) patients) using a previously described ROCK-MUSIC technique[116]. The images were acquired during the steady state intravascular distribution of ferumoxytol, which is used clinically at our center as an off-label MRI contrast agent. Except for two patients, all patients were scanned under general anesthesia according to our institutional clinical protocol. All raw imaging data were obtained under a research protocol approved by our institutional review board. The ROCK-MUSIC technique is a gradient-recalled-echo pulse sequence, which allows variable density data sampling and retrospective motion binning[116]. The ROCK-MUSIC data were acquired with the following sequence parameters: TE/TR = 1.2ms/2.9ms, matrix size ≈ 480×330×180, 0.8-1.1 mm^3 isotropic resolution, total acquisition time = 4.35-9 min, FA=20^o. Due to the fast nature of the patients'

hemodynamics in this study (heart rate ~ 120-180 beats/min), the acquired data were binned into 9-12 cardiac phases for the end-expiration state of the respiratory cycle. By including under-sampling from partial Fourier, the average net k-space under-sampling factor after cardiac phase binning and before end-expiration motion self-gating varied from 2.8X to 7.9X.

N_L dynamically acquired k-space lines were sorted retrospectively using self-gating signal into multiple cardiac phases for the breathing cycle's expiration state. Then the CS-WV with temporal total variation regularizer was used to reconstruct the reference 4D images (X). To reconstruct the highly-accelerated and respiratory motion-corrupted 4D images ($\widetilde{X}_u$), we first sorted the first acquired M=min(50000, N_L/2) k-space lines into only multiple cardiac phases, and then $\widetilde{X}_u$ obtained by inverse Fourier zero-filled reconstruction. Details for generating these data are provided in Appendix III. These patients are divided into three groups including one training dataset (A) and two testing datasets (B1 and B2) based on the quality of the reference data (X) and the total number of acquired k-space lines N_L, i.e., acceleration factor before self-gating:

Group A: Training Set. This dataset included 4D images from 12 patients. The total acquisition time for each of these data sets was 7.25-9 min (150000<N_L<187000 lines). The data in Group A was chosen due to their high overall image quality with minimal temporal artifacts based on visual assessment.

Group B1: Mild Test Set. This dataset included 4D images from 10 patients. The total acquisition time for each of these data sets was 5.8-7.25 min (120000 lines <N_L <150000 lines). Images in Group B1 had slightly lower visual image quality and noisier than Group A.

Group B2: Severe Test Set. This dataset included 4D images from 20 patients. The total acquisition time for each data was 4.35-5.8 min (90000 lines $<N_L<$ 120000 lines). This Group had lower visual image quality and significant temporal artifacts compared with Group B1. Representative image examples for each Group are shown in Figure 5.5.

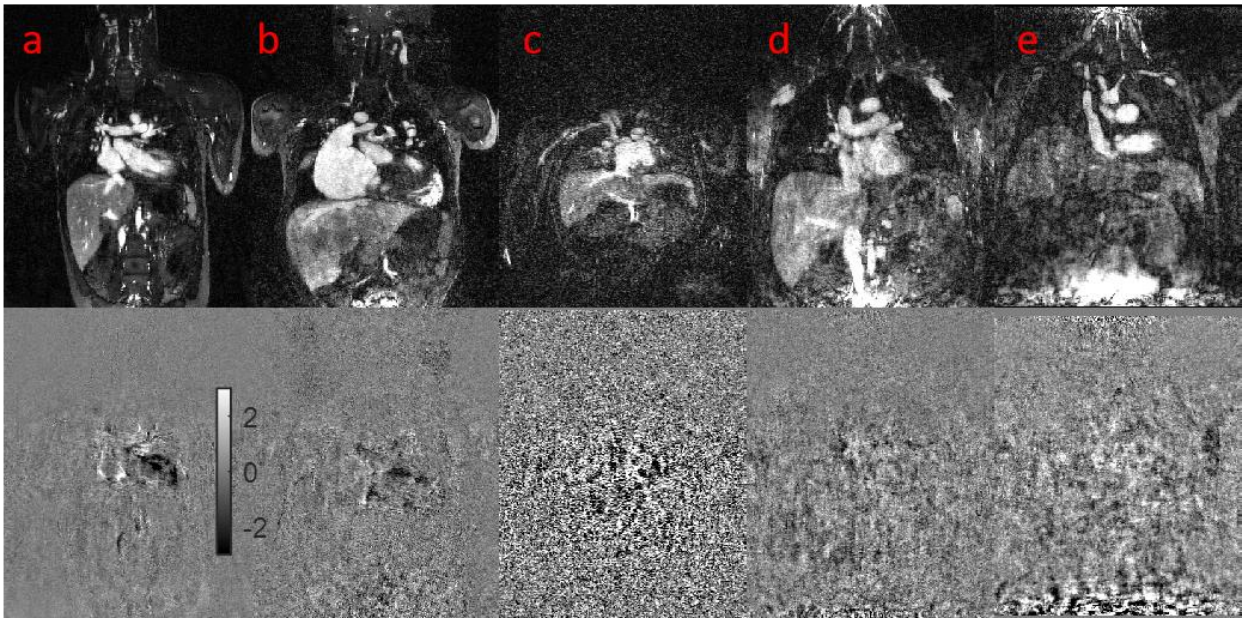

Figure 5-5. Representative examples for the datasets: columns (a), (b), (c-e) represent qualitative examples of the images from the dataset A (training dataset), dataset B1 (mild testing dataset), and dataset B2 (severe testing dataset), respectively. The first row shows the magnitude of a slice from the volumetric images, and the second row shows the difference map between two sequential cardiac phases. As can be seen in (a), it has the lowest noise and flickering artifacts through the cardiac phases among the others. The image in the column (b) has relatively higher noise and flickering artifacts through the cardiac phases than the image in column (a). Based on the calculation of the noise inside a 15×15×15 cubic region from the background, images in the datasets B1 (mean of the standard deviation = 0.076) are 2 times noisier than the images in the datasets A (mean of the standard deviation = 0.038). Column (c) presents image that was profoundly affected by noise. Approximately, the noise level for noisy images in datasets B2 (mean of the standard deviation = 0.304) based on the calculation of the noise inside a 15×15×15 cubic region from the background is, on average, 8 times the images in datasets A. Column (d) shows an image from a CHD patient with breathing irregularities scanned under anesthesia. As shown in column (d), image quality is degraded due to the respiratory motion artifacts. The image in column (e) shows an image from a CHD patient scanned under free-breathing without anesthesia. As shown in column (e), the quality of the image is degraded substantially due to the respiratory artifact and breathing irregularities.

5.3.4 Evaluations

To demonstrate the performance of the TAV-GAN, we performed the following evaluations:

a) Qualitative and quantitative analysis. We trained five different networks, including 2D GAN, 3D U-Net, Volumetric-GAN, Temporal-GAN, and TAV-GAN, using the data from Group A. After network training, we compared them qualitatively and quantitatively against SG CS-WV reconstructions using data in Groups B1 and B2, which were unseen by the networks. SSIM and normalized root mean squared error (nRMSE) were computed based on the cropped cardiac region of each cardiac phase, and the average of all phases was reported for each patient. To compare the sharpness of the results obtained by different networks, the normalized Tenengrad focus measure[100,101] was reported. The sharpness analysis was detailed in Appendix IV. It is important to emphasize that all quantitative analysis was performed on the data from Group B1 only due to its higher reference image quality compared to Group B2.

b) Subjective image quality assessments. Subjective image quality assessments were performed in two stages. In the first stage, movies of 4D images reconstructed with the five different networks based on the "highly-accelerated" test data from all cases in Group B2 and Group B1. The reference image reconstructed using CS-WV were also included, resulting six reconstructed 4D images per patient. The six 4D images were presented in random order to two experienced radiologists blinded to patient information or reconstruction technique, and each radiologist was asked to choose three top rated 4D images out of the six with regard to general image quality. The three top rated techniques were assigned a score of 1 and the remaining scored 0. Based on statistical paired comparisons on the mean scores among all test data sets, the top three techniques were selected for the second stage, where two radiologists performed more detailed and blinded evaluation of randomized 4D images from the three selected techniques with respect to overall

image quality and image artifacts, using a 1-5 grading system with 5= excellent quality, and 1= poor quality. Mean (±SD) of the general image quality and artifact scores were calculated for each technique. Besides, the radiologists evaluated the reconstructed images with regard to presence of any spurious feature that may be introduced to the images due to the generative nature and potential hallucination effects of the GAN-based networks. An image was labeled "spurious feature present" if either of the two radiologists identified any spurious feature in the image.

c) Cardiac function analysis. We selected six cases from the Group B2 which had the highest overall image quality score (≥3.5) for the reference SG CS-WV technique to perform cardiac function analysis and comparison. An expert evaluator in imaging CHD patients contoured the studies to determine left/right ventricular end-diastolic volume (EDV), end-systolic volume (ESV), stroke volume (SV), and ejection fraction (EF). The cardiac function analysis was performed for the best technique, which was determined based on the subjective image quality assessments, and were repeated for the reference CS-WV images acquired on the same six patients.

Paired comparisons were performed using Tukey HSD[150] to test for statistically significant differences between the two methods. A P-value of 0.05 was considered statistically significant.

5.4 Result

Figure 5-6 shows representative image reconstruction and respiratory motion correction results using the 6 techniques compared in this study. This unseen mild test case data was from Group B1. The four 3D networks (TAV-GAN, Temporal-GAN, Volumetric-GAN, 3D U-Net) had better performance in removing aliasing and respiratory motion artifacts than the 2D GAN, which had significant residual artifacts. The GAN based networks (TAV-GAN, Temporal-GAN, Volumetric-GAN) produce sharper images than the 3D U-Net. The temporal difference map shows that the TAV-GAN, Temporal-GAN, and SG CS-WV had the lowest incoherence between the

cardiac phases, as evidenced by the smaller signal differences between two successive cardiac phases (Fig. 5.6, row d). Video S1(available online as a supporting file of our published article[10]) presents complete 4D images for an additional example from Group B1.

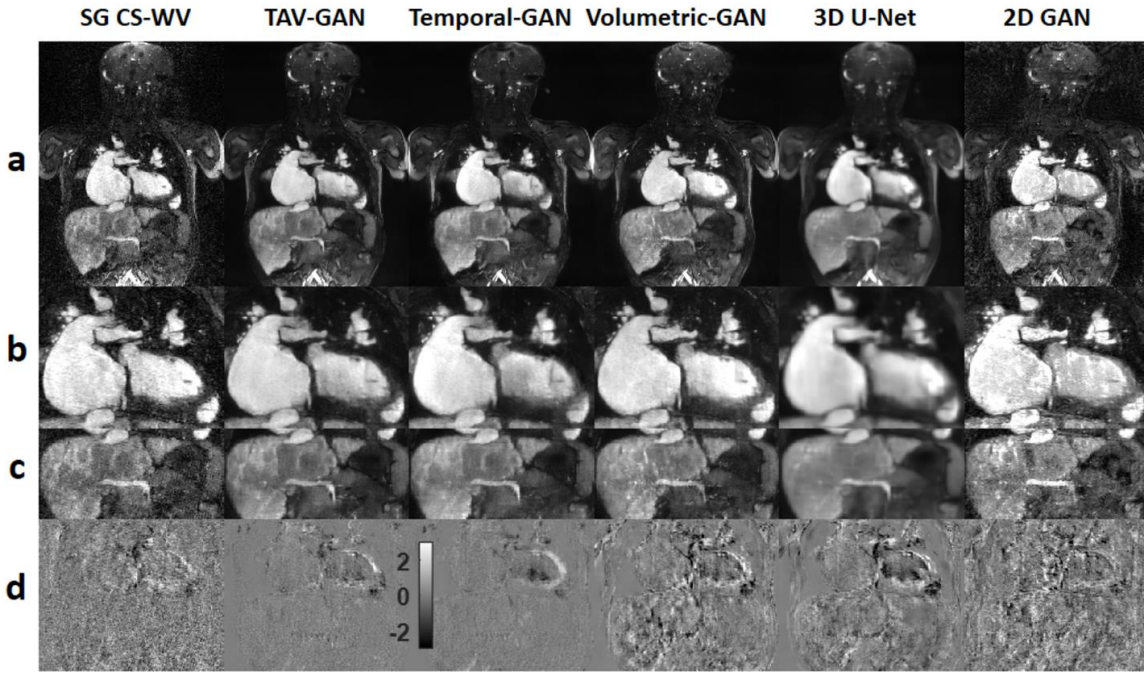

Figure 5-6. Qualitative comparison between different image reconstruction methods for a male CHD patient from test dataset B1 (6 y.o. and 18 kg weight) who was scanned under anesthesia. Row (a) shows the reconstruction/respiratory motion correction results and rows (b) and (c) show the zoomed view of the cardiac and liver region. Row (d) shows the temporal difference between 5th and 6th cardiac phases. The 2D GAN image has substantial residual artifacts. The 3D U-Net image is blurrier than the GAN based methods (TAV-GAN, Temporal-GAN, and Volumetric-GAN). As shown in (d), reconstruction results from TAV-GAN and Temporal-GAN have the lowest incoherency and flickering artifacts, which implies that the proposed TA loss can effectively decrease the temporal incoherency through the cardiac frames. The SG CS-WV was reconstructed based on 5.4X fold under-sampled data; the remaining methods shown were reconstructed based on 14.2X fold under-sampled data.

Figure 5-7 shows representative example results for a patient in Group B2, whose data was heavily affected by noise. The 3D U-Net image was blurrier than the other methods. The Temporal-GAN image was relatively blurrier than TAV-GAN and Volumetric-GAN. Reference SG CS-WV suffered from the residual noise and achieved overall image quality score 3 and artifact score 3 which is inferior than the TAV-GAN (overall image quality score = 4.5, artifact score = 4)

and Temporal-GAN (overall image quality = 4, artifact score = 3.5). The TAV-GAN and the Temporal-GAN had the highest coherency between the cardiac frames, as shown in Figure 5-7(d).

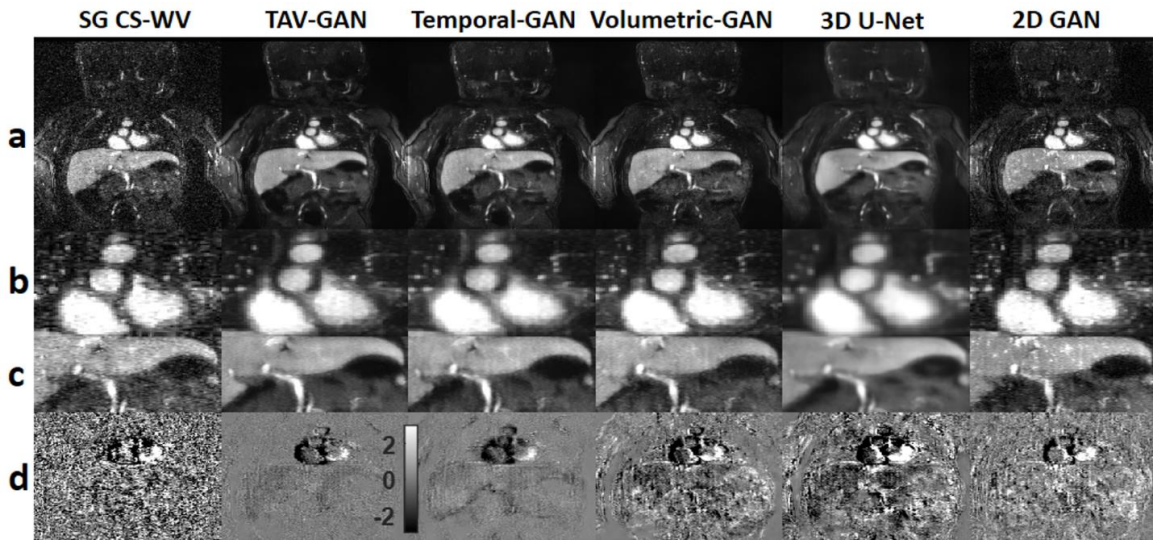

Figure 5-7. Qualitative comparison between different methods for a pediatric male patient from test dataset B2 (1 month old and 3.18 kg weight) who was scanned under anesthesia. Rows (a), (b), and (c) show the image reconstruction using 6 different methods and the zoomed view of the cardiac and liver regions. Row (d) shows the temporal difference between 2nd and 3rd cardiac phases. The 2D GAN image provides the most inferior image quality. The 3D U-Net image was blurrier than the GAN based methods (TAV-GAN, Temporal-GAN, and Volumetric-GAN). The Temporal-GAN image is slightly blurrier than the TAV-GAN and Volumetric-GAN. The reference SG CS-WV image suffers from the residual noise and its quality is inferior to the TAV-GAN and the Temporal-GAN. The SG CS-WV was reconstructed based on 5.7X fold under-sampled data; the remaining methods shown were reconstructed based on 11.4X fold under-sampled data.

Table 5.1 reports the SSIM and nRMSE for the reconstructed results of the different methods tested based on the Group B1. SSIM, and nRMSE were reported based on the Group B1 only because this patient group had high quality reference images. Although TAV-GAN achieved higher SSIM and the lower nRMSE than the other methods, it was only statistically significantly better than the zero-filled reconstruction and the 2D GAN approach. Based on the multiple pair comparison tests, it can be concluded that 3D based approaches are significantly better in terms of quantitative scores, nRMSE and SSIM, than 2D GAN approach.

Table 5.1. Quantitative evaluation: SSIM3D and nRMSE are calculated on reconstructed results from all patients (N=10) in test dataset B1 and mean and standard deviation (Std. Deviation) of them over the patients are reported for different methods. Based on the multiple pair comparisons, there is a statistically significant difference (P<0.05) between the SSIM and nRMSE metrics of the 2D-GAN reconstruction images and other methods. The proposed method (TAV-GAN) achieved the highest SSIM and the lowest nRMSE among the other methods.

Methods	SSIM		nRMSE	
	Mean	Std. Deviation	Mean	Std. Deviation
ZF	0.376S1	0.0446	0.094*	0.0194
2D-GAN	0.481S2	0.0594	0.072**	0.0138
3D U-Net	0.732	0.0483	0.040	0.0085
Volumetric-GAN	0.752	0.0479	0.038	0.0090
Temporal-GAN	0.746	0.0495	0.036	0.0072
TAV-GAN	0.785	0.0389	0.030	0.0058

* There was a statistically significant difference (P<0.05) between the ZF method and other methods with respect to the quantitative metrics SSIM and nRMSE.

** There was a statistically significant difference (P<0.05) between the 2D-GAN method and other methods with respect to the quantitative metrics SSIM and nRMSE.

The mean of the normalized Tenengrad focus measure (±SD) was 0.822±0.1015, 0.828±0.1390, 0.702±0.1408, and 0.286±0.0377, for the reconstructed, respiratory motion-corrected results obtained by TAV-GAN, Volumetric-GAN, Temporal-GAN, and 3D U-Net, respectively. Multiple pair comparison tests shows that the 3D GAN based approaches including TAV-GAN, Volumetric-GAN, and Temporal-GAN produced significantly sharper images than

the 3D U-Net. In this analysis, 2D GAN is excluded mainly because of the sensitivity of the Tenengrad focus measure to the residual artifacts.

Using a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz), the reconstruction time was approximately 6 sec/cardiac phase for the TAV-GAN, and 312 sec/cardiac phase for SG CS-WV.

Figure 5-8 shows representative reconstruction example for a patient in Group B2, who had irregular breathing and low baseline image quality. The reconstructed image from a 6.5X undersampled data (N_L=110000) by SG CS-WV had lower image quality than the reconstructed image by TAV-GAN from a 14.2X undersampled data. The small branch of the vessels in the liver (purple arrow, row d), soft tissue (blue arrow, row c), and the myocardium border (red-arrow, row b) were depicted well using TAV-GAN in comparison to the other methods. Subjective image quality scores for this case show that the TAV-GAN method with overall image quality score 4.5 and artifact score 4 has superior image quality than the Temporal-GAN (overall image quality = 3.5, artifact score = 3.5) and SG CS-WV (overall image quality = 2.5, artifact score = 3). Complete 4D images for Figure 5-8 is provided in Video S2 (available online as a supporting file of our published article[10]).

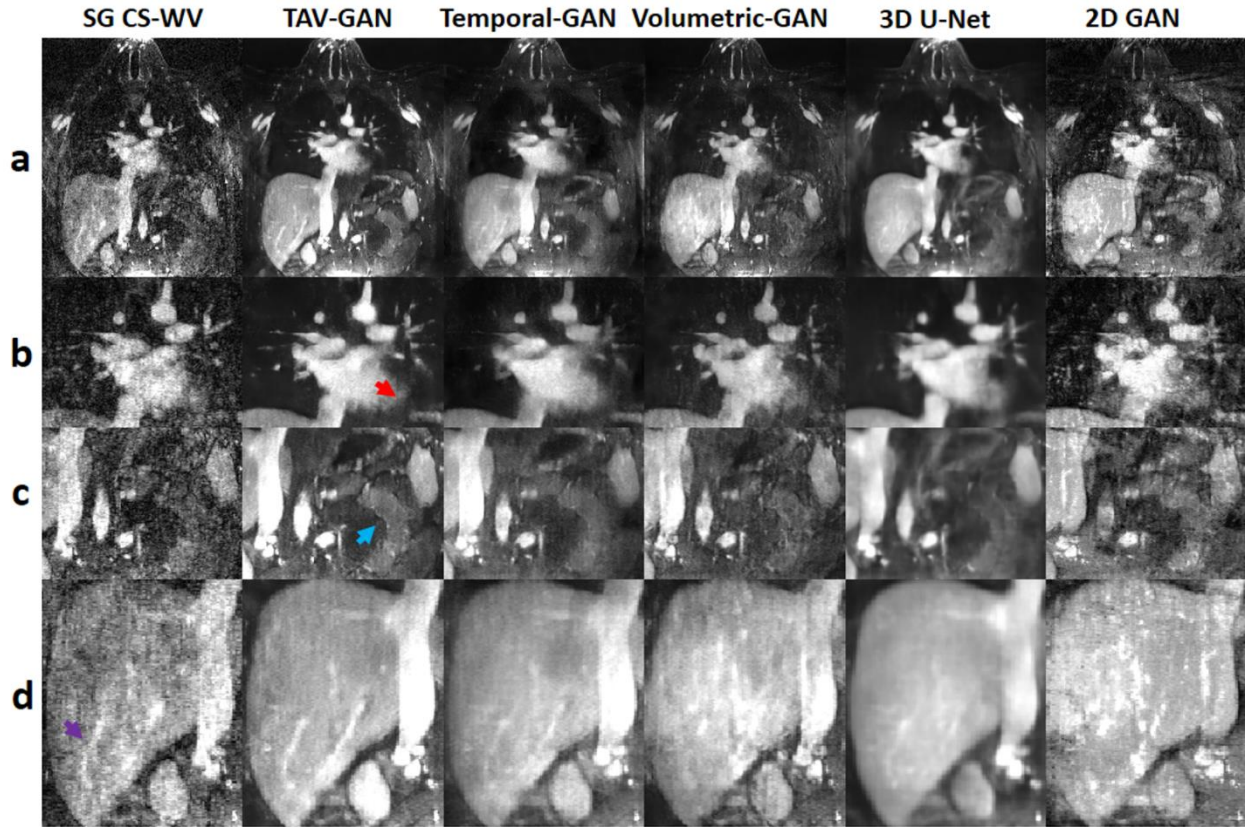

Figure 5-8. Qualitative comparison between different methods for a male CHD patient from test dataset B2 (21 y.o. and 77.4 kg weight). Although the CMR scan was performed under anesthesia, there was breathing irregularity during scanning. Row (a) shows the reconstructed image for a single slice, and rows (b-d) show the zoomed regions. The 2D GAN image not only suffers from residual artifacts but also shows the apparent anatomical change in particular in the liver. The TAV-GAN image appears sharper than the Temporal-GAN and the 3D U-Net. The myocardium border (row b, red arrow), soft tissue (row c, blue arrow), and the blood vessels in the liver region (row d, purple arrow) are all recovered better by TAV-GAN compared to other methods. The SG CS-WV was reconstructed based on 6.5X fold under-sampled data; the remaining methods shown were reconstructed based on 14.2X fold under-sampled data.

The reconstructed image from a 6.5X undersampled data (N_L=110000) by SG CS-WV had lower image quality than the reconstructed image by TAV-GAN from a 14.2X undersampled data. The small branch of the vessels in the liver (purple arrow, row d), soft tissue (blue arrow, row c), and the myocardium border (red-arrow, row b) were depicted well using TAV-GAN in comparison to the other methods. Subjective image quality scores for this case show that the TAV-GAN method with overall image quality score 4.5 and artifact score 4 has superior image quality than

the Temporal-GAN (overall image quality = 3.5, artifact score = 3.5) and SG CS-WV (overall image quality = 2.5, artifact score = 3). Complete 4D images for Figure 5-8 is provided in Video S2 (available online as a supporting file of our published article[10]).

Figure 5-9 shows representative reconstructions and respiratory motion correction results based on unseen data selected from the Group B2 with irregular breathing patterns acquired during spontaneous breathing without anesthesia. In this case, the TAV-GAN image quality was substantially better than the standard SG CS-WV (see arrowheads), despite the fact that the TAV-GAN reconstruction was based on 14.2X under-sampled data while the SG CS-WV was based on 6X (N_L=118000) under-sampled data. Compared with the TAV-GAN reconstruction, Temporal-GAN, Volumetric-GAN, 3D U-Net, and 2D GAN all suffered from artifacts ranging from additional blurring and additional aliasing artifacts. The 2D GAN reconstruction was essentially non-diagnostic. TAV-GAN method achieved 4 as the overall image quality score and 4 as the artifact score for this case which is superior to the SG CS-WV (overall image quality score = 2.5, artifact score = 2.5) and the Temporal-GAN (overall image quality = 3.5, artifact score = 3). Complete 4D images for Figure 5-9 is provided in Video S3 (available online as a supporting file of our published article[10]).

In stage 1 subjective image quality assessments, we identified that the TAV-GAN and the Temporal-GAN were better than the other network-based approaches. The multiple paired comparison results for stage 1 evaluation are reported in Table 5-2.

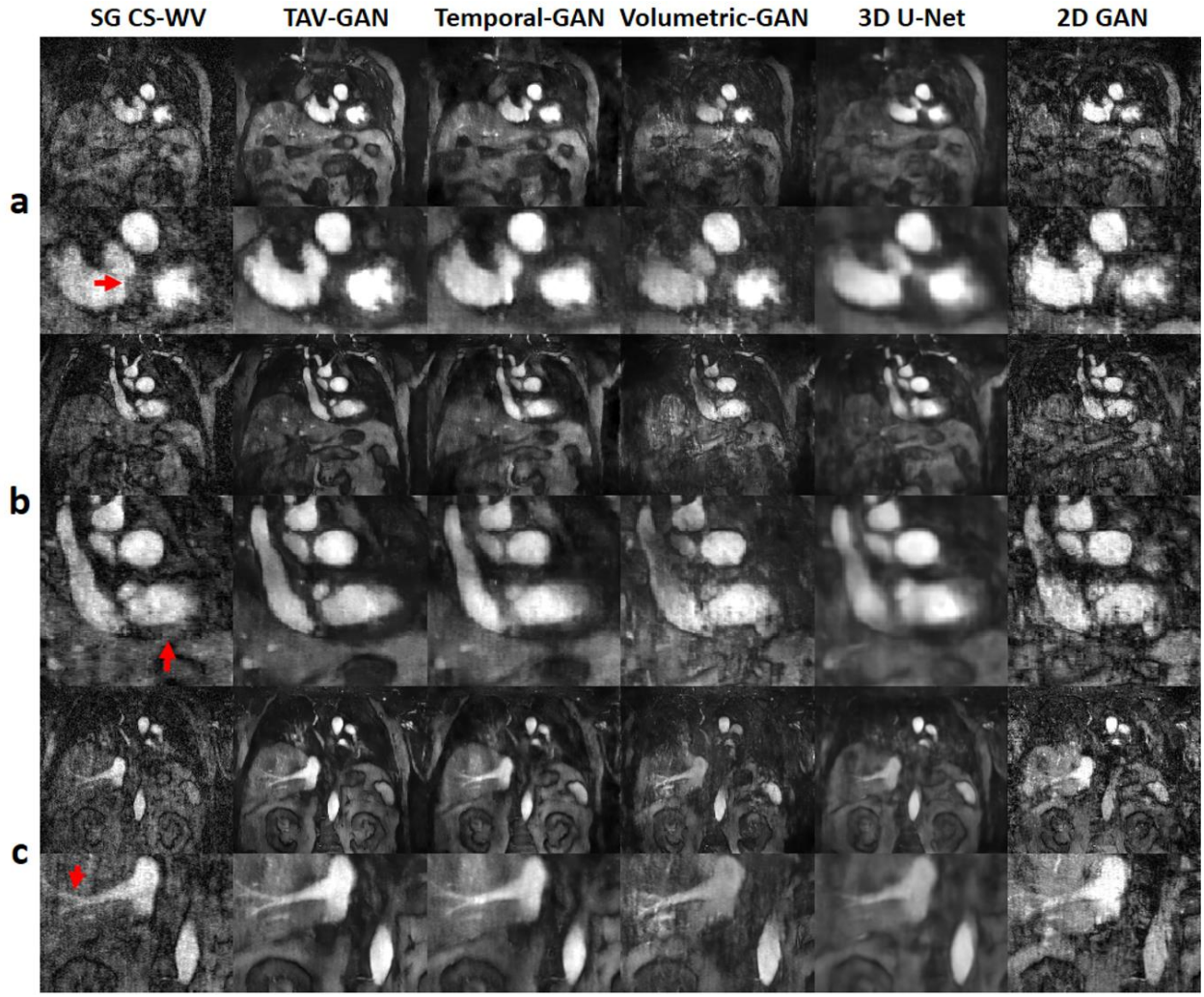

Figure 5-9. Qualitative result for a male patient from test dataset B2 (55 y.o. and 77kg weight), who underwent MRI during free-breathing without any anesthesia. The three rows (a-c) show some representative slices and cardiac phases that were reconstructed by using different methods. The TAV-GAN produced better delineation of various structures (red arrows) compared to all the other 5 methods. Compared to TAV-GAN, the 3D U-Net and Temporal-GAN images are blurrier, the Volumetric-GAN and SG CS-WV images have substantial artifacts, the 2D GAN image is of inferior quality. The SG CS-WV was reconstructed based on 6X fold under-sampled data; the remaining methods shown were reconstructed based on 14.2X fold under-sampled data.

Table 5.2. Multiple comparisons of subjective image quality rank comparisons were performed in Stage 1 subjective image quality evaluation. Among the 6 techniques ranked, only four techniques (Volumetric-GAN, Temporal-GAN, 3D U-Net, and self-gated CS-WV) are shown. We excluded TAV-GAN from this analysis because of its outstanding scores in the rank comparison, and it was consistently ranked highest among the 6 techniques. We also excluded the 2D GAN in this analysis because it was ranked consistently the worst among the 6 techniques. We excluded 2D GAN to ensure that the assumption of the variance's homogeneity is valid for the Tukey HSD test. At the α=0.05 level of significance, images

reconstructed by 3D U-Net had lower scores in comparison to the Temporal-GAN. Mean difference values indicated that the Temporal-GAN has a higher rank score than other methods, including Volumetric-GAN, SG CS-WV, and 3D U-Net, although the difference was not significant.

Comparison Method	(I) Method	(J) Method	Mean Difference (I-J)	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
Tukey HSD	Temporal-GAN	Volumetric-GAN	0.250	0.155	-0.06	0.56
		SG CS-WV	0.233	0.204	-0.07	0.54
		3D U-Net	0.417*	0.003	0.11	0.72
	Volumetric-GAN	Temporal-GAN	-0.250	0.155	-0.56	0.06
		SG CS-WV	-0.017	0.999	-0.32	0.29
		3D U-Net	0.167	0.496	-0.14	0.47
	SG CS-WV	Temporal-GAN	-0.233	0.204	-0.54	0.07
		Volumetric-GAN	0.167	0.999	-0.29	0.32
		3D U-Net	0.183	0.411	-0.12	0.49
	3D U-Net	Temporal-GAN	-0.417*	0.003	-0.72	-0.11
		Volumetric-GAN	-0.167	0.496	-0.47	0.14
		SG CS-WV	-0.183	0.411	-0.49	0.12

*. The mean difference is significant at the 0.05 level. Tukey HSD = Tukey honestly significant difference

Therefore, in stage 2 assessment, we included TAV-GAN and Temporal-GAN, as well the SG CS-WV. In stage 2 subjective evaluations, TAV-GAN, Temporal-GAN, and SG CS-WV achieved mean image quality (±SD) 4.53±0.540, 3.82±0.464, and 3.13±0.681, respectively. In terms of image artifact, TAV-GAN achieved a mean score (±SD) of 4.12±0.429, whereas Temporal-GAN and SG CS-WV received mean scores 3.47±0.370 and 2.97±0.434, respectively. Based on the multiple pair comparison tests, which are reported in Table 5-3, it can be concluded

that the images reconstructed by TAV-GAN had statistically significantly higher quality and lower artifact levels than the Temporal-GAN and SG CS-WV methods (P<0.05 for both comparisons).

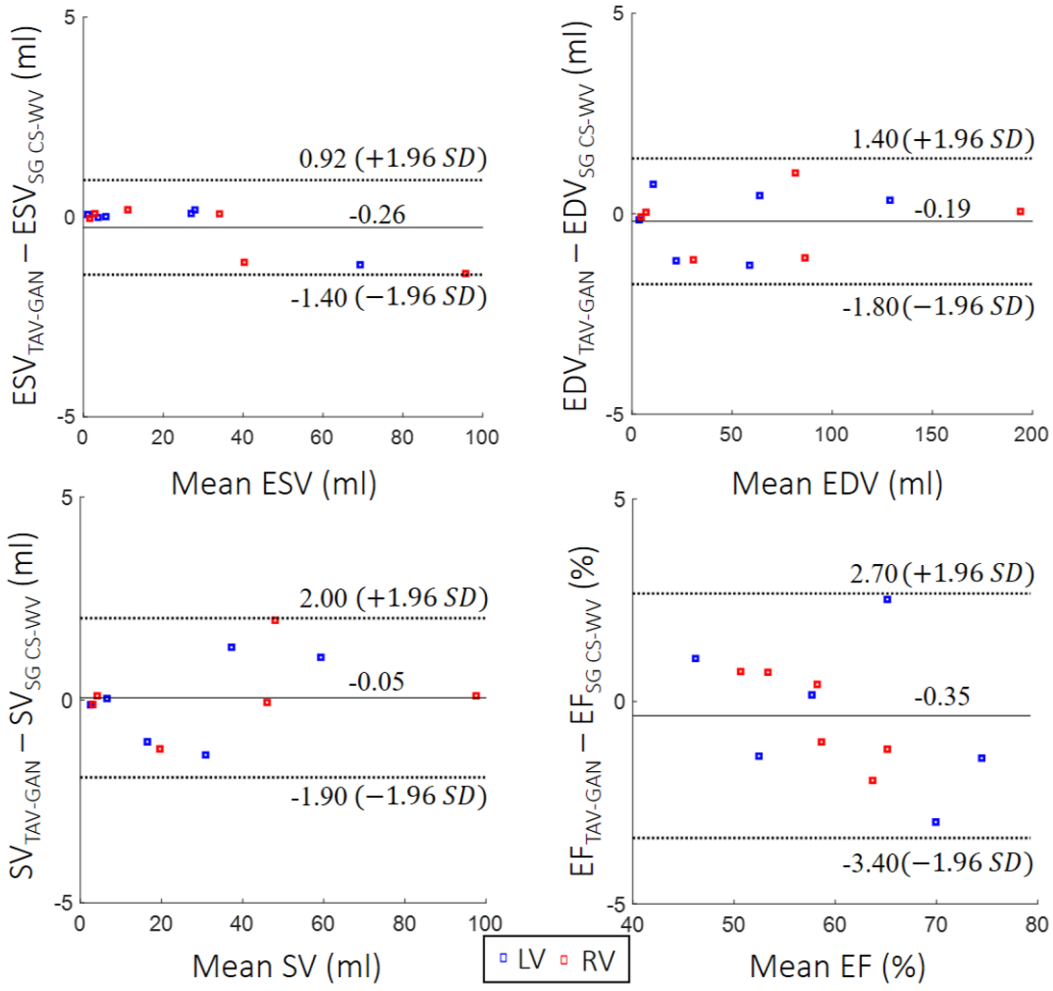

Figure 5-10. Functional analysis: Left and right ventricular endocardial borders were segmented by an experienced expert to compute stroke volume (SV), end-systolic volume (ESV), end-diastolic volume (EDV), and ejection fraction (EF) for 6 test cases. Bland-Altman plots confirm that there is agreement with 95% confidential level between functional metrics measured from the reconstructed images by self-gating CS-WV images and respiratory motion-corrected and reconstructed images by TAV-GAN.

Table 5-3. Multiple comparisons between the overall image quality score and the artifact score of the images which were reconstructed by temporally aware volumetric GAN (TAV-GAN), Temporal-GAN, and self-gated CS-WV (SG CS-WV). At the α=0.05 level of significance, the overall image quality and artifact score of the images were reconstructed by the TAV-GAN is higher than the images reconstructed by

Temporal-GAN or SG CS-WV. Besides, Temporal-GAN reconstructs the images with a statistically significant higher image quality and lower artifact than the conventional SG CS-WV.

Overall Image Quality Score					
(I) Group1	(J) Group1	Mean Difference (I-J)	Sig.	95% Confidence Interval	
				Lower Bound	Upper Bound
TAV-GAN	Temporal-GAN	0.717*	0.000	0.37	1.07
	SG CS-WV	1.400*	0.000	1.05	1.75
Temporal-GAN	TAV-GAN	-0.717*	0.000	-1.07	-0.37
	SG CS-WV	0.683*	0.000	0.33	1.03
SG CS-WV	TAV-GAN	-1.400*	0.000	-1.75	-1.05
	Temporal-GAN	-0.683*	0.000	-1.03	-0.33
Image Artifact Score					
(I) Group1	(J) Group1	Mean Difference (I-J)	Sig.	95% Confidence Interval	
				Lower Bound	Upper Bound
TAV-GAN	Temporal-GAN	0.650*	0.000	0.40	0.90
	SG CS-WV	1.150*	0.000	0.90	1.40
Temporal-GAN	TAV-GAN	-0.650*	0.000	-0.90	-0.40
	SG CS-WV	0.500*	0.000	0.25	0.75
SG CS-WV	TAV-GAN	-1.150*	0.000	-1.40	-0.90
	Temporal-GAN	-0.500*	0.000	-0.75	-0.25

*. The mean difference is significant at the 0.05 level. Tukey HSD = Tukey honestly significant difference

Figure 5-10 shows Bland-Altman plots of the left/right ventricular SV, ESV, EDV, and EF for the cardiac functional analysis. Bland–Altman analysis demonstrates that the cardiac function parameters calculated based on reconstructed images by TAV-GAN were in good agreement with SG CS-WV images by considering both upper and lower 95% agreement limits.

5.5 Discussion

We demonstrated TAV-GAN as a promising technique for reconstructing highly under-sampled and respiratory motion-corrupted 4D data sets. Several previous deep learning-based image reconstruction techniques, in particular, GAN based approach, are focused on under-sampled data recovery[57,58,142,151], or motion compensation[9,84,88,90,143,144]. To our knowledge, this is the first 3D GAN network for simultaneous under-sampled k-space data recovery and respiratory motion compensation. Our work includes several innovations with regard to the loss function and the training process. In particular, our TAV-GAN technique incorporates a temporally aware objective function as an extra regularizer in addition to adversarial loss, L1 and SSIM loss functions to reduce flickering artifacts through the cardiac phases with no explicit need to use the multiple cardiac phases as the inputs for the network. Besides, we addressed the well-known challenges associated with training GANs for high-dimensional images by adopting an effective progressive training strategy based on starting the training from the low-resolution volumetric images and gradually increasing the resolution to reach to the original volumetric image size. Based on the convergence of the adversarial loss components of the generator and the discriminator

and qualitative validation results through the epochs (Fig. 5.11), it can be concluded that the proposed training strategy was effective and successful.

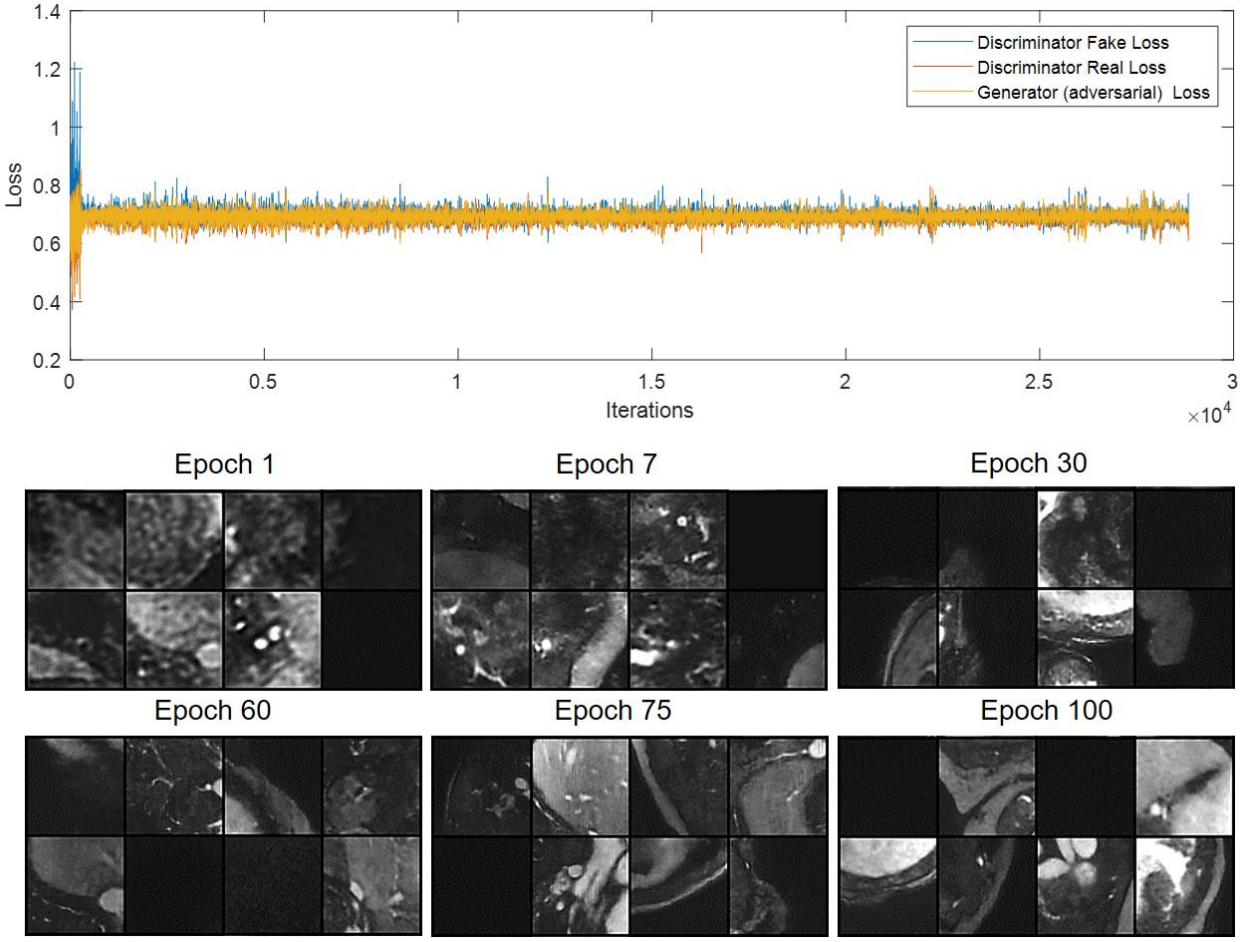

Figure 5-11. Training convergence: first row plots the loss components versus the iterations for the generator and the discriminator of the temporally aware volumetric GAN (TAV-GAN). Only adversarial loss is plotted for the generator, and it means how well the generator can fool the discriminator. The discriminator contains two components associated with classification performance for both real and fake images. As seen in the first row, all three components converge to an equilibrium state (0.7). Besides, this convergence is happening very fast because of the practical training strategy introduced in this work. The second row shows the qualitative validation results through the epochs. It seems that after epoch 60 (15000 iterations), image quality is improved sufficiently.

Our data show that 3D networks outperformed 2D networks for all of our test data sets. In our test datasets of 30 patients, we found that the TAV-GAN network outperformed all of the other 5 techniques compared. Interestingly, our 10.7X-15.8X accelerated TAV-GAN images outperformed the 3.5X-7.9X accelerated SG CS-WV images. This was because we intentionally chose to include images with higher visual image quality in the training dataset. This compelled the network to learn the underlying data distribution of a high-quality dataset and enabled it to reconstruct higher quality images than SG CS-WV for data with noisier and undesirable residual motion as shown in Figures 5-8 and 5-9. Such outperformance of the TAV-GAN over the SG CS-WV could break if sufficient data lines were acquired for SG CS-WV under the regular and high gating-efficiency, which is not guaranteed to exist or if it exists, it can further elongate the scan time. For example, Video S1 (available online as a supporting file of our published article[10]) shows a patient case for which the SG CS-WV with overall image quality 5 outperformed the TAV-GAN with an overall image quality of 4.5. In this case, respiratory motion was low and periodic, and scan time for SG CS-WV (7.2 min, $N_L \approx$ 148000 lines) was almost three times the required scan time for TAV-GAN (2.4min, 50000 lines).

The Temporal-GAN has a better performance than the Volumetric-GAN. It was expected because the Temporal-GAN uses the spatiotemporal redundant information to reconstruct the image from the respiratory motion-corrupted and undersampled zero-filled images. However, the mean of the sharpness score (normalized Tenengrad focus measure) was decreased 15% from the Volumetric-GAN (0.828) to the Temporal-GAN (0.702). Although not statistically significant, it is still visually evident in the qualitative results presented in Figures 5-7, 5-8, and 5-9. It seems that the adjacent cardiac frames in the Temporal-GAN contribute to the blurriness of the results. The mean sharpness of the results obtained by TAV-GAN (0.822) trended marginally lower than

the Volumetric-GAN (0.828), although the comparison was not statistically significant. Since the technical difference between the TAV-GAN and the Volumetric-GAN is the TA loss, it may be concluded that including the TA loss as an additional constraint on the Volumetric-GAN may decrease the residual artifacts and increase the quality of the results as well as preserving the sharpness of the results. For instance, as shown in Figure 5-8, it appears the TAV-GAN image is as sharp as the Volumetric-GAN image, but with reduced residual artifacts.

We note that the Temporal-GAN network is a 3.5D spatiotemporal network. It uses the redundant information in the three sequential aliased and respiratory motion-corrupted 3D cardiac frames t-1, t, and t+1 to reconstruct the cardiac frame t. A 3D spatiotemporal GAN, which can be applied to the ROCK MUSIC data after a Fourier Transform in the readout direction, could also be considered the potential approach for removing the artifacts from the aliased and respiratory motion artifact corrupted images. Based on the results of the comparison study detailed in Appendix V, it can be concluded that the performance of the Temporal- GAN is superior to the 3D spatiotemporal GAN in removing the aliasing and respiratory artifacts from the image. Such superior qualitative performance might be explained by considering that the Temporal-GAN exploits 3D spatial information while the 3D spatiotemporal GAN is only using 2D spatial information.

The TA loss introduced in this work is a data-driven-based loss that requires a pretrained discriminator. It is analogous to the perceptual loss[89], in which the well-known pretrained classifier VGG-16 network is used to compute the perceptual loss for 2D image space. We used the discriminator part of the pretrained Temporal-GAN to compute the TA loss for the 3D images in our work. Indeed, the temporal discriminator can be seen as a 3D classifier trained in an adversarial setting. Based on the empirical results that were shown in Figures 5-6, 5-7, 5-8, and 5-9, the TA

loss had two main advantages. 1) It decreases the flickering artifacts through the cardiac frames without explicitly using the cardiac frames as the input. 2) It acts as an extra constraint on the generator, which results in improved quality of the generated images. Since the TA loss is a squared L2 norm of the two 3D images in the feature space, in which the features were calculated based on the output of the convolutional layers of a pretrained temporal discriminator, it can be used as an extra loss function to regularize other non-adversarial based 3D networks as well. The TA loss's effectiveness could be further increased by considering the joint training scheme for the Volumetric-GAN and the Temporal-GAN.

All data in this study were acquired using the 3D spoiled gradient-echo sequence, i.e., ROCK-MUSIC technique in the steady-state distribution of the ferumoxytol contrast agent. In Ferumoxytol enhanced acquisitions, the images would be very sparse even in the image domain with no transformation, such as wavelet transformation or total variation. Since the proposed method structurally includes several compression stages, i.e., downsampling stages, it achieved relatively high acceleration factors for the Ferumoxytol enhanced datasets which are inherently more compressible than the data acquired without a contrast agent. Therefore, we speculate that the proposed method would achieve the lower acceleration factors in images acquired with no contrast agent. To generalize our technique to data acquired without ferumoxytol, domain adaptation and transfer learning-based technique could enable us to adjust the network to non-contrast-enhanced 4D CMR images. Evaluation of the network performance on non-contrast-based 4D CMR images is warranted in future studies.

We used only cardiac gated zero-filled reconstructed images as the input for training the network. An alternative strategy is to train the network based on cardiorespiratory-gated zero-filled reconstructed images as the input, in which case the network would only need to learn how to

remove under-sampling aliasing artifacts, a task that could be easier than learning to remove both respiratory motion artifacts and under-sampling aliasing artifacts simultaneously. Based on the supplemental study reported in the Appendix VI, TAV-GAN trained using cardiac-gated zero-filled images as the input demonstrated better robustness in the testing stage on the data with irregular breathing than the TAV-GAN based on cardiorespiratory-gated zero-filled images as the input. This is indeed rational because the self-gating signal could not represent the respiratory motion well in the presence of irregular breathing, and residual respiratory motion artifact might have remained in the input data after respiratory self-gating, which presents a challenge to the TAV-GAN that only learned to remove the under-sampling aliasing artifacts.

Respiratory motion is still a major bottleneck in cardiothoracic and abdominal imaging. The entire breathing cycle is highly non-linear with non-rigid and often somewhat non-periodic breathing patterns. In image reconstruction, it is common to model the forward operation of motion in the MRI signal or incorporate these forward motion models in neural networks. As the respiratory motion has a non-rigid nature and no well-defined relationship between non-rigid respiratory motion and k-space, the inverse problems' forward operation is often not fully understood mathematically and represents challenges for incorporation into neural networks. In our TAV-GAN, the network learns the underlying data distribution for a sharp, respiratory motion-compensated image by starting from the initially zero-filled and respiratory motion-corrupted reconstruction. Deep neural networks' ability to learn the non-linear motion during MRI signal encoding holds great promise for addressing the current challenges in cardiothoracic and abdominal imaging.

Two significant concerns exist in the use of the GANs in image reconstruction and respiratory motion compensation tasks in medical imaging. First, can these networks preserve the individual

patient's anatomy and pathology in reconstructing the highly aliased, respiratory motion-corrupted images. Second, can these networks introduce new spurious anatomical features in the images. To address the first concern, we included content loss and TA loss to constraint the generator's output in TAV-GAN and imposed the consistency in the image domain. Indeed the data consistency term which is usually imposed on the raw k-space data was not employed in this work mainly because of the unknown nature of the forward operation for the respiratory corrupted measurements. To address the second concern, we trained the network based on the images with minimal noise by carefully curating the training data. As shown in Figure 5-12, by training the network based on the images with higher noise, e.g., Group B1, new spurious features were introduced to the reconstructed images. In fact, this is expected mainly because of the generative nature of the GANs that can enable them to learn how to turn the noise from the input data into spurious features, which could potentially lead to misdiagnosis. Our subjective image quality evaluation confirms that there were no new generated spurious features in our TAV-GAN images.

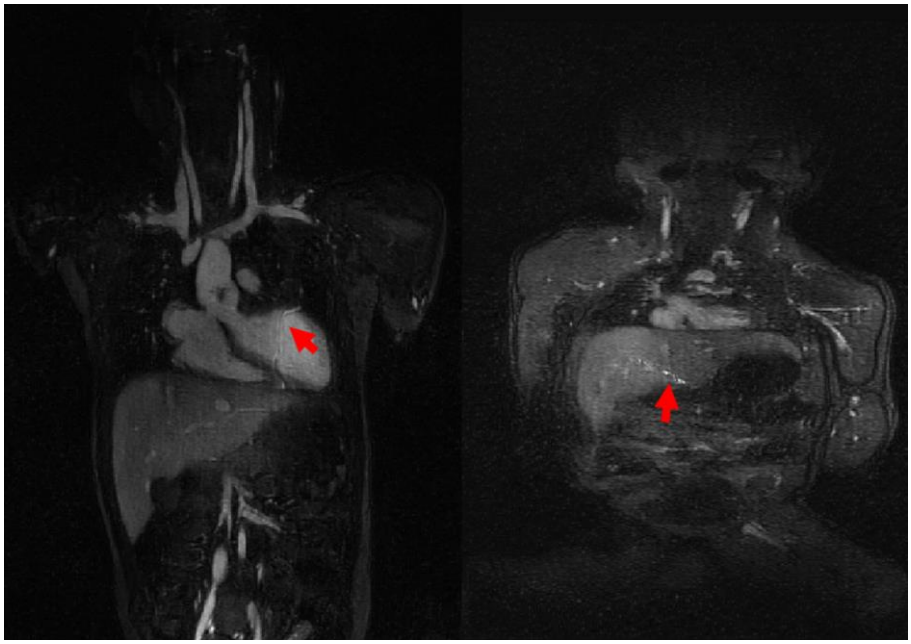

Figure 5-12. Hallucination effect: by training the generative adversarial networks on the datasets with noisy ground truth, some characteristic artifacts were introduced to the image. As pointed with the red arrow, such a network generated spurious artifact has appeared in the left myocardium and liver region. For this case, we trained the network on dataset B1 and tested it on dataset A. We note that on average, the dataset B1 was two times noisier than the dataset A. This result reveals the importance of curating the data and using less noisy target reference images for training GANs. Otherwise, spurious features might be introduced to the reconstructed images.

5.6 Conclusion

This study implemented a novel 3D generative adversarial network (GAN)-based technique for simultaneous image reconstruction and respiratory motion compensation. We showed that the proposed platform could achieve high acceleration factors while maintaining robust and diagnostic image quality superior to state-of-the-art self-gating (SG) compressed sensing wavelet (CS-WV) reconstruction at lower acceleration factors 3.5X-7.9X.

Chapter 6 Fast and Accurate Calculation of Myocardial T1 and T2 Values Using Deep Learning Bloch Equation Simulations (DeepBLESS)

This work aims to develop a convolutional neural network for fast and accurate estimation of myocardium T1 and T2 relaxation values based on a previously proposed Bloch equation simulation with slice profile correction (BLESSPC) method. We proposed the deep learning Bloch equations simulations (DeepBLESS) models to calculate the T1 values of the MOLLI T1 mapping sequence with bSSFP readouts and T1/T2 values of the radial simultaneous T1 and T2 mapping sequence. We evaluated the accuracy of DeepBLESS T1/T2 estimation on the simulated data with different noise levels. We also compared the performance of the DeepBLESS models against BLESSPC in simulation, phantom, and in vivo studies for the MOLLI sequence at 1.5T and radial T1-T2 sequence at 3.0T. The phantom and in vivo studies showed that the trained DeepBLESS model and conventional BLESSPC method achieved statistically similar accuracy and precision in T1/T2 estimations for both MOLLI and radial T1/T2. A version of this chapter has been published[11] in the Magnetic Resonance in Medicine:

1. Shao, J, Ghodrati, V, Nguyen, K-L, Hu, P. Fast and accurate calculation of myocardial T1 and T2 values using deep learning Bloch equation simulations (DeepBLESS). Magn Reson Med. 2020; 84: 2831– 2845. https://doi.org/10.1002/mrm.28321

6.1 Introduction

Quantitative myocardial tissue relaxometry techniques, e.g. T1 and T2 mapping, are emerging and rapidly evolving cardiovascular magnetic resonance techniques for non-invasive, quantitative characterization of cardiac tissue[152-160]. To generate a T1 or T2 map, multiple images with different T1 or T2 weighting are acquired, and the tissue T1/T2 parameters are estimated pixel by pixel by fitting the acquired signal to a model-predicted signal. A commonly used model for T1 or T2 calculation is exponential curve fitting[155,156,160]. The exponential curve fitting model is accurate under certain conditions and is computationally efficient. However, this basic model cannot model the signal evaluation accurately for some cardiac MRI sequences, such as the widely used the Modified Look-Locker inversion recovery (MOLLI) pulse sequence[155], resulting in inaccurate parameter estimation[32]. To address the issue, Bloch-equation simulation-based algorithms have been proposed to model the signal evolution of a sequence to ensure accurate parameters estimation, such as the Bloch equation simulation with slice profile correction (BLESSPC) algorithm for the MOLLI[32] and simultaneous radial T1-T2 mapping[162] sequences and the SQAUREMR algorithm for MOLLI[163]. Bloch equation simulation is also the key for accurate T1 and T2 map calculation in the cardiac MR fingerprinting (MRF) technique[164-166].

However, Bloch-equation-simulation-based approaches are usually time consuming, especially for more comprehensive simulations[32,163,165]. The computation time of the Bloch equation simulation is important to consider in cardiac application because the simulation needs to incorporate the scan-specific heart rate variations after each scan. This is different from the application that use fixed sequence timing, such as brain MRF[167], where the time-consuming computations can be performed in advance to create the dictionary and then be used for subsequent scans[165]. Recently, machine learning has been applied in MRF to accelerate Bloch equation-

simulation-based parameter estimation[168,169], including the deep reconstruction network (DRONE)[168]. DRONE uses a 4-layer neural network containing two 300 x 300 hidden layers. The network was trained with a dictionary generated using Bloch equation simulations, using the simulated signal as input and T1/T2 values as output[168]. As the timing of the MRF sequence is fixed, the DRONE approach does not use the information of signal acquisition time and therefore cannot be directly be used for cardiac parameter mapping without being adapted to scan-specific heart rate variations. To solve this issue, Hamilton et al.[170] demonstrated that deep learning can be used to accelerate dictionary generation for cardiac MRF, followed by gridding and pattern matching to calculate T1 and T2 values. However, in this work, the effect of B1+ variations was not considered and gridding and matching were still needed for T1/T2 calculation, which could potentially reduce the T1/T2 estimation accuracy.

While the cardiac MRF T1/T2 estimation approach is mainly optimized and validated for the cardiac MRF sequence[164-166], the BLESSPC approach has been shown to generate accurate T1/T2 maps for both conventional widely-used Cartesian-based sequences[32,171,172] and radial sequences[162]. Furthermore, BLESSPC is an optimization-based approach, while cardiac MRF T1/T2 estimation needs T1/T2 gridding and the T1/T2 estimation accuracy and precision may be limited by the grid size. However, BLESSPC sometimes suffers from relatively long computation time, such as when used for the MOLLI sequence[32] to improve T1 estimation accuracy and for the radial T1-T2 mapping sequence when both the inversion pulse and T2 preparation pulses were simulated in detail to ensure good accuracy[162].

Therefore, we propose a new approach, DeepBLESS, which applies deep learning to BLESSPC to enable rapid myocardial T1/T2 parameters calculation. DeepBLESS can be adaptive to heart rate variations, achieving the same accuracy and precision with BLESSPC, while reducing

the reconstruction time to be less than one second. Different from the deep learning approach proposed for cardiac MRF T1 and T2 estimation, DeepBLESS considers the effect of B1+, and predicts T1/T2 values directly without the need of gridding and pattern matching. In this work, we demonstrate the benefits of the DeepBLESS using two sequences: the Modified Look-Locker inversion recovery (MOLLI) T1 mapping sequence at 1.5T and a recently proposed simultaneous radial T1 and T2 mapping sequence[162] at 3.0T.

6.2 Methods

6.2.1 Pulse sequence

The radial T1-T2 sequence is an ECG-triggered sequence that uses combined inversion recovery and T2-prearation with golden angle radial spoiled gradient echo readout, acquiring data in a single breath-hold of 11 heartbeats[162], as shown in Figure 6-1. Based on the acquired multi-coil data, 110 images were reconstructed using compressed sensing with spatial and temporal total variation (TV) regularization, 10 images for each heartbeat on a sliding temporal window. The signal polarity for the measured signal was assigned by a phase-sensitive method[173]. Subsequently, both T1 and T2 maps were reconstructed using the extended BLESSPC algorithm[162]. In detail, BLESSPC for radial T1-T2 mapping simulates the signal evolution of the radial T1-T2 sequence using Bloch equation simulations, considering the effect of non rectangular slice profile and non-perfect adiabatic inversion, and the T2 preparation were simulated in detail at a step size of 5us.

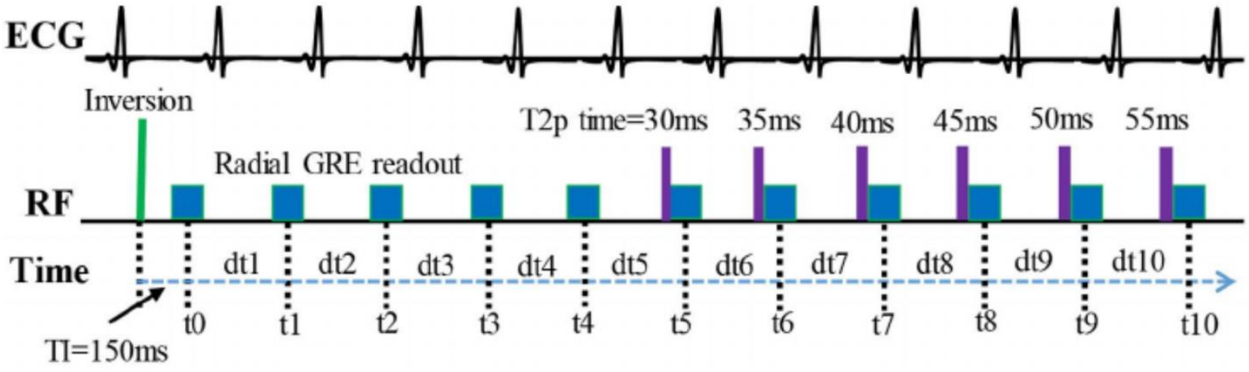

Figure 6-1. The radial T1-T2 sequence image acquisition, where t0, t1, …, t10 indicate the image acquisition time points, defined as the time when the 40th k-space line is acquired, and dt1,dt2, …, dt10 are the durations between each acquisition time point, which are needed in the Bloch equation simulation for T1 and T2 calculation.

In this work, DeepBLESS were compared to BLESSPC for T1 and T2 map reconstruction regarding accuracy, precision and calculation speed, based on the 110 reconstructed images generated by the radial T1-T2 sequence. To demonstrate that the proposed network (described in the next section) can be adaptive to the other cardiac parameters mapping applications, DeepBLESS was also applied to the widely used MOLLI 5-(3)-3 sequence (11) for T1 map reconstruction, with the same network structure but with separate network training and different input layer size. In both phantom and in vivo studies, the flip angle (FA) used was 6° for the radial T1-T2 sequence, and FA = 35° for MOLLI.

6.2.2 Network for DeepBLESS

In DeepBLESS, a deep convolutional neural network was used, which is composed of a cascade of convolutional layers with ResNet blocks[28] and a dense layer as the last layer connected to the output layer, as shown in Figure 6-2.

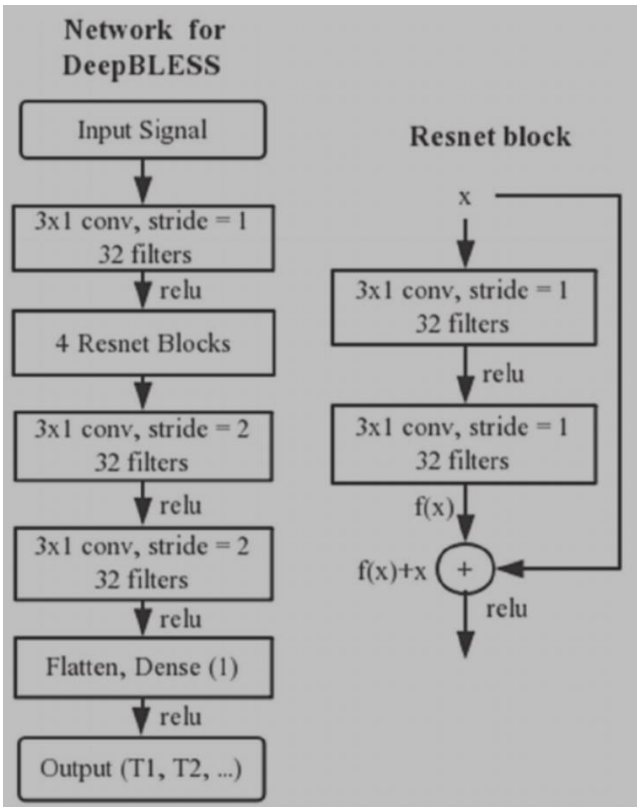

Figure 6-2. Illustration of the proposed network for DeepBLESS. The network composed of 13 layers, including the input layer, one 3x1 convolutional layer followed by 4 ResNet blocks and two 3x1 convolution layers. Then, a dense layer was added to predict T1/T2 value. The number of filters for each convolutional layer was set to be 32 and the stride was set to be 1 except the last two convolutional layers, which use a stride of 2.

The input layer consisted of a 1D time varying signal with several channels varied depending on the sequence; the first channel corresponds to the acquisition time stamps at each heartbeat and the other channels store the actual signals acquired. In this study, DeepBLESS was applied to the simultaneous radial T1 and T2 mapping sequence (referred to as the radial T1-T2 sequence hereafter)[162] to predict T1/T2 values for each pixel, and the MOLLI 5-(3)-3 sequence[32,174] to predict T1 value for each pixel. In our implementation, we used eleven convolution layers, including four ResNet blocks ($R_n = 4$). Each convolutional layer had 32 filters with 3×1 size and a strike of one, except the last two convolutional layers, which used a stride of two. These parameters were empirically selected to ensure accurate functional mapping while avoiding the

risk of overfitting (see Appendix VII and Table A-1 for more information about the optimization). A rectified linear unit (ReLU) activation function[26] was used for the hidden layers and for the output layer. The total number of trainable parameters of DeepBLESS was ~31,000 for MOLLI and ~32,000 for radial T1-T2. The data sizes of different layers of the network from input to the output are shown in Table A-3 of Appendix VII.

6.2.3 DeepBLESS Training

Before applying the proposed model for T1/T2 calculation, DeepBLESS was trained using simulated data for each sequence independently. Bloch equation simulation[32,162] was used to generate the training sets (1,000,000 samples), validation set (100,000 samples) and testing set (100,000 samples) for each sequence. For each simulation, random T1 and heart rate (HR) were randomly sampled from the range 200 – 2000 ms and 40 bpm - 100 bpm, respectively. For T2, 90% of the simulation data had a randomly selected T2 between 20 ms – 100 ms and 10% of the simulation data had a randomly selected T2 between 100 ms – 200 ms. To consider the possible B1 variations, the flip angle α was randomly sampled between 3° - 8° for the radial T1-T2 sequence and between 20°- 45° for the MOLLI sequence. For each group of randomly sampled T1, T2, α, and HR, a HR variation was simulated across the multiple heartbeats of data acquisition according to a Gaussian distribution. In detail, for either the radial T1-T2 or the MOLLI sequence, 10 cardiac cycle lengths (i.e. t1-t10 in Figure 6-1) were simulated for each simulation data set. For a given randomly selected HR, Random HR (bpm), the 10 cardiac cycle lengths were generated using Equation (6-1).

$$\text{Duration(i)} = \frac{60000}{\text{RandomHR}} \times (1 + 0.1 \times \text{Randn1}_i) \qquad (6\text{-}1)$$

where i = 1,2, …, 10 and Randn1i is the i^{th} random value drawn from a Gaussian distribution with mean = 0 and standard deviation = 1. All the randomly selected values, such as T1, T2, HR, etc.

were random floating point values. To ensure the robustness of our network for missed heartbeats, a common occurrence in clinical cardiac scanning, each simulated heartbeat in our training data had 1% chance of being skipped. Therefore, approximately 9.5% (based on the simulation results) of our training data had at least 1 skipped heartbeat. Skipped heartbeats were also simulated for the validation and testing data in a similar fashion.

There were differences in how the sequence was simulated between the radial T1-T2 sequence and the MOLLI sequence. For the radial T1-T2 sequence, both the inversion and the T2 preparation pulses were simulated in detail; while for the MOLLI sequence, the inversion was assumed to be instantaneous and a fixed inversion factor of 0.96 was assumed, which is the estimated average inversion factor on tissues with T1, T2 similar to myocardium for the inversion pulse used [32,175].

A previous study shows that adding noise to the training data promotes robust learning[168]. Therefore, real-valued Gaussian noise was added to the simulated signal before model training. For the MOLLI T1 mapping, the signal-to-noise ratio (SNR) was restively high in each balanced steady state free precession (bSSFP) image, and the final model was trained by adding 1% SD (standard deviation) Gaussian noise to the training data. For radial T1-T2 mapping, the SNR was lower in each reconstructed image, and a wider range of noise levels were simulated. Specifically, four DeepBLESS models were trained after adding Gaussian noise to the training data at SD levels of 1%, 5%, 9% and a composite range of 1%-10%, respectively. To train the networks, we used mean square error (MSE) as the loss function with a batch size of 2000. For the radial T1-T2 sequence, the input signal was a 1D signal with 11 nodes (representing 11 heartbeats) and 11 channels (1 channel for recording the acquisition time stamp signal, and the remaining channels for the 10 acquired signal in each heartbeat). Essentially, the input signal includes 110 signal intensity values on the T1 and T2 relaxation curves for a given pixel, along with the necessary

time stamps. The output was the T1/T2 value for the corresponding pixel. For the MOLLI sequence, the input signal was a 1D signal with 8 nodes (representing 8 heartbeats with data acquisitions) and 2 channels (1 acquisition time stamp signal + 1 acquired signal in each acquisition) for each pixel, and the output was the T1 value for the corresponding pixel.

Assume the input values are X, the function of the network to map from input to output is ($\boldsymbol{\theta 1}$,X) for T1, and $\boldsymbol{f}(\boldsymbol{\theta 2},\boldsymbol{X})$ for T2, where $\boldsymbol{\theta 1}$ and $\boldsymbol{\theta 2}$ are the trainable parameters. Then the loss functions for T1 and T2 are represented as Equations (6-2) and (6-3).

$$\sum_{i=0}^{M}[f(\theta_1 - X_i) - T1(i)]^2/M \quad (6\text{-}2)$$

$$\sum_{i=0}^{M}[f(\theta_2 - X_i) - T2(i)]^2/M \quad (6\text{-}3)$$

Where i indicate the i^{th} sample, M is the batch size.

The Adam optimizer was set with a learning rate of 0.0005 for 500 epochs. The best model parameters were loaded and retrained with a learning rate of 0.0001 for 100 epochs. The model training took 1.0 ~ 1.2 hours using a general-purpose computer with a NVIDIA GTX 1080 GPU. Model parameters with the best MSE from the validation set were saved and used for simulation, phantom and in vivo studies. The learning rate strategy used in this work is a special case of step decay learning rate annealing approach. The performance of two other learning rate annealing methods was compared in Figure A-7 of the Appendix VII.

6.2.4 Simulation Study

After model training, the performance of DeepBLESS was evaluated using testing data sets randomly generated using Bloch equation simulations described in the Section 6.2.3. For the radial T1-T2 sequence, the 4 trained models were used to predict the T1 and T2 values in the test data. Random Gaussian noise with SD from 1% (SNR = 100) to 9% (SNR = 11.1) (1% increments) was added to the testing data before the evaluation. The conventional BLESSPC T1 and T2 estimation

algorithm was applied to the testing data to calculate BLESSPC T1 and T2 values for comparison. The predicted T1 and T2 values were compared to the corresponding reference values using the formula: Error = (Predicted – Reference) and Error% = Error /Reference ×100%. The mean of the absolute percent error was calculated.

6.2.5 MRI

For both phantom and in vivo studies, the radial T1-T2 sequence was performed on a 3.0T MRI scanner (Prisma, Siemens Healthineers, Erlangen, Germany). The MOLLI sequence was performed on a 1.5T MRI scanner (Avanto Fit, Siemens Healthineers, Erlangen, Germany). The manufacturer's body phased array and the spine coils were used for both scanners. The radial T1-T2 sequence was acquired with field-of-view (FOV) = 320 × 320 mm^2, TR/TE = 2.5 ms/1.4 ms, slice thickness = 8 mm, pixel size = 1.7 × 1.7 mm^2, with a reconstructed matrix size of 192 × 192, where 80 radial spokes were acquired in each heartbeat. The images with both magnitude and phase signal were reconstructed off-line after the data was acquired. The MOLLI 5-(3)-3 sequence was acquired with FOV = 340 × 273 mm^2, TR/TE = 2.5 ms/1.1 ms, slice thickness = 8mm, interpolated pixel size = 1.8 × 1.8 mm^2. The magnitude and phase images were reconstructed on-line with 2X GRAPPA with 24 k-space auto-calibration lines. Based on the acquired magnitude and phase images, the real-valued signal for each pixel was calculated based on a phase-sensitive method[173,176], using the phase image with the longest inversion time as the reference phase. Then the real-valued signal was used for T1/T2 estimations using BLESSPC and DeepBLESS.

6.2.6 Phantom Studies

For the radial T1-T2 sequence, eight 50 ml agar and CuSO4 gel phantoms were used. The radial T1-T2 sequence was performed at simulated HR from 40 bpm to 100 bpm (10 bpm

increments) and were repeated ten times at simulated HR of 60 bpm to evaluate T1 and T2 precision. For MOLLI T1 mapping, ten 50-ml agar and CuSO4 gel phantoms were used. The MOLLI sequence were acquired at each simulated HR from 40 to 100 bpm (20 bpm increments) and were repeated ten times at simulated HR of 60 bpm to evaluate precision. While the cardiac cycle lengths were randomly simulated during our model training, the simulated cardiac cycle lengths in the phantom study were different from training data. Reference T1 and T2 values for each gel phantom were determined by a standard inversion recovery spin echo technique with 12 TIs (TI = 50–5000 ms), TR/TE = 10 s/4.6 ms. Reference T2 values were calculated using a standard spin-echo technique with 11 TEs (TE = 5–250 ms), with TR = 10 s. A region of interest (ROI) was manually drawn for each tube and the average T1, T2 values were used as reference T1 and T2 values. The accuracy was evaluated by calculating the difference and percentile difference between the estimated T1/T2 values with reference T1/T2 values. The precision was measured using coefficient of variation (CoV), CoV = SD / Mean × 100%, where SD is the standard deviation of the measured T1 or T2 values over repeated scans for each ROI.

6.2.7 In Vivo Studies

The in vivo study was approved by the Institutional Review Board and was compliant with the Health Insurance Portability and Accountability Act. All subjects provided written informed consent. Standard cardiac shimming was applied to reduce off-resonance variations in the heart region. The radial T1-T2 sequence was performed in ten healthy volunteers (8 males, aged 35.9±14.0 years, range 24 - 65 years) at 3.0T. The MOLLI 5-(3)-3 sequence was acquired in eight healthy volunteers (5 males, aged 28.9±4.3 years) at 1.5T. Images of the mid-left ventricular (mid-LV) short-axis were acquired at end-expiration for each scan. After the radial T1-T2 data were acquired for each volunteer, the average heart rate and heart rate variations (represented using CoV)

were calculated based on the 11 image acquisition time stamps shown Figure 6-1. After the multi-coil radial data was reconstructed using compressed sensing for radial T1-T2[162] or parallel imaging for MOLLI to generate magnitude and phase images, each pixel from the generated images was independently used as input to the corresponding DeepBLESS network. The conventional BLESSPC was also applied to reconstruct T1/T2 maps for comparison. ROIs were drawn in the entire left ventricular myocardial region for the radial T1-T2 mapping and in the septal region in the MOLLI T1 maps. The mean of the T1/T2 values within ROIs were calculated.

6.2.8 Data Analysis

For statistical analysis, two-tailed Student's t-tests were used for pair-wise comparisons. A p value < 0.05 was considered statistically significant. T1/T2 estimations by DeepBLESS and BLESSPC were compared using the Pearson's correlation and Bland–Altman analysis for simulation, phantom and in vivo studies.

6.3 Result

6.3.1 Simulation Study

For the radial T1-T2 sequence, the mean absolute T1 and T2 percent error using DeepBLESS trained at different noise levels (SNR = 11.1, 20, 100 and composite SNRs = 11.1 - 100) compared with BLESSPC as a function of the SNR of the testing data set are shown in Figure 6-3. Results showed that while the T1 and T2 estimation error was increased when the SNR in the testing set was reduced, adding more noise in the training dataset helped reduce the T1 and T2 estimation error when the testing SNR was lower. In Figure 6-3, the mean absolute T1 and T2 error curves based on the training data with composite SNR were similar to that with SNR = 20, and both curves resembled the BLESSPC curve more than the models trained based on data with SNR = 100 and SNR = 11.1. The detailed T1/T2 estimation data for Figure 6-3 is shown in Table A-4 of the

Appendix VII, which shows that SNR = 20 trained model generated the lowest average T1 and T2 estimation error compared to the other three models. Therefore, for the radial T1-T2 sequence, we chose to use the DeepBLESS model trained with SNR = 20 for phantom and in vivo studies.

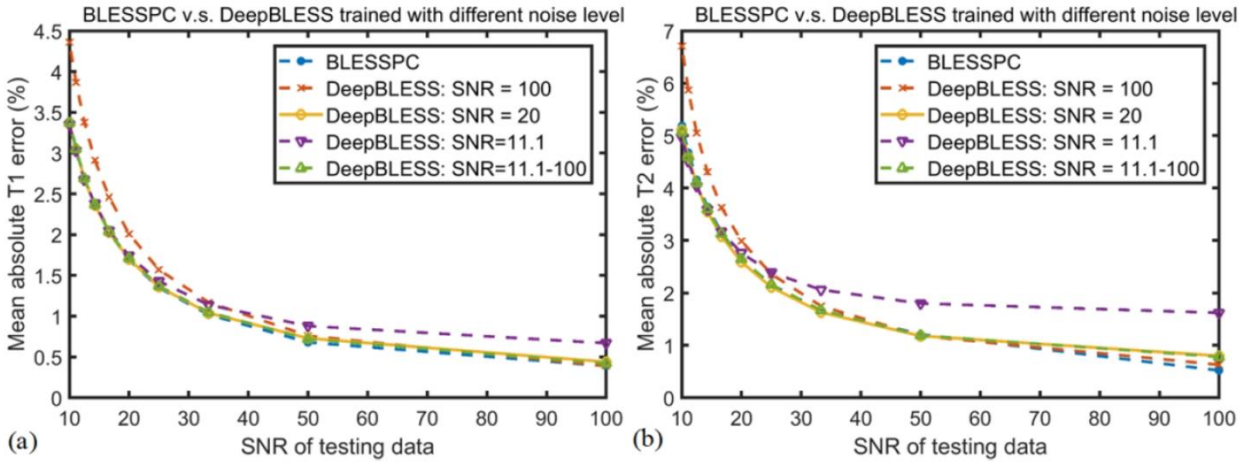

Figure 6-3. The mean percentile absolute T1 (a) and T2 (b) reconstruction error as a function of the testing data noise level (SNR = 10 - 100) for radial T1-T2 mapping using the 4 models trained based on training data with different added noise (SNR = 11.1, 20, 100 and composite SNRs 11.1 - 100), in comparison with conventional BLESSPC.

Figure 6-4 shows a comparison of the T1/T2 estimation results using DeepBLESS (trained with 5% Gaussian noise, SNR = 20) and BLESSPC on testing data with SNR = 20. DeepBLESS values were in excellent agreement with BLESSPC (T1: bias = 0.5 ms, upper 95% limits of agreement = 9.9 ms, lower 95% limits of agreement = -9.0 ms; T2: bias = -0.1 ms, upper 95% limits of agreement = 1.9 ms, lower 95% limits of agreement = -2.0 ms). The correlation coefficient between DeepBLESS and BLESSPC was 1.0000 for T1 estimations and 0.9996 for T2 estimations. For testing data sets (SNR = 20) with at least 1 missed heartbeat, DeepBLESS still agreed well with BLESSPC with similar bias and 95% limits of agreement (see Figure A-8 of the Appendix VII). Example features of DeepBLESS T1 and T2 models for a sample (BLESSPC T1 = 1361 ms,

T2 = 37.7 ms) of the testing set (SNR = 20) simulated based on the radial T1-T2 sequence were shown in Figure A-9 of the Appendix VII.

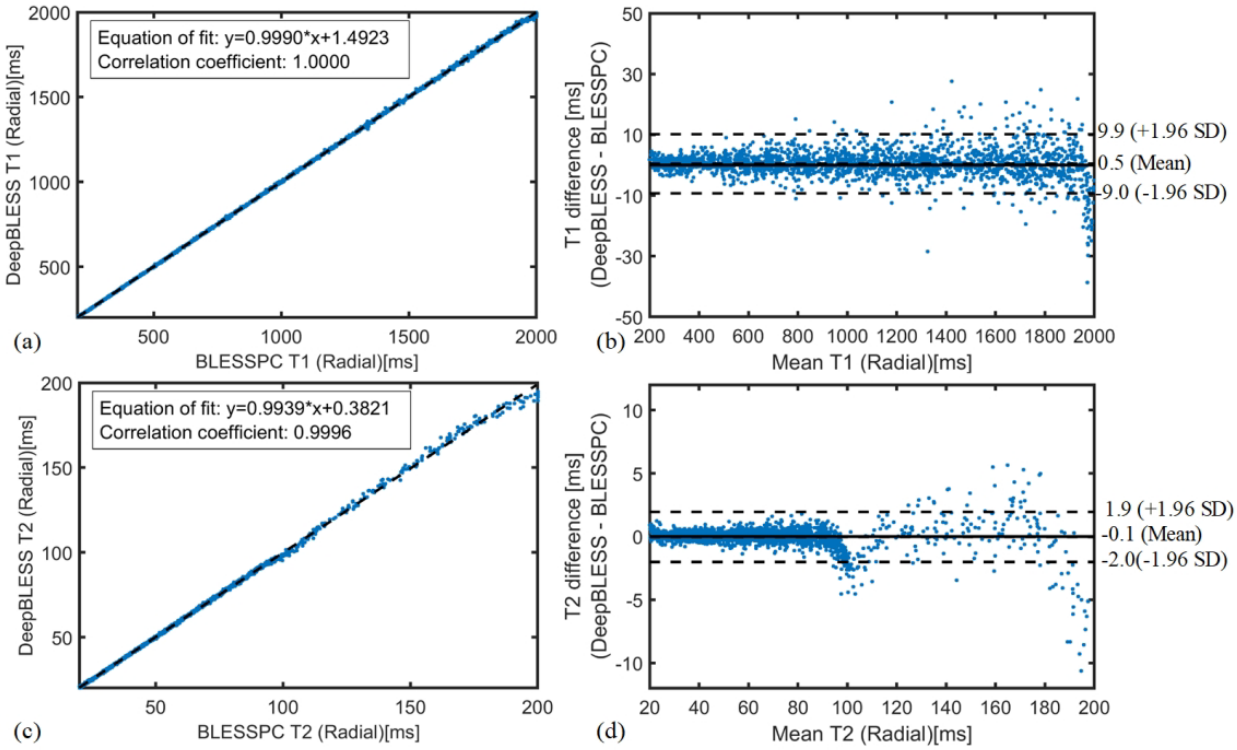

Figure 6-4. Simulation results for radial T1-T2 mapping: Comparison of the T1/T2 estimation results using DeepBLESS (trained with 5% Gaussian noise, SNR = 20) and BLESSPC by plotting DeepBLESS against BLESSPC with equation of fit plot (a for T1 and c for T2) and Bland Altman analysis (b for T1 and d for T2).

6.3.2 Phantom Study

For both the radial T1-T2 and MOLLI sequences, DeepBLESS generated consistent T1 and T2 estimations for heart rates from 40 bpm – 100 bpm, with a maximum standard deviation of 4.5 ms for T1 and 0.6 ms for T2. For both T1 and T2 estimations, DeepBLESS achieved similar accuracy and precision compared to BLESSPC, as shown in Table 6-1. DeepBLESS and BLESSPC both generated accurate T1 and T2 estimations. For radial T1-T2 mapping at 3.0T, the average estimation errors over the 8 phantoms using DeepBLESS were 1.2 ± 4.5 ms (percent error: -0.1%±1.0%) for T1 and -0.1±1.3 ms (percent error: -0.2%±3.3%) for T2. In comparison, the

average estimation errors using BLESSPC were -0.8±4.6 ms (percent error: -0.3%±0.9%) for T1 and -0.3±1.3 ms (percent error: -0.4%±3.2%) for T2. For MOLLI T1 mapping at 1.5T, DeepBLESS and BLESSPC generated T1 estimation errors of -0.2±18.1 ms (percent error: -0.1%±1.7%) and -1.1±18.4 ms (percent error: -0.1%±1.8%), respectively. Regarding precision, both DeepBLESS and BLESSPC had similar CoV (0.8%±0.1% for both radial T1 and MOLLI T1, and 1.3%±0.2% for radial T2, all $p > 0.05$).

Table 6-1. Phantom Study: Average accuracy and precision of BLESSPC and DeepBLESS for the radial T1-T2 and MOLLI sequences using the standard spin-echo sequence as reference.

Parameter	**Method**	**Accuracy**		**Precision**
		Error	Percent Error	Mean CoV
Radial T1	BLESSPC	-0.8 ± 4.6 ms	-0.3%±0.9%	-0.8%±0.1%
	DeepBLESS	1.2 ± 4.5 ms	-0.1%±1.0%	-0.8%±0.1%
Radial T2	BLESSPC	-0.3 ± 1.3 ms	-0.4%±3.2%	1.3%±0.2%
	DeepBLESS	-0.1 ± 1.3 ms	-0.2%±3.3%	1.3%±0.2%
MOLLI T1	BLESSPC	-1.1 ± 18.4 ms	-0.1%±1.8%	0.8%±0.1%
	DeepBLESS	-0.2 ± 18.1 ms	-0.1%±1.7%	0.8%±0.1%

Figure 6-5 shows a pixel level comparison of the DeepBLESS and BLESSPC T1/T2 estimations for radial T1-T2 and MOLLI T1 mapping at selected ROIs by plotting DeepBLESS values against BLESSPC values and using Bland Altman analysis. Similar to our simulation results, DeepBLESS values were in excellent agreement with BLESSPC (Radial T1: bias = 0.1 ms, upper 95% limits of agreement = 3.9 ms, lower 95% limits of agreement = -3.8 ms; radial T2: bias = 0.2 ms, upper 95% limits of agreement = 1.0 ms, lower 95% limits of agreement = -0.9 ms and MOLLI T1: bias = -0.5 ms, upper 95% limits of agreement = 3.7 ms, lower 95% limits of agreement = -4.8 ms).

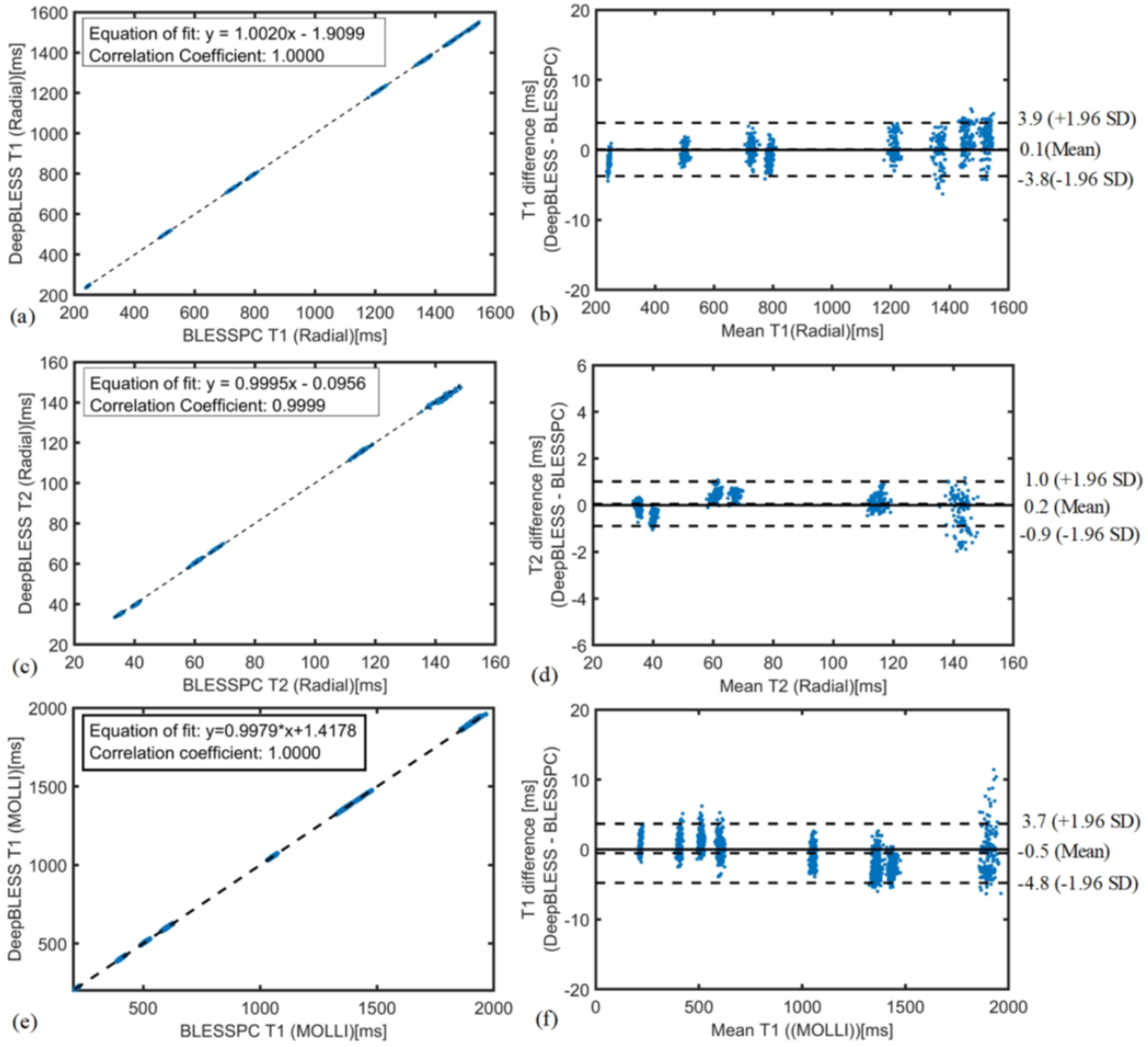

Figure 6-5. Phantom study results for both radial T1-T2 mapping acquired at 3.0T (a-d) and MOLLI (e-f) acquired at 1.5 T, comparing DeepBLESS vs. BLESSPC. Each data point corresponds to a pixel within the phantom.

Figure 6-6 shows an example of radial T1 and T2 maps by DeepBLESS and BLESSPC and their corresponding difference maps. DeepBLESS and BLESSPC provided similar T1 and T2 estimations.

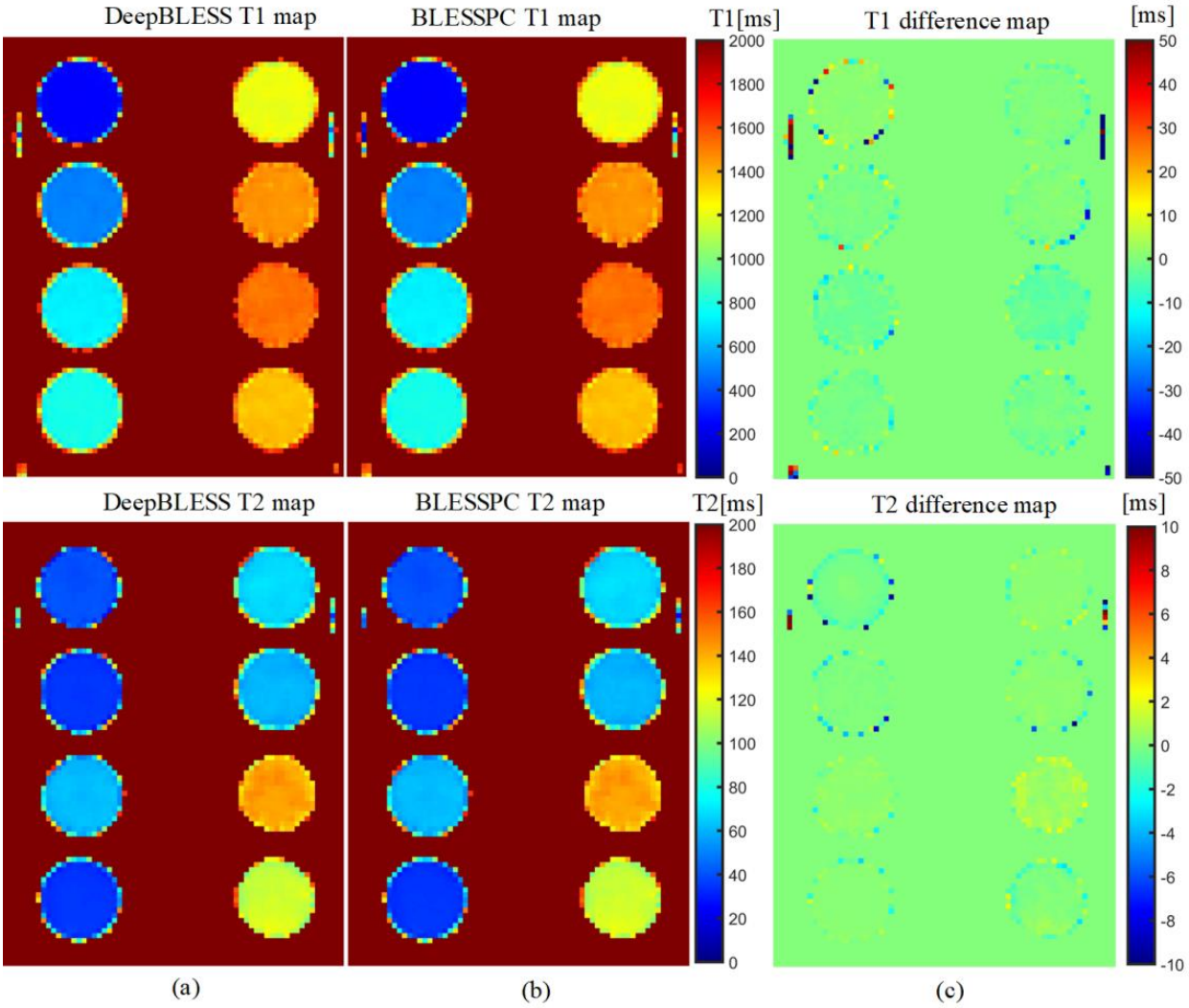

Figure 6-6. Phantom T1 and T2 maps using DeepBLESS (a) and BLESSPC (b) and the corresponding difference maps (c) for the radial T1-T2 mapping sequence acquired at simulated heart rate of 60 bpm. DeepBLESS and BLESSPC generated T1/T2 maps with similar image quality.

6.3.3 In Vivo Study

For the radial T1-T2 sequence, the average heart rate in all 10 healthy volunteers was 62.6 ± 7.8 bpm (min HR = 50.1 bpm, max HR = 77.2 bpm). The average heart rate variation (CoV) was 5.5% ± 8.1% (min CoV = 0.4%, max CoV = 27.9% due to a skipped heartbeat, second max CoV

= 7.2%). DeepBLESS and BLESSPC provided similar myocardial T1 and T2 values at 3.0T (T1: 1366 ± 31 ms for both DeepBLESS and BLESSPC, $p > 0.05$; T2: 37.4 ms ± 0.9 ms for both DeepBLESS and BLESSPC, $p > 0.05$) in all 10 healthy volunteers studied. The correlation coefficients between DeepBLESS and BLESSPC values were 0.9993 and 0.9984 for radial T1 and T2 (Figure 6-7(a) and 6-7(c)), respectively.

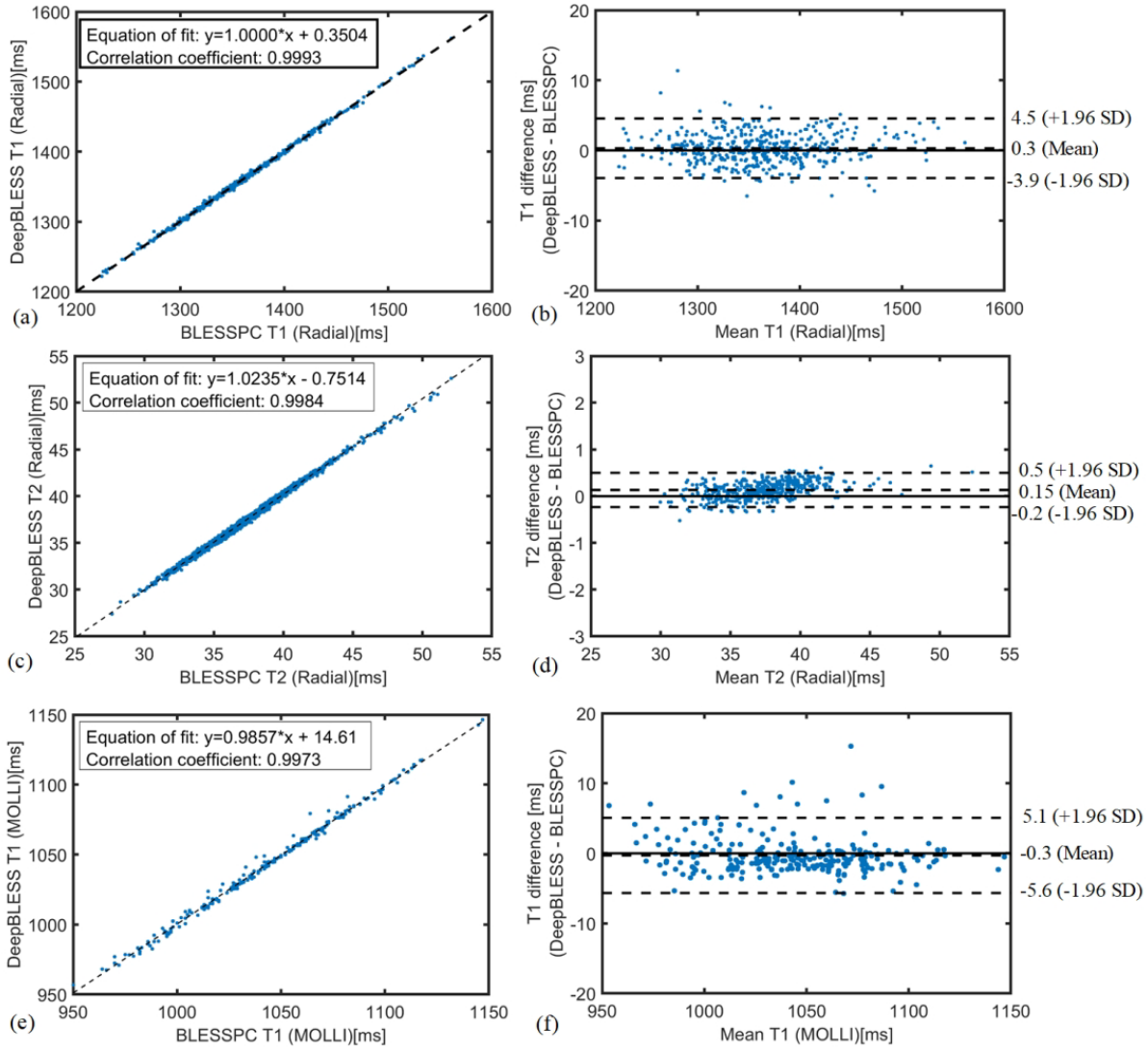

Figure 6-7. In vivo study results for both radial T1-T2 mapping acquired at 3.0T and MOLLI acquired at 1.5T: pixel level comparison of the T1/T2 estimation results in the myocardium using DeepBLESS and BLESSPC by plotting DeepBLESS against BLESSPC with equation of fit plot (a for radial T1, c for radial T2, and e for MOLLI T1) and Bland Altman analysis (b for radial T1, d for radial T2, and f for MOLLI T1).

Bland Altman analysis (Figure 6-7(b) and 6-7(d)) demonstrates that DeepBLESS and BLESSPC T1 and T2 values were in excellent agreement in vivo (radial T1: bias = 0.3 ms, upper 95% limits of agreement = 4.5 ms, lower 95% limits of agreement = - 3.9 ms; radial T2: bias = 0.15 ms, upper 95% limits of agreement = 0.5 ms, lower 95% limits of agreement = -0.2 ms). Figure 6-8 shows example T1 and T2 maps generated using DeepBLESS and BLESSPC and their difference maps in two healthy subjects, one subject without skipped heartbeat (Subject A) and one with a skipped heartbeat (Subject B). For Subject B, there was a missed heartbeat after the 6th data acquisition. For both volunteers, the T1/T2 difference between DeepBLESS and BLESSPC in the myocardial region was negligible.

For the MOLLI sequence, DeepBLESS and BLESSPC generated similar myocardial T1 values at 1.5T (T1 = 1044 ± 20 ms for both DeepBLESS and BLESSPC, $p > 0.05$) in all 8 volunteers studied. Correlation coefficient and Bland Altman analysis (Figure 6-7e and 6-7f) demonstrate that DeepBLESS and BLESSPC values were in good agreement for in vivo MOLLI T1 mapping (correlation coefficient = 0.9973, bias = -0.3 ms, upper 95% limits of agreement = 5.1 ms, lower 95% limits of agreement = -5.6 ms). Figure 6-9 shows example T1 maps generated using DeepBLESS and BLESSPC for the MOLLI sequence in a healthy subject. In this subject, the average difference between DeepBLESS and BLESSPC T1 values in the entire LV myocardial region was -0.5 ± 1.7 ms.

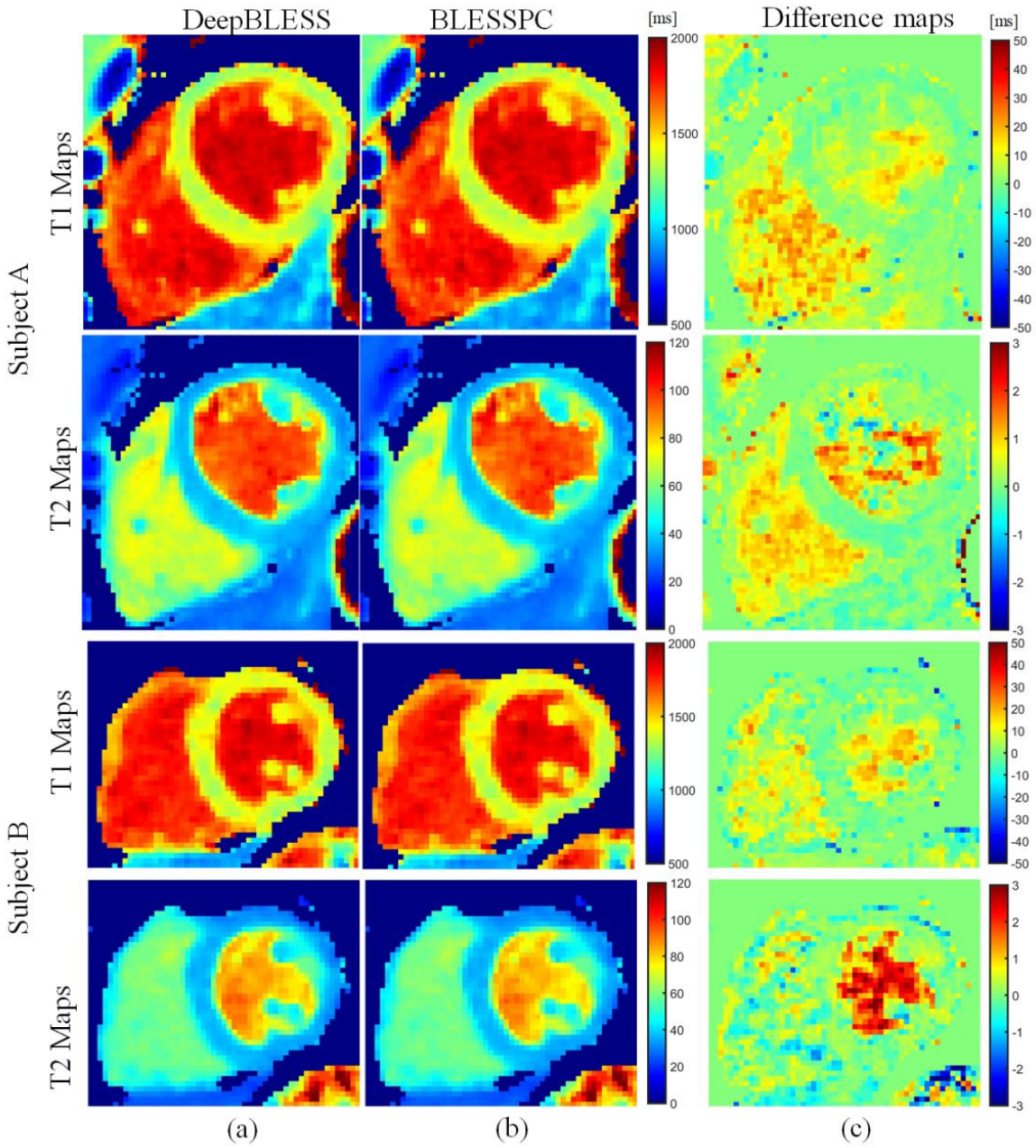

Figure 6-8. In vivo radial T1-T2 mapping acquired at 3.0T: examples of T1 and T2 maps generated using DeepBLESS (a) and BLESSPC (b) and the corresponding difference maps (c) in two healthy subjects. Subject A had no skipped heartbeat while Subject B had a skipped heartbeat after the 6th data acquisition. For both subjects, the maps generated by DeepBLESS and BLESSPC were similar in the myocardium.

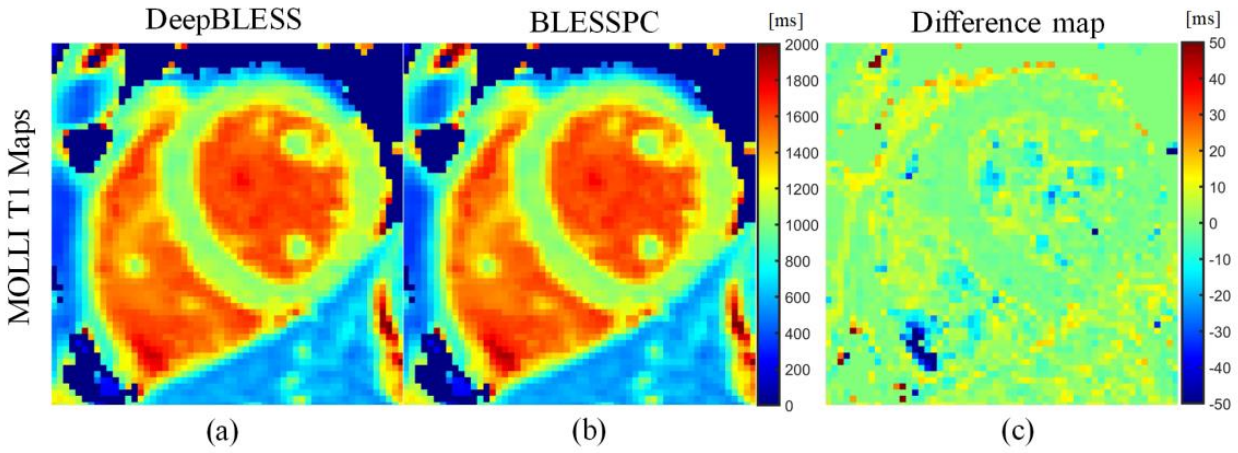

Figure 6-9: In vivo MOLLI T1 mapping acquired at 1.5T: example of T1 maps generated using DeepBLESS (a) and BLESSPC (b) and the corresponding difference map (c) in a healthy subject. All the pixels that BLESSPC did not fit well (R2 < 0.98) were set to 0 for all the corresponding maps. The maps generated by DeepBLESS and BLESSPC were similar in the heart region. In the left ventricular myocardial region, the average T1 difference between DeepBLESS and BLESSPC was -0.5 ± 1.7 ms.

To compare the computation speed, a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz) was used for all the T1 and T2 maps reconstruction (BLESSPC and DeepBLESS), and a single thread was used for a fair comparison. For radial T1-T2, after compressed sensing image reconstruction, a slice of T1 and T2 maps could be generated in ~3.0 hours using BLESSPC [algorithm B in[162]]. In comparison, DeepBLESS was able to reconstruct a slice of T1 and T2 maps in ~0.6 seconds, achieving up to 18,000-fold acceleration. For MOLLI, a slice of T1 map could be generated in ~98 seconds using BLESSPC and ~0.2 seconds using DeepBLESS, achieving 490-fold acceleration by DeepBLESS.

6.4 Discussion

In this work, we studied the use of a deep convolutional neural network to learn the Bloch equation simulations (DeepBLESS) to replace the previously reported Bloch equations based approach (BLESSPC) for rapid myocardial relaxation parameter prediction. Conventional Bloch equation simulations based approaches enable accurate myocardial relaxation parameter

estimation at the cost of increased reconstruction time[32,165]. The proposed DeepBLESS approach enabled almost instantaneous estimation of myocardial relaxation parameters by offloading the time-consuming task of Bloch equation simulations to off-line. Our results show that, for the radial T1-T2 sequence, DeepBLESS could achieve 18,000 times acceleration while achieving similar accuracy and precision compared to BLESSPC. DeepBLESS was also trained for the standard MOLLI 5-(3)-3 sequence for rapid T1 estimation. Our phantom and in vivo results demonstrated that the T1 values generated using DeepBLESS agreed well with those generated using BLESSPC. We previously reported BLESSPC reconstruction time of 6 seconds when using a spoiled gradient echo readout[171]. However, a full simulation of the bSSFP readout in MOLLI needed 98 seconds to generate a slice of T1 map while the sequence scan time was only approximately 10 seconds. The relatively long reconstruction time could be a roadblock for widespread clinical utility. DeepBLESS reduced the T1 map reconstruction time from 98 seconds to 0.2 seconds. As the MOLLI image reconstruction used parallel imaging with reconstruction time<1s, we expect our technique to immediately enable fast and online image reconstruction and T1 calculation for MOLLI.

For simultaneous myocardial T1 and T2 mapping, besides the radial T1-T2 mapping sequence, several other techniques have been proposed1[60,164,177–181]. The potential benefits of using radial T1-T2 mapping over other joint T1 and T2 mapping techniques have been well described in[162]. Specifically, most of the techniques used Cartesian acquisition1[60,177–180], limiting the number of images that can be reconstructed for parameters fitting and therefore can potentially suffer from reduced precision. The average myocardial T1 values measured at 3.0T using the multitasking[181] native T1 and T2 mapping sequence (1216 ± 67 ms) was lower than the standard MOLLI (1244 ± 48ms), which itself has been known to underestimate T1. As for cardiac MRF, improvements have

been made to improve its accuracy by also considering the effect of imperfect slice profile, inversion and T2 preparation[165] and the T1/T2 calculation speed using deep learning[170]. However, all of these improvements still used the relatively long 16-heart beat version cardiac MRF sequence with relatively long acquisition window (240 – 280 ms). In comparison, the radial T1-T2 mapping technique requires only 11 heartbeats and a shorter window (~200 ms). In[182], it has been stated that the acquisition window may potentially be reduced to 150ms and the breath-hold time may be reduced to 5 heartbeats. However, the T1/T2 measurement accuracy/precision using this shortened version of cardiac MRF sequence remains to be evaluated. For shortened cardiac MRF sequences, the precision and reproducibility need to be evaluated carefully due to limited data acquired, which may potentially reduce the parameter estimation precision. In comparison, we show that the radial T1-T2 mapping sequence can achieve similar precision and reproducibility as the widely used MOLLI sequence and conventional cardiac T2 mapping sequence[162]. While there are potential benefits of using radial T1-T2 mapping over the other simultaneous myocardial T1-T2 mapping techniques, further studies are warranted to compare the radial T1-T2 sequence with the other techniques in clinical applications. While DeepBLESS achieved almost instantaneous T1/T2 map reconstruction for the radial T1-T2 mapping sequence, the compressed sensing reconstruction took approximately 3 min, a limitation for using the radial T1-T2 sequence for simultaneous myocardial T1 and T2 mapping. Recent studies have shown that deep learning can be applied to replace compressed sensing to reduce reconstruction time[7,8,183]. These techniques may be combined with our proposed T1 calculation technique to further reduce total imaging time and enable online use of the radial T1-T2 mapping technique.

Recently, deep learning models have been applied to MRF for fast quantitative parameters prediction, such as the MRF deep reconstruction network (DRONE)[168]. However, this model only

considers the measured signal as the input, and are only applicable to the sequence that has fixed acquisition timing. For parameter quantification in cardiac applications, the actual image acquisition timing varies due to patient-specific heart rate variations. To be adaptive to heart rate variations, DeepBLESS used both the image acquisition time stamps and imaging signal as the input for cardiac parameters prediction. To ensure the robustness of DeepBLESS to various heart rate variations, we included variable heart rates in our model training data. Our results demonstrate that DeepBLESS agrees well with BLESSPC for various heart rates. Regarding the deep learning model used in this work, we choose the 3 × 1 size 1-D filter due to the following two reasons: 1) The input layer size is relatively small (11 × 1 for each channel); therefore, using 3 × 1 size 1-D filters should be sufficient. 2) For the same number of trainable parameters, a smaller filter size with deeper network is in general better than a larger filter size with shallower network. Recently, deep learning has been applied to automatic segmentation of cardiac T1 images[28]. These techniques could potentially be combined with our proposed technique to further improve reliability and efficiency.

Instead of comparing DeepBLESS to conventional dictionary matching approaches[168], DeepBLESS was compared with BLESSPC, an optimization approach based on the non-linear least squire fitting[32,171]. The benefit of BLESSPC over the dictionary matching approach is that there is no need to generate a large dictionary, which requires more computer memory, and the accuracy and precision is not limited by the size of the dictionary. For cardiac applications, both BLESSPC and the dictionary matching approach need Bloch equation simulations after the sequence was performed so that the image acquisition timing is known for simulation. DeepBLESS performs the time consuming task of Bloch equations simulations during the offline training stage, which learns the non-linear mapping from acquisition time and signal to relaxation

parameters, allowing for fast, accurate and precise relaxation parameter calculation. After the model was trained, the speed of DeepBLESS for parameter calculation is not affected by how detailed the sequence was simulated using Bloch equation simulations; while the conventional approaches such as BLESSPC or dictionary-matching approaches[164] are substantially affected. Therefore, when more details are considered in Bloch equation simulations, DeepBLESS may achieve more acceleration compared to these conventional approaches. For instance, BLESSPC simulates more details in radial T1-T2 mapping compared to the MOLLI sequence, including simulating the adiabatic inversion pulse and multiple T2-prep pulses. As such, DeepBLESS achieved more reconstruction time acceleration over BLESSPC for radial T1-T2 mapping (18,000 times) than for MOLLI (490 times). DeepBLESS is even more promising for applications that require more timing consuming simulations, such as in incorporating the effect of magnetization transfer effects in parameter mapping, which was not considered in this study, but has been shown to have an effect on myocardial T1 underestimation using inversion recovery based sequences[32,184].

Previous study has shown that the deep learning approach can help to reduce T1 and T2 estimation errors compared to the conventional approach when the noise is higher for MRF[177]. However, we did not see obvious improvement using DeepBLESS over BLESSPC regarding T1/T2 errors for cardiac T1 and T2 mapping, despite our demonstrated advantage compared to the DRONE network for cardiac applications (See Table A-1 of the Appendix VII). This may due to the following reasons: (1) BLESSPC, due to its comprehensive simulation of essentially every aspect of the pulse sequence, without the need for building a dictionary, may already minimize T1/T2 errors. It is not subject to dictionary size issues associated with the dictionary matching approach that the DRONE network compared to. (2) For cardiac applications, the image

acquisition timing need to be considered, which may have been more complex for the deep learning network to learn.

To train the network, different from the conventional approach of using a predetermined series of parameters (e.g. T1, T2) with predetermined step sizes to generate the training data set, we randomly sampled a set of parameters (e.g. T1, T2, FA, heart rate) in a certain range for each simulation. This has two benefits: 1) flexible training, validation, and testing data size setup; and 2) Compared with the conventional approach, the proposed approach will generate more different T1, T2, FA, and heart rate values for training. For instance, if the conventional approach had n_1 different T1 values, n_2 different T2 values, n_3 different FAs and n_4 different heart rates to generate a training set with $n_1 \times n_2 \times n_3 \times n_4$ samples, for the same number of training samples, the proposed approach will generate $n_1 \times n_2 \times n_3 \times n_4$ different values for each parameter. This may improve the training results. Similar to the non-uniform dictionary sampling in DRONE[167], we sampled T2 more densely in the range of 20 - 100 ms because this is the expected range of the myocardial T2. Sampling more data in the 20-100ms range gives more weights for T2 errors in this range in the training, which may help reduce errors in this T2 range.

In this work, we added Gaussian noise to the training data due to two reasons: (1) It is a common approach used in network training to reduce overfitting; (2) Although the noise-like artifacts from under-sampling were certainly not of Gaussian distribution, it was not possible for us to fully characterize the noise characteristics from under-sampling. Therefore, assuming a Gaussian distribution would be our alternative approach. The results from phantom and in vivo experiments confirmed that the DeepBLESS trained with added Gaussian noise agreed well with BLESSPC. Similarly, a recent machine learning technique for MR finger printing[167] also added Gaussian noise to the training data, while the signal from the MR fingerprint sequence was under-

sampled. Our results show that using different noise levels will affect the final results. For example, Figure 6-3 shows that for noisier testing data sets with SNR range of 10-30, the model trained using less noisy data training data sets with SNR=100 was less accurate than the model trained using noisier data sets with SNR=20. For less noisy testing data with SNR>60, the model trained using less noisy training data had better performance. We could not find a model that is always the best for a wide range of SNR from 10 – 100, therefore in this work we chose the model that generated the lowest average T1 and T2 estimation error.

In order to demonstrate the proposed network can be adaptive to different cardiac T1/T2 mapping sequences, we used the same network (except the input) for both radial T1-T2 and MOLLI sequences. Since the MOLLI sequence was simpler than the radial T1-T2 sequence, it is possible to use a shallower network with fewer parameters for MOLLI. However, based on the results from Table A-2 of the Appendix VII, even for the MOLLI sequence, the proposed network with 4 Resnet blocks was still better than the less deep networks using 0 or 2 Resnet blocks, indicating that a deeper network can still help to achieve better results for MOLLI. There was no obvious overfitting using the same network for MOLLI, potentially due to the large training data sets available for MOLLI network training (1 million training data sets for only 31,000 trainable parameters).

The current annealing approach used in this work is a simple version of traditional step decay annealing approaches (i.e. only one step decay) with slight modification. We choose to use it because with it we can tune the first learning rate and epoch number to obtain the best MSE, load the model with best validation MSE, and tune the second learning rate and epoch number to further improve the results. In comparison, the conventional step annealing approaches does not load the best model when reducing the learning rate and the number of training epochs is fixed for each

step. Our results in the Appendix VII show that the proposed annealing approach generated better results than the conventional step decay and exponential decay for the hyper-parameters tested. However, it does not mean that the proposed annealing approach is the most accurate way.

We point out that the MOLLI sequence was performed on a 1.5T scanner only and the radial T1-T2 sequence was performed on a 3.0T scanner only. The measured average myocardial T1 values using MOLLI with BLESSPC/DeepBLESS at 1.5T (1044 ± 20) was higher than the conventional MOLLI T1 values at 1.5T (950 ± 21)[185]. The measured average myocardial T1 values using radial T1-T2 with BLESSPC/DeepBLESS at 3.0 T (1366 ± 31 ms) was also higher than the conventional MOLLI T1 values at 3.0T (1052 ± 23 ms)[185]. These are expected as conventional MOLLI fitting is known to underestimate T1 values. The measured average myocardial T2 values using radial T1-T2 with BLESSPC/DeepBLESS at 3.0 T (37.4 ms ± 0.9 ms) was similar to that measured by cardiac MRF with slice profile, preparation pulse efficiency, and B1+ corrections (37.2 ms ± 1.5 ms) in[165].

In this work, we focused on myocardial T1/T2 measurements. The blood T1/T2 measurements based on our technique needs to be further evaluated because blood flow was not simulated when building our models. Due to blood flow, the BLESSPC-fitted apparent flip angle in the blood region were much lower than that in the myocardial region (usually < 3° for radial T1-T2), while in the DeepBLESS training data, we only simulated a reasonable apparent flip angle range (3°-8° for radial T1-T2). This could be the main reason why there were larger T1 differences between BLESSPC and DeepBLESS in the blood region. It is possible to simulate the blood flow to improve blood T1/T2 estimation accuracy, and this could be a potential benefit of DeepBLESS over BLESSPC as adding additional simulations should not affect its T1/T2 calculation speed. In BLESSPC for MOLLI, a fixed T2 = 45 ms was assumed in the Bloch simulation to avoid the need

for fitting T2. In comparison, for DeepBLESS, a wide range of T2 was simulated, which may potentially be more accurate. The difference map in Figure 6-9 for MOLLI T1 mapping shows larger T1 differences at edges of the heart between BLESSPC and DeepBLESS, which may be due to cardiac motion, blood flow and off-resonance, as these effects were not considered in the DeepBLESS model.

Our study has limitations. While DeepBLESS with the proposed network and hyper-parameters was relatively optimal compared to other network or parameters tested in this work, the current network with the proposed hyper-parameters and learning rate strategies may not be the most optimized one, as it was not possible to evaluate all possible networks, hyper-parameters and training strategies. Nevertheless, we reached the main goal that the proposed DeepBLESS approach can achieve similar accuracy and precision compared to BLESSPC while greatly reducing the reconstruction time. As DeepBLESS can be trained on data with noisy data, it can potentially be better than BLESSPC for low SNR data. Further studies are warranted to further optimize the DeepBLESS network and training strategies to achieve better results than BLESSPC. DeepBLESS was trained for heart rates between 40 – 100 bpm with 10% variations in cardiac cycle lengths. It is conceivable that a model training based on larger variations in cardiac cycle lengths could be applied for T1 and T2 mapping for patients with arrhythmias. However, a number of other issues need to be addressed, including motion artifacts, cardiac morphology changes due to varying pre-load and after-load conditions, which are beyond the scope of the current study. The heart rates between 40 – 100 bpm is suitable for most of the cardiac applications, and for applications out of the current trained range, we can potentially fine tune DeepBLESS using the training data with a larger range of heart rates and beat-to-beat variations. This study was performed in a small cohort of healthy volunteers at mid-ventricular slice location only. Further

clinical evaluations on larger cohorts are warranted to evaluate the performance of DeepBLESS. A limitation of training our DeepBLESS network based on simulated data is that it may not entirely reflect the complexity of the in vivo environment. This limitation is not DeepBLESS specific, and it is also true with conventional Bloch equations based approaches, such as BLESSPC and MR fingerprinting. Based on our in vivo data, we have shown satisfactory T1/T2 accuracy using our network. The simulation data essentially enable the network to learn the non-linear Bloch equation, which is the foundation for in vivo MRI. Therefore, it would not be surprising that our model worked well for our in vivo studies.

6.5 Conclusion

In conclusion, DeepBLESS offers an almost instantaneous approach for estimating relaxation parameter maps with good accuracy and precision similar to the conventional Bloch equation-based approach (BLESSPC). The acceleration provided by DeepBLESS is promising for multiparametric mapping in cardiac applications.

Chapter 7 Automatic Peripheral Arteries and Veins Segmentation

This chapter aims to develop an automated platform for segmentation of the arteries and veins in the lower extremities, i.e., thigh and calf, from the Ferumoxytol-enhanced MR angiography (FE-MRA) images. To achieve our goal, we implemented a two-staged based platform in which in the first stage, by using a deep neural network, we extracted the blood vessels, and in the second stage, we used time-resolved images to initially label the arteries, and then we applied a region growing algorithm to complete the artery/vein segmentation process. We made comprehensive comparisons for both the blood vessel segmentation task and the artery/vein separation task. We also performed a quantitative and qualitative evaluation to ensure that the developed platform could potentially translate to the clinics. Our proposed platform can perform the segmentation process automatically in less than 4 minutes. Also, it could potentially reduce the inter-observer variability. A version of this chapter has been published[12] in the Magnetic Resonance in Medicine:

1. Vahid Ghodrati, Yair Rivenson, Ashley Prosper, Kevin de Haan, Fadil Ali, Takegawa Yoshida, Arash Bedayat, Kim-Lien Nguyen, J. Paul Finn, Peng Hu. Automatic segmentation of peripheral arteries and veins in ferumoxytol-enhanced MR angiography. Magn Reson Med. 2021; 00: 1– 15. doi:10.1002/mrm.29026

7.1 Introduction

Contrast-enhanced magnetic resonance angiography (CE-MRA) is widely used for the diagnosis of peripheral artery disease[186-192]. One of the first steps involved in CE-MRA post-processing is blood vessel segmentation. For more recent steady-state CE-MRA applications using

intravascular agents such as ferumoxytol[193], the post-processing also involves the additional step of separating the arteries from veins, as both arteries and veins are enhanced in these steady-state acquisitions. The large image data size of high-resolution MRA makes vessel segmentation and separation highly labor-intensive. Therefore, automated algorithms are highly desirable; however, automating blood vessel segmentation is associated with challenges, including hardware imperfections, extreme data imbalance, complex geometry of the blood vessels, and heterogeneous tissue near the blood vessels. Several algorithms have been developed to address the aforementioned challenges[194,195]. These algorithms can be categorized into two groups: 1) classical techniques that rely on combinations of geometry, appearance, and statistical model-based approaches with morphological and handcrafted intensity-based features[194,196-201]. 2) learning-based techniques that can adaptively find highly representative features by training on the data[195, 202-220]. Lei et al. proposed a semi-automatic fuzzy connectedness algorithm for pelvic vessel segmentation[194]. They used fuzzy-connectedness to extract the entire vascular bed from the background, followed by separation of arteries and veins in an iterative process. Shahzad et al. proposed an automatic multi-atlas-based approach to extract and label the arterial skeleton from whole-body MRA images[195]. Their algorithm learns the anatomical knowledge from several atlases to initialize labels for the arterial skeleton, which are then refined via a rule-based approach.

More recently, deep learning-based segmentation has gained attention in medical image segmentation. Abraham et al. proposed a novel multi-scaled attention U-Net with the focal Tversky loss for breast and skin image segmentations[221]. They adapted a 2D attention gated U-Net for lesion segmentation and showed the effectiveness of the focal Tversky loss in handling data imbalance. Automatic vessel segmentation based on 2D-convolutional neural networks (CNNs) has also been explored to segment 2D images of the retina[204,205], computed tomography of the

liver[206], and vascular segmentation for time of flight MRA images of the brain[207]. Even though using 2D-CNNs could potentially provide flexibility in designing high-capacity networks, they cannot take full advantage of the information encoded across the three spatial axes of the images. To extract more powerful volumetric representations, 3D-Fully Convolutional Networks (FCNs) and its variants like 3D U-Net and Volumetric-Net (V-Net) have been proposed and achieved state-of-the-art performance in various medical image analysis challenges[210,212,213]. However, there are still relatively few dedicated deep learning platforms for 3D blood vessel segmentation. These include Uception[218], vessel segmentation and analysis pipeline (VesSAP)[217], and Volume Composition Network (VC-Net)[219]. The Uception technique[218] incorporates 3D inception modules in the convolutional layers of the 3D U-Net to segment the cerebrovascular network from MRA images. Tetteh et al. showed promising accuracy in segmenting blood vessels from brain time of flight (TOF) MRA and synchrotron radiation X-ray tomographic microscopy (μCTA)[220].They trained the network first on synthesized vessel structures, and then refined that based on real annotated datasets via a transfer learning approach. The VC-Net approach incorporates the structural information as an extra input in the neural network to effectively segment high-fidelity 3D sparse microvascular structure[219]. It enhances the volumetric vessel segmentation qualitatively by including the maximum intensity projection (MIP) of the data into the 3D U-Net in a learnable fashion.

To our knowledge, no study has been performed to adapt and apply deep 3D-CNNs to the segmentation of peripheral blood vessels. In this study, we developed a fully automated platform of peripheral vessel segmentation for ferumoxytol-enhanced MRA (FE-MRA) images, with a particular focus on reducing computational time and automating the segmentation process. The proposed platform has two stages: 1) extraction of the peripheral vasculature and 2) classification

of the extracted blood vessels as arteries and veins. For the first stage, we developed a 3D deep neural network (DNN) with an attention-gated 3D U-Net structure and trained it using deep supervision mechanisms. Focal Tversky loss was included to cope with extreme data imbalance, and region mutual information loss (RMI)[222] was used to increase the structural similarity between the ground truth mask and predicted mask. In the second stage, we classified extracted blood vessels to arteries and veins by applying a region-growing algorithm based on the initial seeds obtained from a time-resolved image.

7.2 Methods

7.2.1 FE-MRA Dataset

All the data in this work was acquired as part of a clinically indicated FE-MRA scan and was retrospectively collected for this study under a protocol approved by our institutional review board. Our clinical FE-MRA protocol included an initial series of dynamic, spatially low-resolution, time-resolved volumetric images of the peripheral arteries during the first passage of an initially diluted injection representing1/6th of the total ferumoxytol dose. Subsequently, the remaining dose of ferumoxytol was infused, and a high-resolution MRA was acquired during the steady-state distribution of ferumoxytol, in which the arteries and veins were equally enhanced.

All imaging were performed at 3T (Skyra and Prisma; Siemens Healthcare, Erlangen, Germany) with the following parameters: TR/TE= 2.9ms/1.1ms, flip angle=16o, matrix size $\approx$ $400 \times 1200 \times 212$ mm^3, slice thickness=1mm, and in-plane resolution = 0.7×0.7mm^2. Among 45 retrospectively-obtained FE-MRA, seven volumes were selected randomly as the test set, and the remaining 38 were used as the training set. An expert radiologist (5+ years of clinical MRI reading) manually segmented the peripheral arteries and veins in the test set. Another expert radiologist (8+

years of clinical MRI reading) reviewed and modified the segmentation masks. An experienced researcher annotated peripheral vessels in the training datasets manually. All input volumes were normalized by subtracting the mean and dividing by twice the standard deviation of all voxel intensity inside each volume.

7.2.2 Deep Convolutional Neural Network Architecture

Figure 7-1 shows an overview of the proposed method. In the first stage, the proposed 3D U-Net + deep supervision (DS) + attention gates (AG) (U-Net+DS+AG) extracts blood vessels from high-resolution FE-MRA datasets. In the second stage, a region growing algorithm separates the arteries from the veins based on initial seeds obtained from the time-resolved imaging volume.

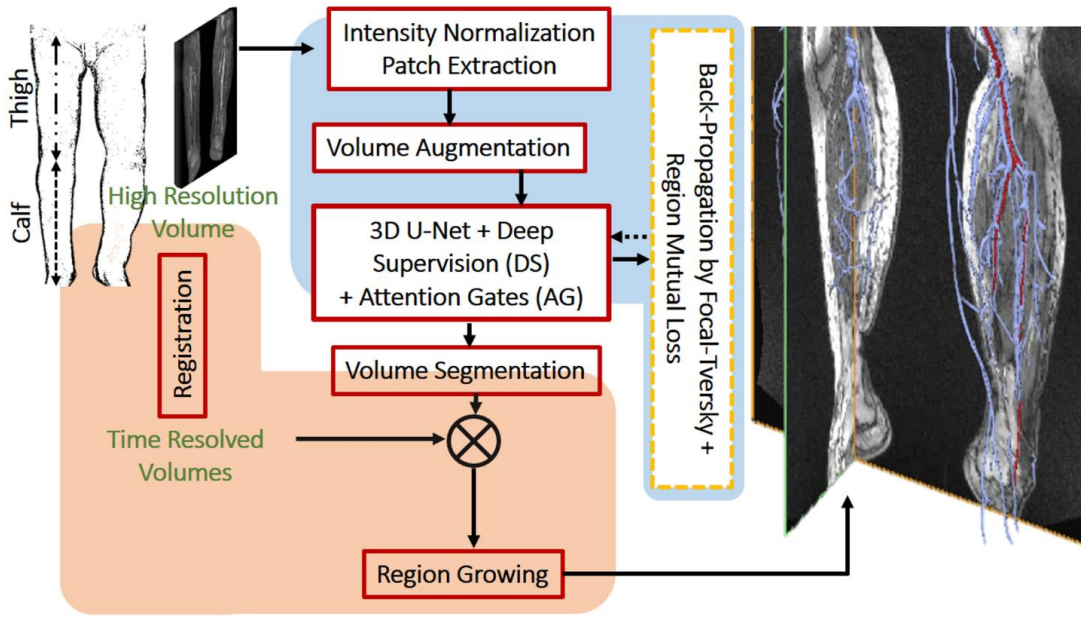

Figure 7-1. Overview of the proposed peripheral blood vessel segmentation and artery/vein separation platform for FE-MRA. Steps in the blue region occur during the blood vessel segmentation stage, where our 3D segmentation neural network extracts the blood vessels from the high-resolution FE-MRA. Steps in the orange region represent the subsequent artery/vein separation stage, where time-resolved imaging volumes are used to initiate the arterial branches followed by application of a region growing algorithm to separate the arteries from the veins.

Figure 7-2 shows the network structure, which comprises three distinctive modules added to a generic 3D U-Net structure: 1) a pyramid of input modules, 2) local AG modules, and 3) a multi-level auxiliary DS module.

7.2.3 3D U-Net structure with pyramid of input images

Vessels in the lower extremities vary in diameter from approximately 20 pixels to 1-2 pixels. Therefore, exploiting multi-level information can potentially be effective in vessel extraction. U-Net's capability to learn features at multiple scales without increasing the depth of the network makes it a well-suited choice for our task. As shown in Figure 7-2, U-Net consists of two paths: (I) the encoder path, which contains four downsampling stages, (II) the decoder path, which includes four up-sampling stages.

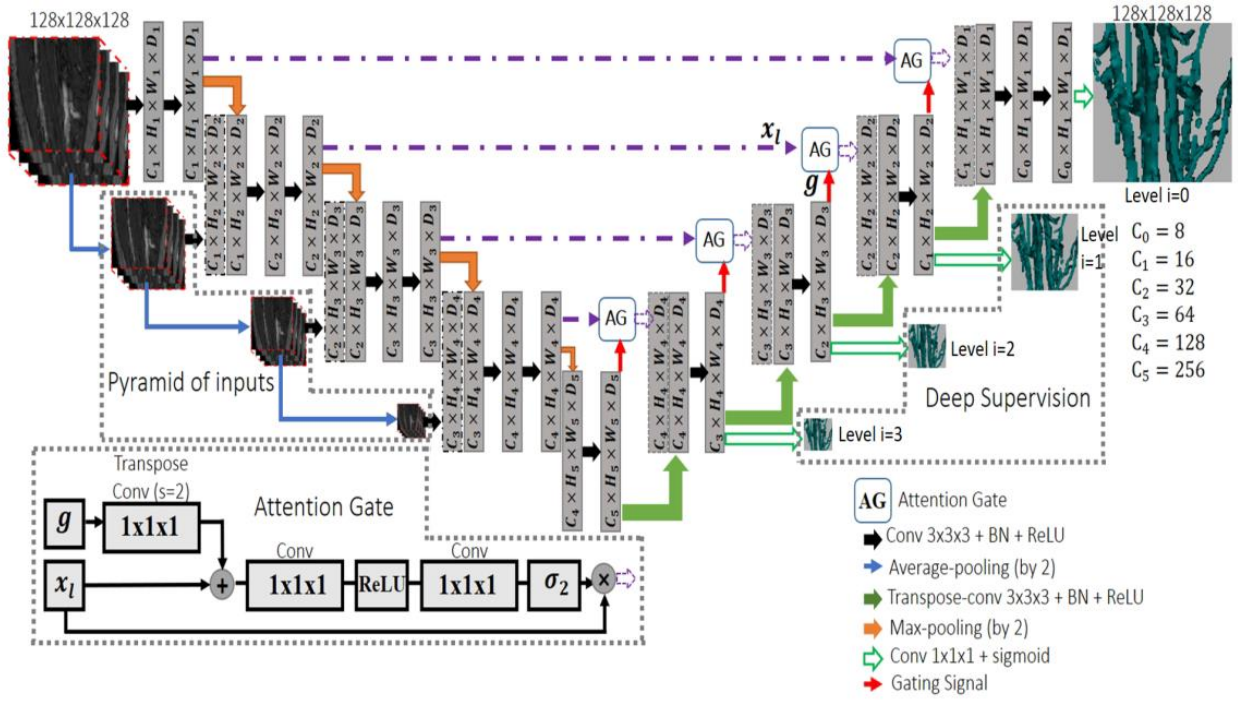

Figure 7-2. Detailed network architecture used in this work. The segmentation network is a modified 3D U-Net. It incorporates three main components: 1) pyramid of the input volumes, 2) local attention gates (AG), and 3) deep supervision (DS) mechanism. The pyramid of input volumes helps the network to minimize the risk of missing thin branches of the blood vessels. Attention gates force the network to learn

the more relevant features of the blood vessel segmentation. Auxiliary outputs in the multiple levels as a variant of the deep supervision approach facilitate the network training and the AG's parameter updating. They also force the network to learn the more discriminative features. Ci represents the number of the extracted features, and Hi×Wi×Di represents the 3D spatial dimension of the features for network in the level i.

Each stage contains two convolutional layers, with each layer containing learnable convolution filters followed by batch normalization and ReLU. The size of the extracted features after each convolutional layer is denoted as Channel × Height × Width × Depth (C × H × W × D), in which C represents the number of extracted features, and H × W × D corresponds to the spatial dimensions of the features. Due to the limited memory of the GPU, the first stage of the network exploits 16 convolutional kernels, and the number of kernels doubles in each deeper stage. Stages in the encoder and decoder paths are connected via max-pooling and transpose convolution. The pyramid of inputs was concatenated to incoming features from the previous stage to reduce the risks of missing features of the thin blood vessels.

7.2.4 Local Attention Gate

Attention mechanisms in image segmentation tasks guide the network to highlight salient features corresponding to the region of interest. We incorporated soft AGs to weigh the extracted features from the encoder path, before propagating them to the decoder path. The structure of the AG module used in this work, adapted from Schlemper et al.'s work[216], is shown in Figure 7-2. Suppose x_L represents the set of activation maps of a given layer L. Each component of x_L is a F-vector, where F is equal to the number of features. The AG computes a coefficient F-vector ($0 \leq \alpha_i \leq 1$), to put emphasis only on the most task-relevant features. In our work, coarser activation maps, which contain global information, were combined with activation maps from the layer L in a learnable fashion to calculate the attention coefficient vectors for each voxel:

$$q_{attn}^{L} = \Psi_2^T \left(ReLU(\Psi_1^T (x_L + W_g^T + b_g) + b_{\Psi_1})\right) + b_{\Psi_2} \quad (7\text{-}1)$$

$$\alpha_L = \sigma_2(q_{attn}^L(x_L, g)) \quad (7\text{-}2)$$

The linear activation coefficients q_{attn}^L were computed by combining additive operation and trainable parameters $W_g^T, b_g, \Psi_1^T, \Psi_2^T$. The trainable parameters were the weights and bias terms of the convolutional layers in the AGs. As shown in Equation (7-2), a sigmoid function σ_2 was applied to the linear activation coefficients to restrict the range to $[0, 1]$. The activation maps from layer L were pruned through multiplication with attention coefficients α_i and were subsequently concatenated to the corresponding up-sampling stage.

7.2.5 Deep Supervision

Previous studies have shown that DS mechanisms can improve discriminativeness and robustness of learned features in the first layers and address the "vanishing" gradient issues in the training process[215,223]. Therefore, we included auxiliary outputs in the network architecture to supervise the network on multiple levels. Each of the up-sampling stages is supervised via an auxiliary output. As shown in Figure 7-2, auxiliary outputs in the multiple levels are considered to reinforce the propagation of gradient flow. Besides, this method provides gradient flow to the local attention modules and increases their ability to influence the responses to the broad range of image foreground content.

7.2.6 Objective Function

In our task, less than 1 percent of the voxels within a training patch corresponds to blood vessels on average. Various objective functions have been proposed to address the data imbalance issue, such as weighted Cross-Entropy (CE), focal loss (FL)[221,224], and generalized dice function[225]. FL attempts to down-weigh the contribution of a relatively large size of irrelevant background region so CNN can focus more on relevant dense regions of interest. The well-known F1 score (Dice), as a loss function, typically performs better than CE in unbalanced medical image

segmentation problems. Since the F1 is a harmonic mean of precision and recall, it weighs false positive and false negative equally:

$$F1 = \frac{2\times TP}{2\times TP + FN + FP} \quad (7\text{-}3)$$

TP, FN, FP stand for the true positive, false negative, and false positive, respectively. Tversky index F_β as a generalization of the F1 has flexibility in trading off between precision and recall[226].

$$F_\beta = \frac{(1+\beta^2)\times TP}{(1+\beta^2)\times TP + \beta^2 \times FN + FP} \quad (7\text{-}4)$$

Where β weighs the FN and FP and is called the balancing factor between the recall and precision.

For blood vessel segmentation, deep neural networks trained based on the voxel-wise loss functions such as FL, weighted cross-entropy, and generalized dice may struggle in identifying the voxel when its visual evidence is weak; therefore, it is important to consider the strong interdependencies between the voxels in the image. In this work, we used the region mutual information (RMI) loss[222] to consider the dependency between the voxels in the vessel segmentation. The central voxel of a small cubic patch (inside the training patch) with size d (d=R×R×R) and all voxels inside that small patch collectively represent the central voxel in the RMI calculation. For instance, by considering small patch size 3×3×3 (d=27), we use a 27-vector (one central voxel and 26-neighbors) to represent the patch's central voxel. For a 3D image, we have many 27-vectors; thus, the image can be cast into the multivariate distribution of 27-vectors. After getting the multivariate of 27-vectors for the ground truth and the predicted map by the segmentation network, higher-order consistency between the target and predicted map can be achieved by maximizing the mutual information (MI) between their multivariate distributions. Zhao et al. derived a lower bound of the MI and tried to maximize it instead of directly maximizing the MI which is highly memory consuming[222]. Equation (7-5) formulates an approximation of a

lower bound of the MI between a two multivariate distribution of d-vectors y and p from ground truth (Y) and predicted map (P), respectively.

$$MI_l(y,p) \approx \frac{-1}{2d} Tr(\log(M)) \quad (7\text{-}5)$$

$Tr(.)$ is the trace operator and M is defined as Equation (7-6):

$$M = \Sigma_y - cov(y,p)(\Sigma_p^{-1})^T cov(y,p)^T \quad (7\text{-}6)$$

Σ_y is the variance matrix of y and $cov(y,p)$ is the covariance matrix of y and p. To calculate the RMI loss, the sum of the MI values for all d-vectors was divided by the number of the d-vectors inside the training batch.

In this work, we used a linear combination of the Tversky Index and RMI loss as the loss function for the final stage (level i = 0) of the network to achieve higher recall relative to precision and achieve a better structural similarity between the ground truth (Y) and predicted map by the network (P). Due to the computational bottleneck, we used patch size 3×3×3 to calculate the RMI loss. For the intermediate stages (levels i = 1, 2, 3) of the network, focal Tversky loss[221] was considered. Equation (7-7) describes the total loss function used in the network training:

$$\sum_{i=0}^{3} L_i(P_i, Y_i) = (1-\alpha)(-\mathrm{RMI}(P_0, Y_0)) + \alpha(1 - F_\beta(P_0, Y_0)) + \sum_{i=1}^{3}(1 - F_\beta^{\,i}(P_i, Y_i))^{\frac{1}{\gamma}} \quad (7\text{-}7)$$

Where γ is a tunable focusing parameter, L_i, P_i and Y_i are the loss function, predicted map, and target on the level i, respectively. As shown in Figure 7-2, the network has four levels in which the level i=0 shows the highest resolution and the level i=3 represents the lowest resolution level. The trade-off between precision and recall is presented by β and the linear combination between the RMI loss and the Tversky loss is controlled by α.

7.2.7 Training and Post-Processing

Non-overlapping-patches of size 128 ×128 ×128 were extracted from the image volumes and used to train the network. 3D on-the-fly data augmentation was performed in a controlled manner in the training stage using the SimpleITK library. Linear translation, flipping, and permutation were performed on both the inputs and corresponding masks. Also, a spatially low variant intensity field was multiplied by the inputs to mimic the effects of radiofrequency coil modulation. In each iteration, three real patches and two augmented patches were fed as a batch into the network. Based on the learning rate range test, a learning rate of Lr = 0.005 was selected as an initial learning rate. Step decay was used to adjust the Adam optimizer's learning rate through the iterations. The learning rate started with Lr_{init}= 0.005 and was halved if, for 5 sequential epochs, the validation loss did not improve, or the number of epochs for the current learning rate surpassed 10. Besides, early stopping was considered if the validation loss did not improve for 50 sequential epochs. The training was performed with the Tensorflow on an NVidia Titan RTX GPU, 24GB RAM, and took approximately ~44 hours. Once the network was trained, it was tested based on full-size 3D high-resolution images, rather than 3D image patches, using a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz). As the 3D U-Net+DS+AG has 4 down-sampling stages, we padded the test input volume to the next size divisible by 16 before feeding to the network. In the post-processing stage, small floated shells with less than one percent of the largest volume are removed.

7.2.8 Separation of the Arteries from Veins

We focused on artery and vein separation in the calf region, which is more challenging than the thigh region due to smaller vessel size and more complex vessel geometry. A fast vessel enhancement function[227]was applied to the masked-volume to enhance the separation between closely co-localized vessels. Afterward, sequential series of the time-resolved first pass FE-MRA

images, which were spatially registered to the high-resolution FE-MRA, were used to initiate an arterial mask region in the high-resolution volume. This process was shown graphically in Figure 7-3.

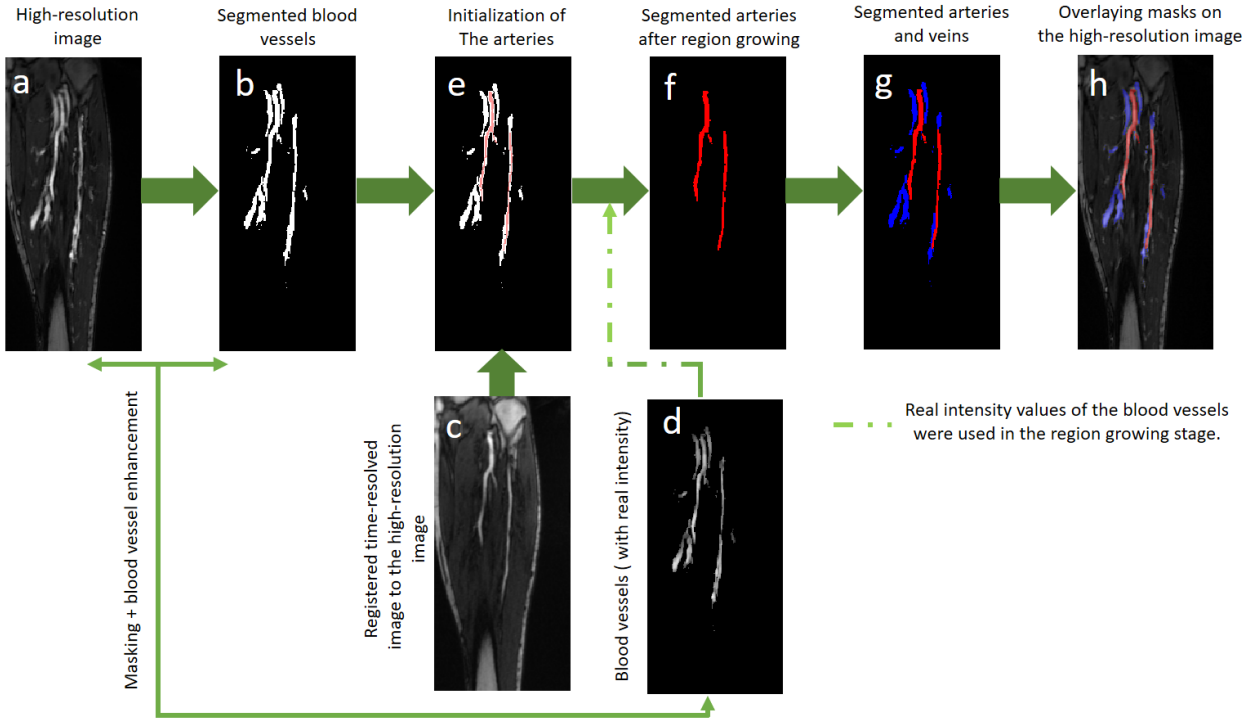

Figure 7-3. For the sake of simplicity, only a cross-section of the volume is visualized. To separate the arteries from the veins, the following steps were performed. First, the volumetric blood vessel binary mask (b) was extracted from the high-resolution image volume (a) using our blood vessel segmentation network. Also, at the same time, time-resolved image volume (c) was automatically registered to the high-resolution image. To obtain the only blood vessels with their real intensity (d), the high-resolution image was masked by the binary blood vessel mask, and a fast vessel enhancement algorithm42 was applied to enhance the obtained blood vessels. As noted in the main manuscript, the blood vessels' intensity values are required for the region growing algorithm. To obtain the initial arterial seeds, we first masked the time-resolved image by the blood vessel mask, and then adaptive binary thresholding was applied on the masked-region to detect the initial arterial seeds. A sample of the arterial seeds was shown in (e). Ultimately, the region growing algorithm was applied to the initial seeds to extract the arteries (f). Once the arteries were segmented, the remaining blood vessels were considered as the veins. A sample of the arterial and venous masks was shown in (g). Final overlaid masks on the high-resolution image were shown in (h).

A region-growing algorithm starting from the initial seeds obtained from the time-resolved FE-MRA data was applied to distinguish the arteries from the veins in the high-resolution FE-

MRA data. Pixel neighbors were added to the region if its intensity difference with the mean of the initial seeds was less than one standard deviation of the region's intensity. Two constraints were considered to stop the region-growing algorithm: 1) when the points reached an edge; 2) when the points reached to the maximum distance = 3.5 mm from the mass center of the seeds. The seed center was updated iteratively.

7.2.9 Evaluation

We performed two sequential grid searches to find the semi-optimized value for the balance factor (β), focusing factor (γ), and RMI weight ($1-\alpha$). In the first grid search, $\alpha=1$ was assumed, and the network was trained for the different combinations of β and γ. After finding the proper value for β and γ, we performed the second grid search to find a proper value for the RMI weight ($1-\alpha$). We used precision, F1, and recall to quantify the blood vessel segmentation performance. It is important to note that we calculated quantitative segmentation scores based on the network's volumetric results without post-processing.

We rationalized the architecture of the segmentation network by evaluating the role of the AG and DS mechanisms. We analyzed the learning curves, learned kernels and features from the first layer of the network, and pre-activation maps from the last layer of the network as well as segmentation results for the proposed network (3D U-Net+DS+AG) and the baseline method (3D U-Net). To evaluate the benefits of the AG, we also compared our 3D U-Net+DS+AG with a 3D U-net with DS but without AG (3D U-Net+DS), using similar metrics. To show the RMI loss's impact, we compared the 3D U-Net+DS+AG with and without the RMI loss qualitatively and quantitatively.

We compared the proposed segmentation network's performance with recent state-of-the-art networks, V-Net[213], DeepVesselNet-FCN[217,220] (with/without cross hair filters), and Uception[218].

It is important to mention that there are two versions of DeepVesselNet-FCN, one with cross-hair shape filters[220] and another one with 3D convolutional kernels[217]. For the remainder of this paper, DeepVesselNet-FCN refers to the network with cross-hair shape filters unless explicitly stated otherwise. We used the generalized dice loss function for training the V-Net and Uception. To train the DeepVesselNet-FCN (with and without cross-hair shape filters), class-balancing cross-entropy with proper weight was used. Since the initial results of DeepVesselNet-FCN (with and without cross-hair shape filters) were not satisfactory for our datasets, we slightly modified their structures to dense structures, in which we used concatenated-skip connections between first four layers of both networks.

We performed qualitative and quantitative analysis for the arteries and veins segmentation in the calf region. We evaluated qualitatively the volume-rendered images based on anatomical knowledge. For quantitative evaluation, we reported the F1 score for the segmented arteries and veins from the unseen test dataset.

For statistical analysis, two-tailed Student t-tests were used for pair-wise comparisons. A P-value less than 0.05 was considered statistically significant.

7.3 Result

7.3.1 Parameter Tuning

Table 7-1 reports the mean (±SD) of precision, recall, and F1 of the evaluation set for the first grid search. As β increased, recall increased, and precision decreased. As shown in Table 7-1, the F1 score was slightly higher for β=0.7 and γ=0.75 than other parameters.

Table 7-1. In the first grid search, parameter α, which controls the weight of the RMI loss, was considered as a fixed number (α=1), and the grid search was performed to find the proper value of β and γ. β controls the trade-off between the false negative (FN) and false positive (FP), and γ is the focusing factor in the focal loss. The F1 score was slightly higher for β=0.7 and γ=0.75 than other parameters.

	Precision			F1 (Dice)			Recall		
	β = 0.50	β = 0.60	β = 0.70	β = 0.50	β = 0.60	β = 0.70	β = 0.50	β = 0.60	β = 0.70
γ =0.25	0.82±0.05	0.76±0.06	0.70±0.03	0.79±0.04	0.77±0.04	0.76±0.05	0.73±0.06	0.77±0.05	0.82±0.07
γ =0.50	0.80±0.04	0.77±0.05	0.73±0.04	0.78±0.04	0.77±0.03	0.77±0.04	0.75±0.07	0.75±0.06	0.81±0.05
γ =0.75	0.82±0.05	0.78±0.07	0.75±0.03	0.78±0.04	0.78±0.03	0.80±0.02*	0.73±0.06	0.78±0.04	0.84±0.04
γ = 1	0.82±0.03	0.76±0.08	0.71±0.02	0.75±0.02	0.76±0.05	0.75±0.06	0.73±0.06	0.77±0.02	0.80±0.04

Table 7-2 reports the mean (±SD) of precision, recall, and F1 of the evaluation set for different α values and fixed γ and β values (β=0.7, γ=0.75) in the second grid search. For α=0.7, the F1 score is marginally higher than the others. Therefore, for the rest of the manuscript, β=0.7, γ=0.75, and α=0.7 were used as the objective function parameters.

7.3.2 Training Convergence

Figure 7-4 shows the learning curves for the 3D U-Net+DS+AG and baseline network (3D U-Net). The loss value on the level i=0 of the 3D U-Net+DS+AG over the epochs was compared against the loss value of the 3D U-Net. It is worth noting that for training the 3D U-Net, only level i=0 component of the total loss function described in Equation (7-7) was used as the loss function.

Table 7-2. Second grid search for the hype-parameter tuning: α controls the weight of the RMI loss. The F1 score was slightly higher for α =0.7 than others. It is important to emphasize that reported values are based on the fixed β and γ values (β=0.7 and γ=0.75).

	Precision	F1 (Dice)	Recall
α = 1	0.7509±0.0311	0.7976±0.0201	0.8448±0.0402
α = 0.9	0. 7621±0.0308	0.8026±0.0198	0.8402±0.0397
α = 0.8	0. 7645±0.0307	0.8084±0.0198	0.8405±0.0400
α = 0.7	0. 7658±0.0296	0.8087±0.0208*	0.8410±0.0407
α = 0.6	0. 7661±0.0308	0.8041±0.0199	0.8395±0.0398
α = 0.5	0. 7659±0.0313	0.8015±0.0201	0.8387±0.0405
α = 0.4	0. 7652±0.0310	0.7984±0.0206	0.8362±0.0410
α = 0.3	0. 7648±0.0318	0.7920±0.0316	0.8332±0.0414
α = 0.2	0. 7655±0.0293	0.7894±0.0275	0.8280±0.0374
α = 0.1	0. 7652±0.0285	0.7856±0.0190	0.8275±0.0423
α = 0	0. 7645±0.0300	0.7844±0.0226	0.8258±0.0419

The validation loss for 3D U-Net and the 3D U-Net+DS+AG decreased as the training loss drops. In the early epochs, i.e., epochs 1-20, 3D U-Net had a slightly faster loss reduction rate. However, in the middle stage of learning, i.e., epochs 20-70, the validation loss for 3D U-Net+DS+AG had a higher reduction rate than 3D U-Net.

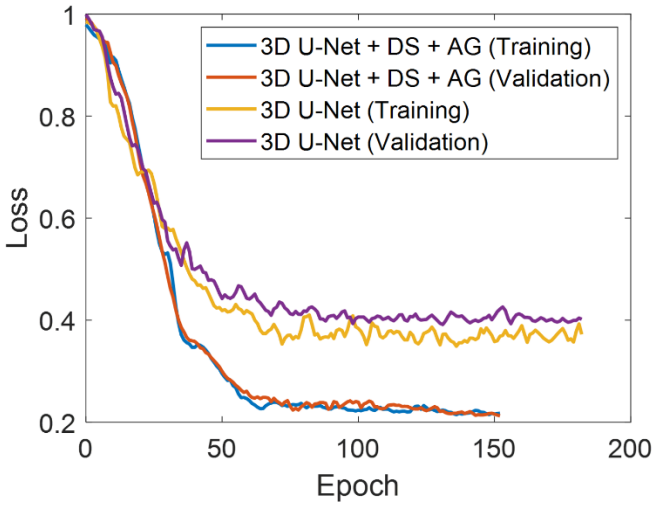

Figure 7-4. Learning curve comparison between 3D U-Net as a baseline model and 3D U-Net with deep supervision and local attention gates (3D U-Net+DS+AG) as our proposed method. 3D U-Net+DS+AG has a higher rate of loss reduction and faster convergence than 3D U-Net.

Compared with 3D U-Net, 3D U-Net+DS+AG had less oscillation in the learning curves and faster convergence. The 3D U-Net+DS+AG ultimately achieved lower training and validation losses than the 3D U-Net.

7.3.3 Learned Kernels and Intermediate Features

Figure 7-5 shows all learned kernels (upper left panels), cross correlation matrix between the learned kernels (upper right panels), and extracted features (lower panels) from a representative input volume for the first convolutional layer of the 3D U-Net (A) and 3D U-Net+DS+AG (B). It is evident that both kernel maps have organized patterns. However, the diversity of the patterns for the 3D U-Net+DS+AG was greater than 3D U-Net. As evident in Figure 7-5, there is more redundancy in the learned kernels and extracted features for the 3D U-Net than the 3D U-Net+DS+AG. The sum of the absolute value of the cross-correlation matrix for the 3D U-Net was 68.506 (out of 256) and for the 3D U-Net+DS+AG was 42.500(out of 256). This indicates that the learned kernels for the 3D U-Net had more similarities than the 3D U-Net+DS+AG.

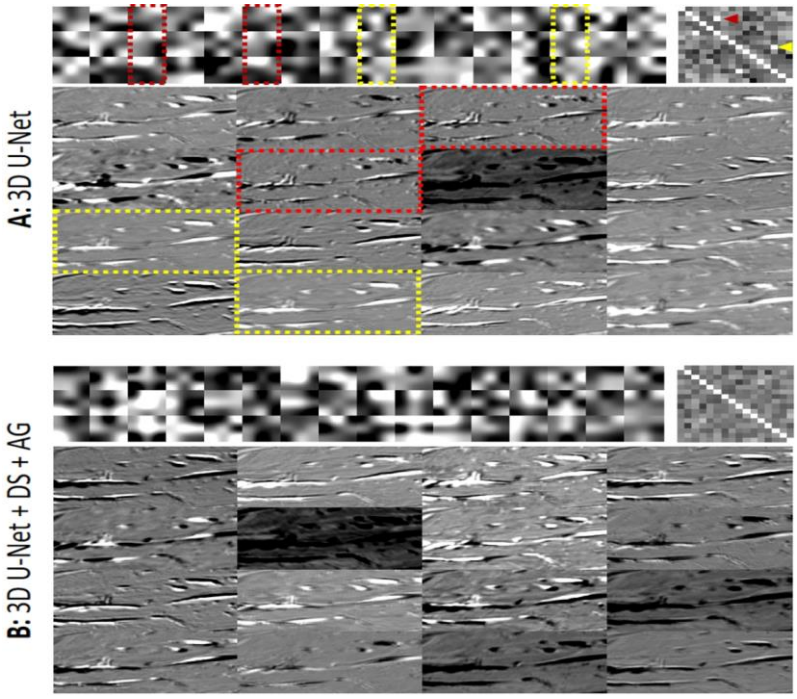

Figure 7-5. Learned kernels, cross correlation matrix between the learned kernels, and intermediate feature visualization for the first convolutional layer of the baseline 3D U-Net model (A) and the proposed 3D U-Net+DS+AG method (B). Learned kernels, cross correlation matrix between the learned kernels, and a slice of the extracted 16 features are shown in the upper-left, upper-right and lower panels of each method, respectively. Similar learned kernels and their corresponding extracted features are shown inside the dashed-red and dashed-yellow rectangles. Samples of the extracted features show that the diversity of the features extracted from 3D U-Net+DS+AG is higher than 3D U-Net, which is expected to translate to higher discriminatory capability. The red and yellow arrowheads (panel A, top right) show high cross correlation coefficients representing similarity in the learned 3D U-Net kernels; whereas 3D U-Net+DS+AG did not have these high cross correlation values due to its greater diversity.

7.3.4 Impacts of Local Attention Module on the Training Process

Figure 7-6 shows the learning curves and pre-activation probability maps from the last layer of the 3D U-Net+DS and 3D U-Net+DS+AG. The 3D U-Net+DS+AG has a faster convergence rate than the 3D U-Net+DS. Based on the small gap between the training loss and the validation loss, the 3D U-Net+DS and the 3D U-Net+DS+AG were able to generalize from the training data to the unseen test data. The final loss values of the 3D U-Net+DS+AG network was lower than the 3D U-Net+DS. Moreover, it had a shorter training time in comparison to the 3D U-Net+DS. As pointed by a blue arrow, probability map of the 3D U-Net+DS is more diffused than the 3D U-Net+DS+AG.

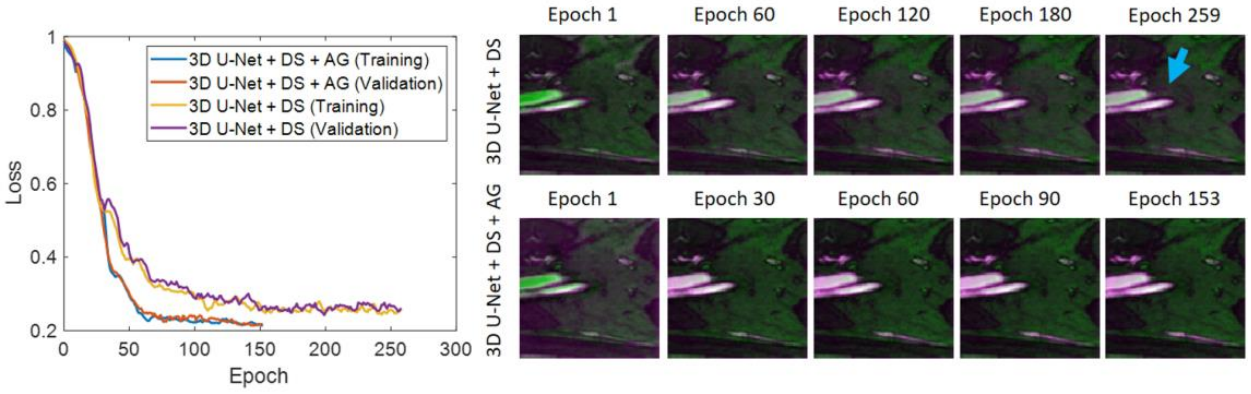

Figure 7-6. A comparison of the 3D U-Net+DS+AG with the 3D U-Net+DS. The training and validation loss is plotted for both methods on the left side. Two representative pre-activation probability maps are shown on the right side. As pointed by a blue arrow in the pre-activation probability maps, using the attention module results in more focused probability maps.

7.3.5 Impact of RMI loss on the Segmentation Results

Figure 7-7 shows qualitative blood vessel segmentation results for the proposed network with (1-α=0.3) and without (1-α=0) the RMI loss. It is evident that the network with RMI loss preserves the blood vessel connectivity better than the network without RMI loss. Quantitative scores reported in Table 7-2 show that the proposed network with RMI loss increased the precision score of the segmentation results from 0.7509 (without RMI loss) to 0.7658. There was also an incremental improvement in the F1 score where the F1 score was increased from 0.7976 to 0.8087.

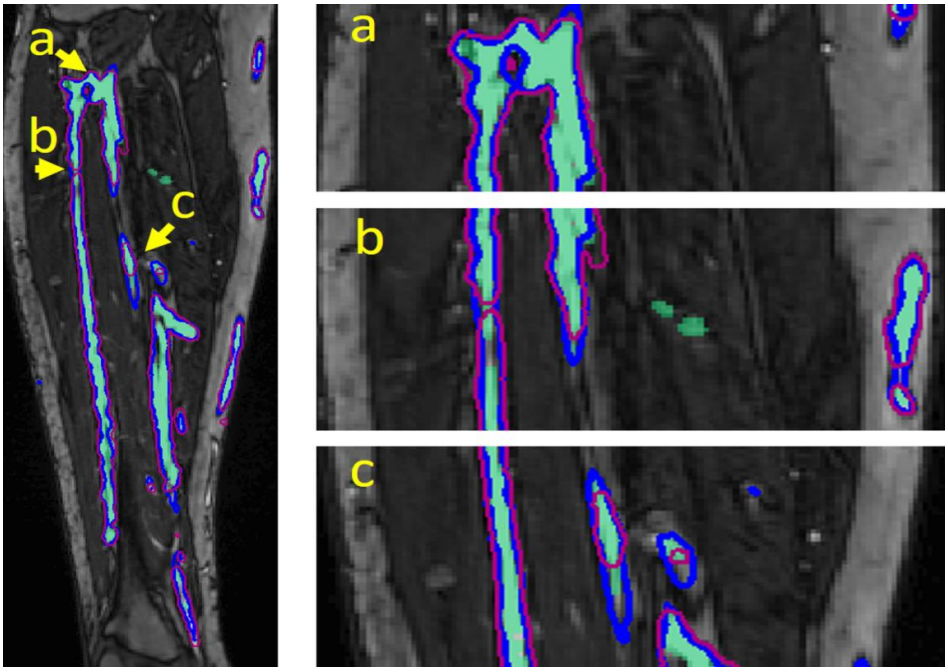

Figure 7-7. Effect of the Region Mutual Information (RMI) loss on the segmentation results. This figure shows a representative coronal slice of a patient segmented by the 3D U-Net + DS + AG with and without the RMI loss. Zoomed in regions are shown in (a,b,c) on the right. The obtained segmentation results with RMI loss and without RMI loss are contoured with blue and red color, respectively. The ground truth region is filled with light-green color. Including RMI loss in the 3D U-Net + DS + AG training stage leads to better preservation of the blood vessel connectivity compared to 3D U-Net + DS + AG without the RMI loss.

7.3.6 Segmentation Results

Figure 7-8 shows representative segmentation results on a test dataset. We contoured the segmentation results for the 3D U-Net (a), 3D U-Net+DS(b), and 3D U-Net+DS+AG(c), with the ground truth segmentation shown in light-green. Figure 7-8 (d) shows a comparison between the segmentation result obtained from 3D U-Net+DS+AG, shown as the region filled with pale red, and the result from 3D U-Net (light-gray contour) and 3D U-Net+DS (yellow contour).

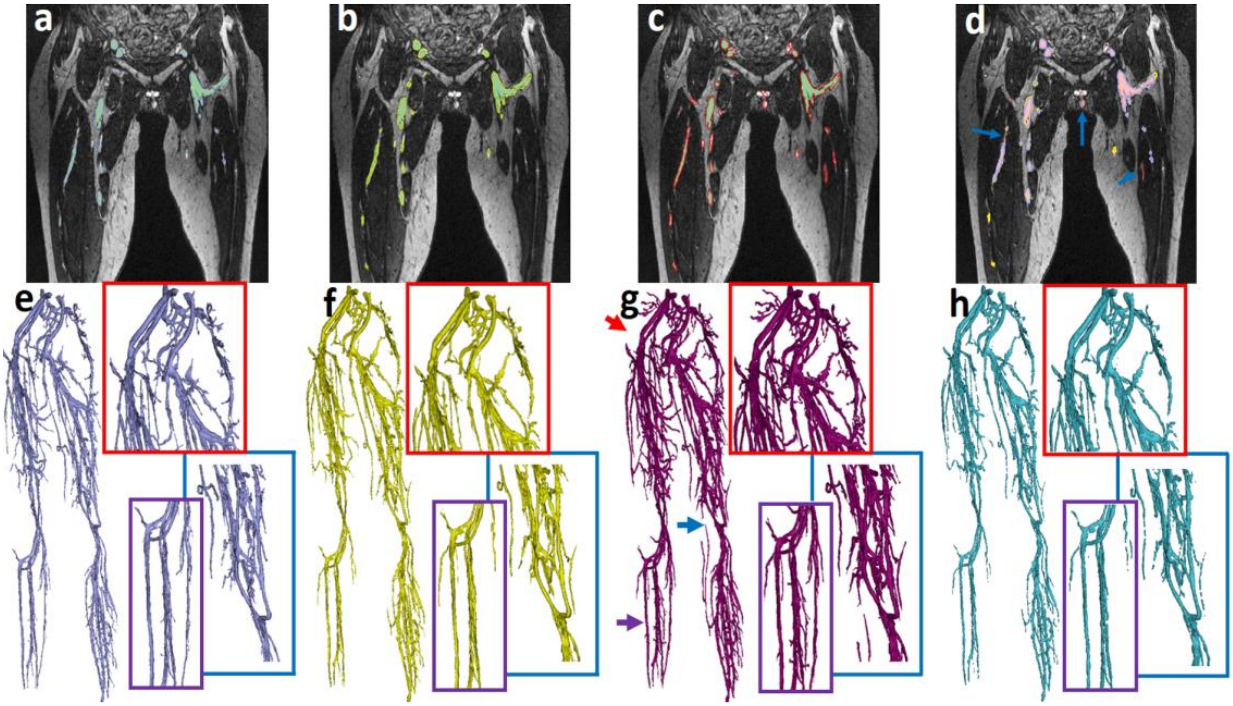

Figure 7-8. Representative segmentation results and qualitative comparisons. (a) Results from 3D U-Net, (b) 3D U-Net+DS, (c) 3D U-Net+DS+AG are visualized with gray, yellow, and red contours, respectively. Ground truth is shown with green filled region. (a-c) show the comparison of the networks with ground truth, and (d) shows the comparison of the 3D U-Net (gray contour) and 3D U-Net+DS (yellow contour) with 3D U-Net+DS+AG (filled with pale red). (e-h) show volume-rendered images for the 3D U-Net, 3D U-Net+DS, 3D U-Net+DS+AG, and ground truth (obtained by two expert radiologists) with their respective colors used in (a-d). The proposed method captures blood vessels that were not captured by other methods (blue and red arrows in (g)). Besides, segmented blood vessels in the left calf using 3D U-Net+DS+AG has a higher density than 3D U-Net and 3D U-Net+DS (purple arrow in (g)). An expert radiologist confirmed that these extra-segmented vessel branches (blue, red, and purple arrows in (g)) are blood vessels that were initially missed by the radiologists in the manual segmentation.

As highlighted by the blue marker in Figure 7-8 (d), thin blood vessel branches were captured by our-proposed method, but missed by both 3D U-Net and 3D U-Net+DS. Figure 7-8 (e-h) displays the volume-rendered image for 3D U-Net (e), 3D U-Net+DS(f), 3D U-Net+DS+AG (g), and ground truth (h). Areas pointed by the colored arrows in Figure 7-8 (g) are magnified in each volume rendered images. As can be seen in blue rectangles Figure 7-8 (e-h), our proposed network is captured more blood vessels than the two other methods compared. The segmented blood vessels by our proposed method (Fig. 7.8 (g)) contains more structures than the initial ground truth blood vessels (Fig. 7.8 (h)). An expert radiologist subsequently evaluated the blood vessel masks and confirmed that the mentioned extra structures segmented were indeed blood vessels.

Table 7-3. Quantitative comparisons. 3D U-Net+DS+AG achieved higher F1 and recall scores than other methods. Also, 3D U-Net+DS+AG achieved a higher precision score than other methods except for Volumetric Net and Uception. There was no statistically significant difference ($P>0.05$) between the precision score of our proposed method (3D U-Net+DS+AG) and the precision scores of state-of-the-art networks (Volumetric-Net, DeepVesselNet-FCN, and Uception). 3D U-Net+DS+AG achieved a statistically significant higher precision than 3D U-Net. For the F1 and Recall scores, our proposed method (3D U-Net+DS+AG) achieved a statistically higher score ($P<0.05$) than other methods.

	Precision	F1	Recall
3D U-Net[25]	0.6326±0.0437*	0.6502±0.0428*	0.6878±0.0700*
3D U-Net+DS	0.7321±0.0519	0.7603±0.0287*	0.7807±0.0510*
3D U-Net+DS+AG	0.7658±0.0296	0.8087±0.0208	0.8410±0.0407
DeepVesselNet-FCN (with CHS# filters)[35]	0.7501±0.0665	0.7573±0.0408*	0.7570±0.0876*
DeepVesselNet-FCN (without CHS# filters)[32]	0.7514±0.0480	0.7481±0.0405*	0.7546±0.0703*
Volumetric-Net[28]	0.7678±0.0760	0.7604±0.0402*	0.7791±0.0602*
Uception[33]	0.7690±0.0564	0.7651±0.0381*	0.7774±0.0560*

#: CHS stands for the cross hair shape.

*: statistically significant difference when compared with 3D U-Net+DS+AG.

Table 7-3 reports the Precision, F1, and recall scores for the 3D U-Net, 3D U-Net+DS, and 3D U-Net+DS+AG for the evaluation set. When compared with 3D U-Net, the 3D U-Net+DS and 3D U-Net+DS+AG had a less dispersed F1 score (lower standard deviation) for the unseen data. The 3D U-Net+DS+AG achieved statistically superior quantitative F1 and Recall scores than the 3D U-Net and 3D U-Net+DS methods ($P<0.05$).

7.3.7 Comparison with state-of-the-art Networks

Figure 7-9(a-d) shows the qualitative results in the axial plane for (a) DeepVesselNet-FCN (gray contour), (b) V-net (yellow contour), (c) Uception (blue contour) and (d) 3D U-Net+DS+AG (red contour).

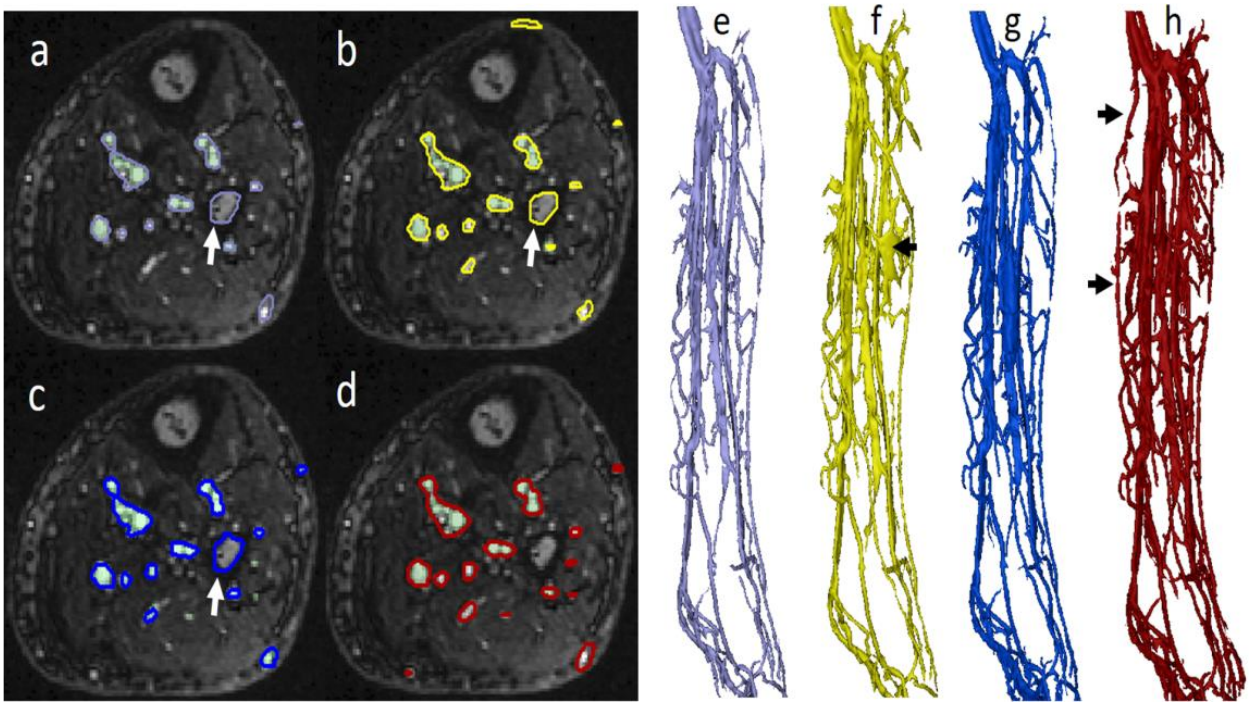

Figure 7-9. Qualitative comparisons of our network (3D U-Net+DS+AG) with state-of-the-art networks DeepVesselNet-FCN, Volumetric-Net (V-Net), and Uception in blood vessel segmentation. (a-d) show the results obtained by DeepVesselNet-FCN (a; gray contour), V-Net (b; yellow contour), Uception (c; blue contour), and 3D U-Net+DS+AG (d; red contour). Ground truth regions in (a-d) are filled by green color. As pointed out by white arrows in (a-c), DeepVesselNet-FCN, Uception, and V-Net incorrectly segment the bone as a blood vessel; whereas this mistake was avoided by 3D U-Net+DS+AG (d). (e-h) represent the volume rendered images for the DeepVesselNet-FCN (e), V-Net (f), 3D Uception (g), and 3D

U-Net+DS+AG (h). As pointed out by black arrows in (h), our proposed method segmented out a branch of the blood vessel that was missed by other segmentation networks. The black arrow in (f) shows a portion of the segmented blood vessel with extravascular soft tissue contamination.

Since the DeepVesselNet-FCN (without cross-hair shape filters) had a similar qualitative performance to the DeepVesselNet-FCN (with cross-hair shape filters), we only showed the qualitative results for the DeepVesselNet-FCN (with cross-hair shape filters). The ground truth region is filled with green color in Figure 7-9 (a-d). While part of the bone was incorrectly segmented as blood vessels (white arrows) by the DeepVesselNet-FCN, V-net, and Uception, the proposed network correctly omitted it. Figure 7-9 (e-h) shows the volume-rendered images in the calf region for (e) DeepVesselNet-FCN, (f) V-net, (g) Uception, and (h) 3D U-Net+DS+AG. In Figure 7-9 (h), black arrows show regions where our proposed method captured more blood vessels than the DeepVesselNet-FCN, V-net, and Uception. The black arrow in Figure 7-9 (f) highlights the region segmented by V-net mixed with skin. Based on the quantitative scores reported in Table 7-3, our method had statistically higher F1 and recall scores when compared with V-Net, Uception, and DeepVesselNet-FCN (with/without cross-hair shape filters) ($P<0.05$). There was no statistically significant difference between the precision score of the 3D U-Net+DS+AG, V-Net, Uception, and DeepVesselNet-FCN (with/without cross-hair shape filters) ($P>0.05$).

7.3.8 Separation Results

Figure 7-10(a) shows the segmented artery by the proposed method based on a selected coronal image of a high-resolution FE-MRA. The three axial sections of the right calf are shown in Figure 7-10(b). The popliteal artery, with two veins in proximity to it, is correctly segmented by the proposed method (Fig. 7-10(b)). Figure 7-10(c) shows MIP from the scanner console (top view) and the MIP of the extracted artery from the high-resolution image via our pipeline (bottom view). Figure 7-10 (d) shows a 3D rendered artery of the same patient's calves.

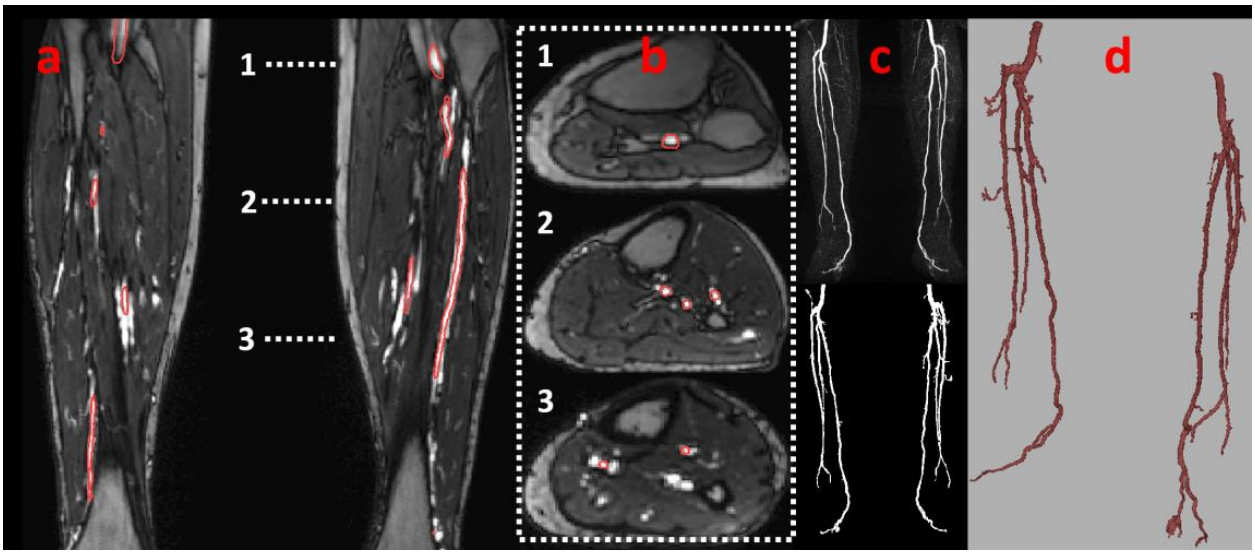

Figure 7-10. Arterial tree extraction for a representative case: Arteries in from a coronal view (a) of high-resolution FE-MRA and three axial views (b) for the right calf are shown. (c) represents the maximum intensity projection (MIP) of the data obtained by the scanner (c; top panel) and the extracted-arterial tree based on our method (c; lower panel). (d) represents the volume-rendered arterial tree extracted by our proposed algorithm. As shown in (c), the MIP image based on the extracted arterial tree from our algorithm is in good agreement with the arterial MIP image generated by the scanner.

Figure 7-11(a,b) shows a qualitative comparison of the artery segmentation for a patient using the proposed method (green-contour) vs. manual annotation by a radiologist (red-contour and filled with pale red). Figures 7-11 (c and d) show the volume-rendered image obtained by radiologists and our proposed method, respectively, and they confirm that the segmented artery using our method is similar to the ground truth concerning the structure and direction of the branches. Videos S-1 and S-2 (available online as a supporting file of our published article[12]) provide the movie of this case with full segmentation of the arteries and veins and comparison between our proposed method and radiologist annotation for the artery.

Quantitatively for the seven test cases, the proposed method achieved higher mean F1(±SD) for the artery and vein segmentation than the classic Fuzzy-based approach [9](0.8274 ±0.0152

vs. 0.7321±0.0921 for the artery segmentation, and 0.7863±0.0643 vs. 0.7405±0.1061 for the vein segmentation; see Appendix VIII).Using a general-purpose desktop computer (Intel Core i7-8700 CPU, 3.10 GHz), the entire peripheral artery/vein segmentation pipeline took less than 4 min (Data loading and registration ≈ 41.1 seconds, deep learning-based blood vessel segmentation ≈ 28.4 seconds, filtering small shells + fast vessel enhancement ≈ 72.2 seconds, and the region growing ≈ 85.3 seconds) to generate the full set of results for a typical matrix size of 400 × 1200 × 212.

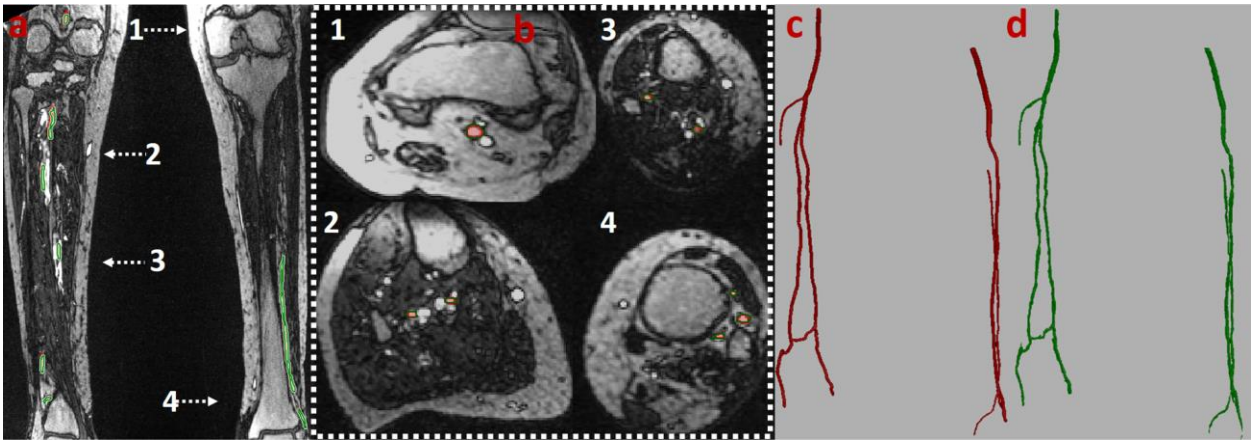

Figure 7-11. Arterial tree extraction for a case with peripheral arterial disease. Arteries from the coronal view (a) of the high-resolution FE-MRA and four axial views (b) for both calves are shown. Arteries segmented by our proposed method are represented with the green color, and arteries annotated by an expert radiologist are represented by red color. (c) shows the volume-rendered image obtained by an expert radiologist, and (d) shows the extracted arterial tree by our algorithm. Visually, the extracted arterial tree using our algorithm is similar to that defined by expert annotation.

7.4 Discussion

In this work, we proposed an automatic pipeline to segment peripheral vasculature from high-resolution FE-MRA datasets and labeled them as arteries and veins based on region growth initialization using time-resolved MRA images. For vessel segmentation, we implemented a 3D U-Net structure with a pyramid of inputs that incorporated local AGs and DS mechanism to take

advantage of multilevel information, facilitate the segmentation of the vessels in low tissue contrast regions, overcome vanishing gradient issues, and expand the discriminative capability of the network. While V-Net, DeepVesselNet-FCN, and Uception, gained 0.7604, 0.7573, and 0.7651 mean dice scores on the evaluation sets, respectively, our method achieved a mean dice score of 0.8087 ($P<0.05$). Finally, arteries were successfully labeled in the calf and assessed based on the anatomical knowledge and MIP image obtained from the scanner. Quantitatively, the proposed platform segmented the arteries and veins in the calf region with a mean dice score of 0.8274 and 0.7863, respectively.

We evaluated the effectiveness of the local AGs and the DS mechanism in the blood vessel segmentation. Using DS forced the network to learn more discriminative features and facilitated updating the learnable parameters in the AGs. Adding a pyramid of inputs in multiple scales helped the network to minimize the risk of missing thin blood vessels. Besides, the designed objective function, based on the combination of the focal loss and RMI loss, may help the network cope with data imbalance and preserve the structural similarity between the segmented vessels and the ground truth.

As presented in Figure 7-6, the probability map shows diffused probability in the 3D U-Net+DS image beyond the boundary of the blood vessels. For this case, the haze in the probability map is not sufficiently strong to result in a difference in blood vessel segmentation between the 3D U-Net+DS vs. 3D U-Net+DS+AG, mainly because there is still strong contrast between the blood vessel and surrounding environment. However, the contrast between blood vessels and some parts of the tissues, e.g., those that are modulated with coil intensity or close to the fat component of the bones, is low, and an attention mechanism could potentially be helpful in extracting the vessels from these regions.

We demonstrated our network's ability to segment thin vessels in some regions of the lower extremities that were not initially identified by human experts. Based on the volume-rendered images presented in Figure 7-8(g), the segmented vessels by our proposed method labeled additional structures as blood vessels that were not labeled in the initial manual segmentation (Fig. 7.8(h)). An expert radiologist subsequently evaluated the blood vessel masks and confirmed that the aforementioned extra structures were blood vessels. This finding highlights the potential advantages of well-validated machine learning-based segmentation methods compared to manual segmentation – it offers the possibility to reduce human errors in tedious and labor-intensive tasks or in applications where subtle findings may elude the human visual perception.

Although the inclusion of the RMI loss in the proposed network's training stage increases the mean precision score from 0.7509 to 0.7658 (Tab. 7.2) for the evaluation sets, it is still relatively lower than the recall score. Two potential reasons may explain the lower precision score compared to the recall score in the proposed blood vessel segmentation network. First, we weighed the recall more than precision in our proposed network's loss function. Based on Table 7-1, higher β results in higher recall scores and lower precision scores. Second, the manual ground truth was not perfect and missed some smaller blood vessels.

We used additional dynamic MRA datasets in the artery/vein separation step of our pipeline to initialize arterial seeds for region growth in the high-resolution steady-state images. In contrast to the first pass gadolinium-enhanced MRA, there is an appreciable signal intensity difference between arteries and veins; there is no observable contrast between arteries and veins in our steady-state FE-MRA. Therefore, if dynamic MRA datasets were not to be used, several other potential approaches could be employed for artery-vein separation for our FE-MRA: 1) Identify structural differences between the arterial tree and venous structure and separate them based on the

geometrical information. Such an approach could be susceptible to failure due to the wide range of geometrical variations across a large patient population. 2) Manipulate the MRI acquisition signal in the high-resolution FE-MRA scan and sensitize the signal to inherent characteristics of the veins and arteries, such as blood flow direction or oxygen level[228-230]. It is possible to combine these approaches with our segmentation network to achieve artery/vein separation without the need for time-resolved images.

When used clinically, FE-MRA quality could sometimes, although rarely, be degraded by motion artifacts. Patients requiring diagnostic FE-MRA might feel pain in their lower extremities due to their underlying clinical condition. This could result in the patient's inability to hold still for an extended time and, consequently, motion artifacts in the high-resolution images. In this scenario, a repeat scan is currently required to obtain motion artifact-free images. We view this challenge as a potential opportunity for our segmentation method to extract the vasculature from artifact-degraded data as a future direction of our work. We aim to add a rigid motion artifact to the training input and train the network to extract the vessels from motion-contaminated data in future feasibility studies, with eventual testing on real-life motion-contaminated data.

At our institution, the clinical workflow for these vessel segmentation tasks typically requires substantial manual input and hours of time for each case. The proposed platform substantially reduced the data processing time from several hours to less than 4 minutes. Due to the subjective nature of manual labeling and its dependency on the physician's experience and knowledge, the proposed platform also has the potential to reduce inter-observer variability.

While the proposed technique used FE-MRA images of peripheral lower extremities, we speculate the segmentation and discrimination approach could be applied to any peripheral lower

extremity MRA images acquired at steady-state using a similar intravascular (blood pool) contrast agent. Readers interested in the off-label diagnostic ferumoxytol use in MRI are referred to several excellent published review papers[231,232]. With 6.5 million Americans over the age of 40 having PAD, we expect this work to have high clinical relevance.

7.5 Conclusion

In conclusion, by taking advantage of the time-resolved images and 3D convolutional neural network, a blood vessel segmentation and artery/vein separation platform was successfully implemented and evaluated using clinically obtained FE-MRA images. The proposed method for segmentation of peripheral arteries and veins from lower extremity FE-MRA images achieved its task in less than 4 minutes, whereas several hours may be needed by an expert radiologist accomplishing the same task.

Chapter 8 Conclusion

In this dissertation, we presented several deep neural network-based applications in CMRI. More specifically, one method was discussed in Chapter 3 to accelerate the cardiac cine imaging, and two methods were discussed in Chapters 4 and 5 to reduce the respiratory motion artifacts in 2D and 3D cardiac cine imaging. Chapter 6 discussed one method to speed up the T1/T2 computation in cardiac imaging, and finally, in Chapter 7, we discussed one method to segment the peripheral arteries and veins from the MRA images in the lower extremities. Although we designed these methods with application specificity and clinical utility in mind, they are applicable to many other applications that are similar to ours. We tried to implement these particular applications and improve their performance and build ideology for addressing the potential limitations that may occur in future applications. This chapter briefly summarizes the technical developments of the applications mentioned in this dissertation and then describes the potential directions for future works.

8.1 Summary of Technical Development

8.1.1 Deep learning based Dynamic Cardiac Magnetic Resonance Image Reconstruction Pipeline

In Chapter 3, we implemented a neural network-based 2D cardiac cine MR reconstruction pipeline to speed up the imaging process. The designed neural network used the redundant temporal dimension information to learn the more effective Spatio-temporal regularizer. Also, data consistency was included in the reconstruction platform via the hard replacement scheme. Instead of focusing on specific imaging orientations, short-axis (SA), or horizontal long axis (HLA), we included the complete anatomical cardiac exams dataset in the training stage to increase the

diversity of the dataset. Moreover, we prepared the data in a way to be consistent with the actual scanner's outputs. We also introduced a sampling strategy that can effectively cover the k-space through the cardiac frames and minimize the eddy current-related artifacts. The reconstruction platform achieved the higher acceleration factors, e.g., 8X-10X, with minimal loss of the cardiac structures. Compared to the previously developed reconstruction pipeline [7], this pipeline includes the temporal dimension and faster data consistency module and the efficient undersampling strategy, which enables achieving the higher acceleration factors.

8.1.2 Retrospective Respiratory Motion Correction in Cardiac Cine MRI Reconstruction

In chapter 4, we implemented a deep neural network-based platform to compensate for the respiratory motion in the free-breathing cardiac cine MRI. Conventional deep neural network-based methods usually require paired data to be trained. However, accessing the paired data in cardiac cine MRI, i.e., one without breathing artifact and another with the breathing artifact, is almost impossible. In this work, we used the potential of the autoencoder architecture to implement a neural network that does not require paired data for the training stage. We first tested the neural network on the simulated dataset, analyzed the motion compensation accuracy, and reported the quantitative metrics. After confirming the motion compensation accuracy, we trained and tested the neural network on the real datasets. We showed that the proposed approach could reduce the respiratory motion artifact and achieve a quality that does not significantly differ from the breath-hold acquisition. The proposed approach potentially can remove the breath-hold assumption in the clinical cardiac cine MRI, thus increasing patient comfort; besides, since it does not require breath-hold, it can also decrease the duration of the clinical exam.

8.1.3 Temporally Aware Volumetric GAN-based 4DMR Image Reconstruction and Respiratory Motion Compensation

In chapter 5, we implemented a neural network-based approach to compensate for the respiratory motion and reconstruct the highly undersampled dataset in 3D dynamic cine cardiac MRI. We incorporated a novel temporally aware objective function as an extra regularizer and adversarial loss, L1 and SSIM loss functions to reduce flickering artifacts through the cardiac phases with no explicit need to use the multiple cardiac phases as the inputs for the network. Besides, we addressed the well-known challenges of training GANs for high-dimensional images by adopting an effective progressive training strategy based on starting the training from the low-resolution volumetric images and gradually increasing the resolution to reach the original volumetric image size. We compared the network's performance in the relatively large cohort of CHD patients against the existing methods. It is worth mentioning that the designed neural network can reconstruct at least 2X more undersampled 3D dynamic cine cardiac data with significantly higher image quality and lower artifact compared to the conventional SG WV method.

8.1.4 Fast and accurate quantification of myocardial T1 and T2 values using Deep learning Bloch Equation Simulations (DeepBLESS)

In this chapter, we implemented a neural network to accelerate the myocardial T1/T2 values calculation. Although BLESSPC can generate accurate T1/T2 maps for both conventional widely-used Cartesian-based sequences and radial sequences, it is slow. We replaced this intensive computation with a neural network to speed up the T1/T2 maps generation. We used the simulated datasets to train the neural network and validated and tested the performance on the real acquired datasets. We showed that, for the radial T1-T2 sequence, the proposed method could achieve 18,000 times acceleration while achieving similar accuracy and precision compared to BLESSPC.

8.1.5 Automatic Peripheral Artery and Vein Segmentation

In this chapter, we implemented a pipeline to automate the segmentation of the peripheral arteries and veins in ferumoxytol-enhanced MR angiography. We first divide the segmentation of the arteries and veins into two parts. In the first part, we implemented a convolutional neural network to segment the whole vasculature from the lower extremities. In the second part, we separate the arteries from the veins using a conventional region growing algorithm and take advantage of the extra information, i.e., time-resolved images. In the blood vessel segmentation task, we performed relatively extensive evaluations to understand better the different modules' roles, such as the attention gates, deep supervision, etc. The proposed method for segmentation of peripheral arteries and veins from lower extremity FE-MRA images can complete its task in less than 4 minutes, whereas several hours may be needed by an expert radiologist accomplishing the same task.

8.2 Future outlook

8.2.1 Deep learning based Dynamic Cardiac Magnetic Resonance Image Reconstruction Pipeline

In this work, we implemented and tested the reconstruction pipeline in a retrospective manner. In order to use this application in practice, we need to implement it on the scanner, which requires changing the undersampling pattern of the dynamic cardiac cine imaging pulse sequence and replacing the reconstruction part of the scanner with the trained network. Such translation, although it seems trivial from the technical perspective, can pave the way for the radiologists and MR technicians to assess the quality of the reconstructed images, and possibly such assessments might provide proper feedbacks to us to know more about the performance of the neural network in the practical scenarios.

8.2.2 Retrospective Respiratory Motion Correction in Cardiac Cine MRI Reconstruction

In this work, we mainly focused on finding a way to remove the assumption of accessing the paired data for respiratory motion correction in cardiac cine MRI. One alternative way that we could consider for future studies is to simulate the realistic respiratory motion and its induced artifact in cardiac cine MRI. Such an approach will enable us to train the network on a large dataset in a supervised manner and test the realistically respiratory motion-contaminated images. Another direction that might be relevant to follow is taking advantage of more sources of information such as channel and temporal dimensions and the external motion sensor such as a belt or internal motion surrogates.

8.2.3 Temporally Aware Volumetric GAN-based 4DMR Image Reconstruction and Respiratory Motion Compensation

In this work, we did not include the multi-channel information in our network, mainly because of the large dimensionality challenges in training the network. Probably a practical approach for including the multi-channel information in the high dimensional network is to divide the problem into several subproblems, which can be handled with the available GPUs. Another possibility for future studies is to test the proposed method and assess a large cohort of patients under free-breathing conditions without anesthesia. Although we trained the proposed method on CHD patients who underwent cardiac MRI under anesthesia, it showed promises for a patient with breathing irregularity and a patient during spontaneous free-breathing without anesthesia. So, thorough evaluations are warranted before it can be applied to adult patients during free-breathing.

8.2.4 Fast and accurate quantification of myocardial T1 and T2 values using Deep learning Bloch Equation Simulations (DeepBLESS)

In this work, the implemented DeepBLESS method, although it achieved promising results, might be susceptible to motion, particularly in patients who could not hold their breath appropriately. Because the developed network uses the series of pixel's values to calculate the T1/T2 values, in the presence of the motion, it can result in inaccurate T1/T2 calculations. For future studies, working on the registration methods could potentially solve this issue.

8.2.5 Automatic Peripheral Artery and Vein Segmentation

In this work, we used an additional dynamic MRA dataset in our pipeline's artery/vein separation step to initialize arterial seeds for region growing algorithm in the high-resolution steady-state images. To remove the assumption of the availability of the dynamic MRA dataset, we proposed several other potential directions for the artery-vein separation stage: 1) Identify structural differences between the arterial tree and venous structure and separate them based on the geometrical information. 2) Manipulate the MRI acquisition signal in the high-resolution FE-MRA scan and sensitize the signal to inherent characteristics of the veins and arteries, such as blood flow direction or oxygen level.

APPENDIX I SSIM

In general, SSIM quality metrics is comprised of the multiplication of the three terms, including the luminance term $L(.)$, contrast term $C(.)$, and structural term $S(.)$. SSIM per pixel/voxel between two 2D/3D images A and B can be formulated as Equation (A-1)[233].

$$SSIM(x,y) = [L(x,y)]^{\alpha}[C(x,y)]^{\beta}[S(x,y)]^{\gamma} \quad \text{(A-1)}$$

Where A and B are inputs to all functions, but they are omitted for the sake of clarity .The x and y are the pixel/voxel intensity values from the input images.

Luminance, contrast, and structural terms can be defined as Equations (A-2) to (A-4):

$$L(x,y) = \frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1} \quad \text{(A-2)}$$

$$C(x,y) = \frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2} \quad \text{(A-3)}$$

$$S(x,y) = \frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3} \quad \text{(A-4)}$$

Where μ_x, μ_y, σ_x, σ_y, and σ_{xy} are the local means, standard deviations, and cross-covariance. By considering $\alpha = \beta = \gamma = 1$, and $C_3 = C_2/2$, which has been proposed by Wang et al.[233], the original SSIM quality metric (Eq. A-1) can be simplified to Equation (A-5).

$$SSIM(x,y) = \frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1} \times \frac{2\sigma_{xy}+C_2}{\sigma_x^2+\sigma_y^2+C_2} \quad \text{(A-5)}$$

Where C_1 and C_2 are two small constants to stabilize the division with a weak denominator, local statistics are computed by applying the 2D/3D Gaussian filter with standard deviation σ_G.

Finally, the mean of the calculated SSIM map, a scalar value, can be used as a similarity metric between two given 2D/3D images. Equation 6 shows the mean of the SSIM map, which commonly used as an image quality assessment metric or loss function in the neural networks:

$$MSSIM(A,B) = \frac{1}{N}\sum_x \sum_y SSIM(x,y) \quad \text{(A-6)}$$

N is the number of the pixels/voxels inside the input images.

In our work, we used SSIM as the part of the loss function in the 3D U-Net and the generator part of the GANs. To calculate the SSIM for the 2D GAN, C_1 =0.0001 and C_2 =0.0009, 2D Gaussian filter with window size = 11×11, and σ_G=1.5 were used. To calculate the SSIM for the 3D networks including TAV-GAN, Temporal-GAN, Volumetric-GAN, and 3D U-Net, C_1 =0.0001 and C_2 =0.0009, 3D Gaussian filter with window size = 11×11×11, and σ_G=1.5 were used.

APPENDIX II 2D-GAN

Network architecture. The detailed network architecture for 2D GAN is shown in Figure A-1. The generator is a 2D U-Net which consists of two paths: (I) the encoder path, which includes four downsampling blocks; (II) the decoder path, which contains four up-sampling blocks. Each block has two convolutional layers, with each layer containing learnable convolution filters followed by the non-linear activation function Leaky ReLU (LReLU). Convolutional layers in the first block of the network contain 64 convolutional kernels, and the number of kernels doubles in each deeper block. Down-sampling and up-sampling blocks in the encoder and decoder paths are connected via average pooling (strides = 2) and up-sampling (strides = 2). A skip connection is used to pass the data between each pair of same-sized up-sampling and down-sampling blocks. The discriminator is a 2D binary classifier which contains four downsampling blocks. Each block contains two convolutional layers in which each convolutional layer contains convolutional kernels followed by LReLU. The starting number of channels used in the discriminator was 64, which was doubled in each deeper block. The last two layers are the fully connected layer followed by dropout and LReLU, and a single decision fully connected layer with a sigmoid activation function. Discriminator takes the magnitude of the generated images to decide whether it is "generated" or "clean" images. The input and output of the generator for the 2D GAN in the training stage is a complexed-valued image patch with size 320×192×2 (real and imaginary), and magnitude-valued image patch with size 320×192×1, respectively.

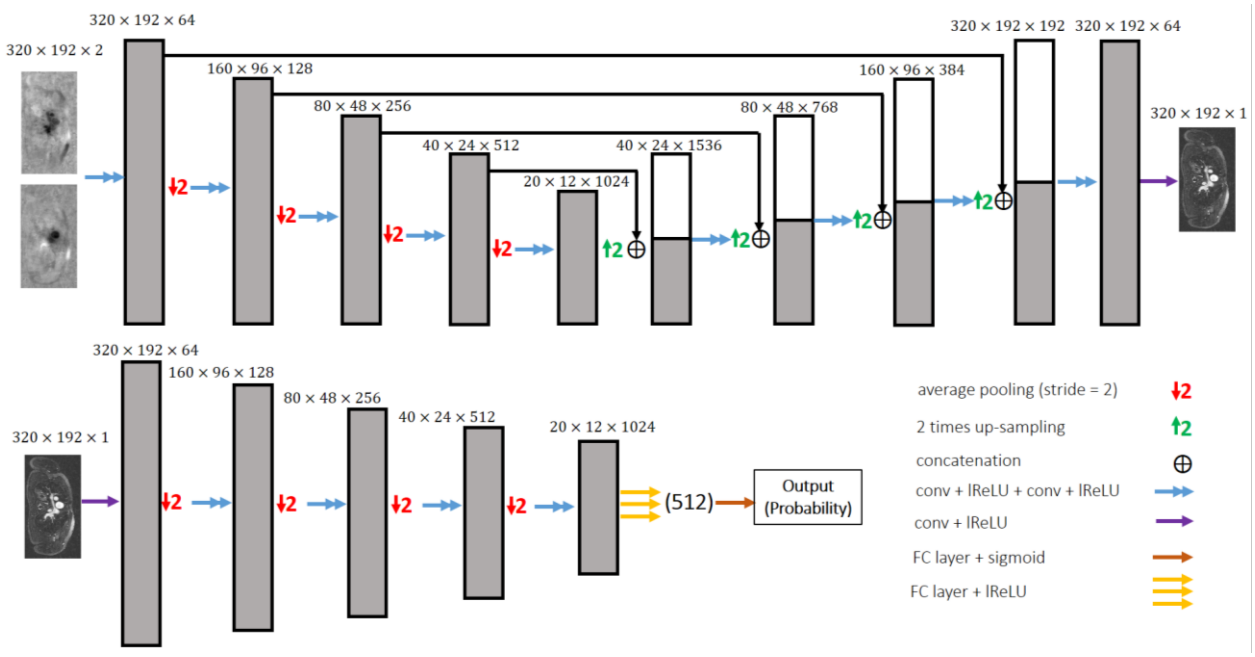

Figure A-1. The detailed network structure for 2D GAN. The generator part is a 2D U-Net with 4 downsampling blocks and 4 up-sampling blocks. The discriminator part is a 2D binary classifier with four downsampling blocks. The number of the convolutional kernels and type of the activation functions are reported in the Figure. Network training was performed on the image patches with size 320×192.

Loss function. The total loss function of the generator part of the 2D GAN $L_{G^{2D}}^{Total}(.)$ is a linear combination of the adversarial loss $L_{G^{2D}}^{a}(.)$, normalized L1 norm, and SSIM2D. The total loss function of the discriminator $L_{D^{2D}}^{Total}(.)$ is an adversarial loss $L_{D^{2D}}^{a}(.)$. Equations (A-7) and (A-8) formulated the generator's objective function and the discriminator's objective function, respectively:

$$\min_{\theta_{g^{2D}}} L_{G^{2D}}^{Total}\left(x^{i,t},\, G^{2D}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right) = \min_{\theta_{g^{2D}}} \gamma\left[L_{G^{2D}}^{a}\left(D^{2D}\left(x^{i,t};\theta_{d^{2D}}\right),\, G^{2D}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right)\right] +$$

$$\lambda\left[\frac{1}{N}\left\|x^{i,t} - G^{2D}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right\|_1\right] - \zeta\left[SSIM_{2D}\left(x^{i,t}, G^{v}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right)\right] \quad \text{(A-7)}$$

$$\min_{\theta_{d^{2D}}} L_{D^{2D}}^{Total}\left(D^{2D}\left(x^{i,t};\theta_{d^{2D}}\right),\ G^{2D}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right) =$$

$$\min_{\theta_{d^{2D}}} \gamma\left[L_{D^{2D}}^{a}\left(D^{2D}\left(x^{i,t};\theta_{d^{2D}}\right),\ G^{2D}\left(\tilde{x}_u^{i,t};\theta_{g^{2D}}\right)\right)\right] \quad \text{(A-8)}$$

Where $\tilde{x}_u^{i,t}$, $x^{i,t}$ stands for the aliased and respiratory motion-corrupted, and un-aliased and free of the motion 2D image patches for the t^{th} cardiac phase of the i^{th} patient case. γ, λ, and ζ are the hyperparameters that control the contribution of the adversarial loss, spatial sparsity and local patch wise similarity. N is the normalization factor and is equal to the number of the pixels inside $x^{i,t}$.

Training procedure. The training process for the 2D GAN is similar to the training process of the TAV-GAN. As shown in Figure A-2, the training process consists of five stable phases and four transition phases. The training started with a first stable phase. Only the layers with the lowest resolution are built and trained for an epoch in the first stable phase. Then first transition phase is started where new layers are added gradually to the lowest resolution layer to transit to the second stable phase. It is important to emphasize that as shown in Figure A-2, after each transition resolution of the image is doubled. In the transition phase, new layers were added with weight 1-α to the existing layers with weight α. The parameter α was linearly decreased from 1 to 0 through the iterations of the epoch's number. For instance, from the beginning of the transition phase (α=1), the newly added layers were getting zero weight, and as α decreases, the new layers had more weight until the part of the existing layers were faded (α=0). Once α reached 0, the transition phase was finished, and the next stable phase was started. These stable and transition phases were alternated while more layers were added progressively until the stable phase 5 was finished, which concluded the training process. Figure A-3 shows the first stable and transition phases for the 2D GAN. For the first to fourth stable and transition phases, the network is trained for an epoch. The

number of the required epochs for the last stable phase is decided empirically. Two criteria for stopping the training process were considered: 1) outputs' quality through the training and 2) equilibrium state of the adversarial loss for the generator and the discriminator.

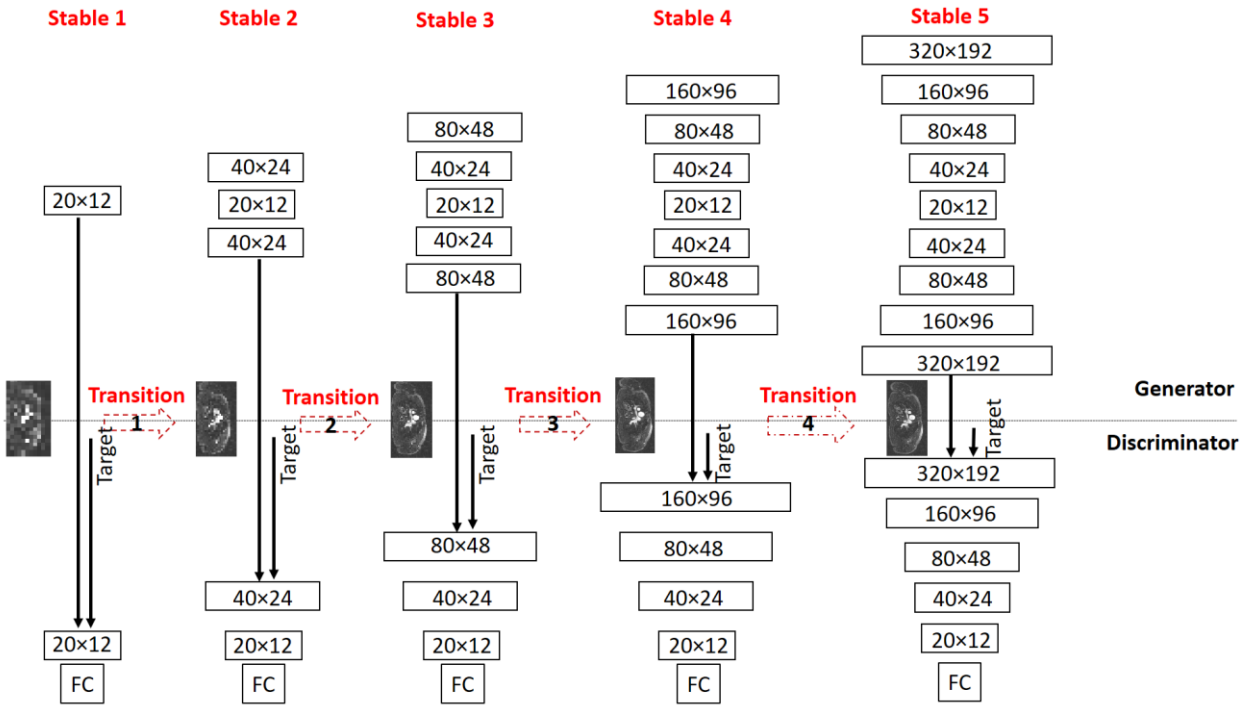

Figure A-2. Progressive training strategy for 2D GAN. Intuitively, building the network with few layers with low resolution and training them and gradually adding more layers to reach the high-resolution images can alleviate the training process of the GANs. The training procedure contains five stable phases and four transition phases. As can be seen, in the stable phase 1, only layers with the lowest resolution were built. In the transition phase 1, new layers were gradually added to the old layers to reach stable phase 2. Parameter α controls the rate of gradual pointwise addition. It linearly reduced from 1 to 0 through the iterations of the training in each transition phase. Sample of transition and stable phases were explained in Figure A-3. This alternation between stable and transition phases was continued until to reach to the last stable phase 5. For the last stable phase, training was performed for the number of epochs. The number of the required epochs was decided based on the quality of the test results in the training stage, and the equilibrium state of the generator loss and the discriminator loss.

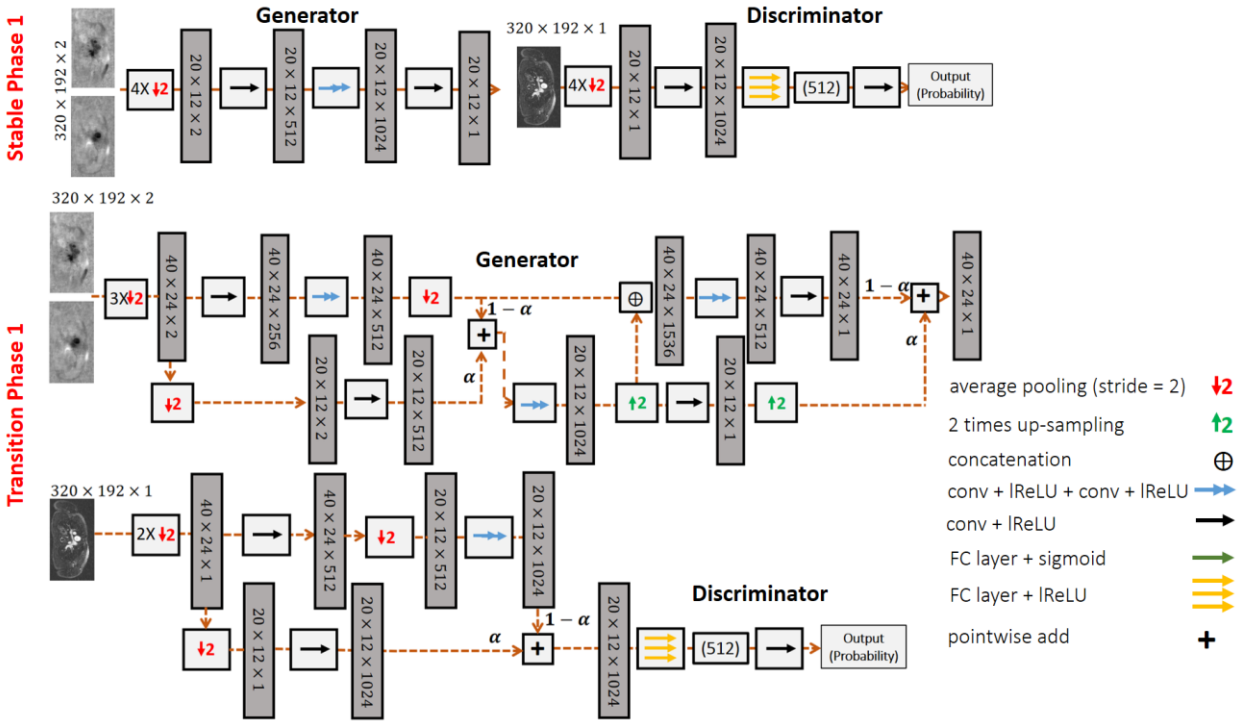

Figure A-3. Illustration of the stable and transition phases of the 2D GAN in this work. For the sake of simplicity, we only showed the first stable and transition phases. Only layers with the lowest resolution were built for the generator and the discriminator in the first stable phase. The input complex image was downsampled four times and fed to the generator. The first convolution layer in the generator and the discriminator is increasing the channel dimensions of the input. The network was trained for an epoch in the first stable phase. Then, in the first transition phase, layers with twice resolutions were added gradually to the pre-trained layers. As can be seen, new layers were added to the generator and the discriminator progressively. The parameter α controls the addition process. It is linearly decreasing from 1 to 0 through all iterations in the epochs. We trained this phase only for an epoch. To make the idea clear, for α=1, we are at the beginning of the transition phase. For α=1, the graph for the generator and the discriminator is the same as the graph in the stable phase 1. Suppose α=0; it means that the first transition phase is finished, and training will enter the second stable phase. By considering α=0, it can be seen that adapting layers in the first stable phase were faded, and new layers with higher resolution were added to the graph.

Training Parameters. For the 2D GAN, $\gamma = 1$, $\lambda = 0.6$ and $\zeta = 0.4$ are selected based on the limited search as the weight of the adversarial loss, normalized L1-loss, and $SSIM_{2D}$ loss. Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 64, an initial learning rate of 0.0005 for the generator, and an initial learning rate 0.00005 for the discriminator. Weights of the network are initiated with random normal distributions with a variance of $\sigma = 0.01$ and mean

μ=0. The training was performed with the Pytorch interface on a commercially available graphics processing unit (GPU) (NVIDIA Titan RTX, 24GB RAM).

APPENDIX III Data-Preparation

As shown in Figure A-4 (b), each ROCK MUSIC2 scan continuously acquired NL Cartesian k-space lines grouped in quasi-spiral interleaves in the Ky-Kz plane that are arranged in a golden-angle manner, shown in Fig. A-4 (a). For each ROCK MUSIC raw data in Group A, a pair of image volumes were reconstructed for network training: the reference image and the highly accelerated, aliased and respiratory motion-corrupted image. To reconstruct the reference image, data were binned into 9-12 cardiac phases of the end-expiration respiratory state by using the cardiac and respiratory self-gating signal derived from the k-space center lines as shown in Figure A-4(b) and reconstructed based on Equation (A-9)[116,234]:

$$\hat{d} = \underset{d}{\operatorname{argmin}} \sum_{i=1}^{N} \|DFS_i d - m_i\|_2^2 + \lambda_1 \|R_1 d\|_1 + \lambda_2 \|R_2 d\|_1 \quad \text{(A-9)}$$

Where F, Si, and D are the Fourier transform, sensitivity maps, and undersampling mask, respectively. d is the multiphase images, mi is the acquired undersampled k-space from each of the N receiver coil channels. R1 is the spatial wavelets and R2 is the temporal total variation. Hyperparameters λ_1 and λ_2 control the weight of the regularizers R1 and R2, respectively. The k-space under-sampling factor after cardiac and before respiratory motion SG ranged 2.8X-7.9X. To reconstruct the "highly accelerated" image volume, as shown in Fig. A-4 (c), we extracted the first $M = min\,(50000, N_L/2)$ k-space lines out of the data, resulting in a further retrospective under-sampling of the acquired data by a factor of at least 2. Because the quasi-spiral k-space interleaves were arranged in a golden-angle manner, the k-space sample uniformity is maintained even when the second half (or more) of acquired data was discarded. The total k-space under-sampling factor was 10.7X-15.8X for the "highly accelerated" image volumes.

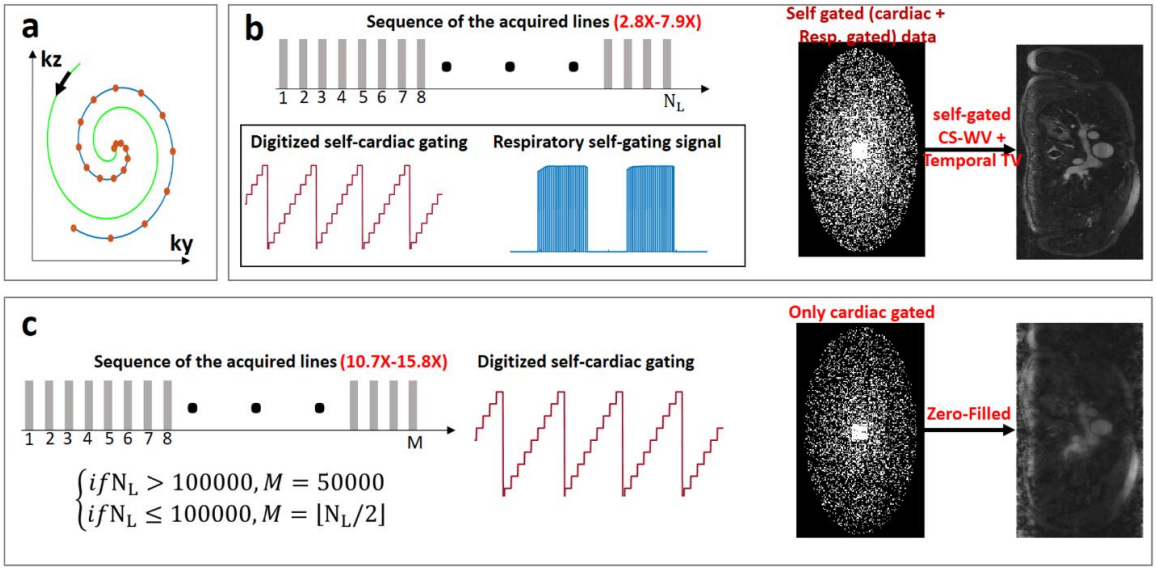

Figure A-4. Data preparation process: (a) shows the ROtating Cartesian K-space (ROCK) sampling strategy used to acquire the data. (b) shows the SG CS-WV reconstruction process to create the clean reference volumetric images. (c) shows the zero-filled reconstruction process to create the aliased, respiratory motion-corrupted images. As shown in (c), the first half of the acquired lines (if NL<100000 lines) or the first 50000 of the acquired lines (if NL>100000) were used to create the inputs for training and testing the network. Also, only a self-cardiac gating signal is used to sort the data to multiple cardiac phases. No respiratory motion gating was performed when generating the input images in (c).

We note that, although the data were retrospectively under-sampled, we expect our data to accurately represent a prospectively under-sampled in vivo imaging scenario with the same under-sampling factor, because the prospective data would have been acquired using the exactly the same sequence timing and temporal order for the k-space lines and quasi-spiral interleaves. We subsequently binned the resulting highly accelerated k-space data into appropriate cardiac phases using the cardiac-gating signal, zero-filled each cardiac phase data, performed an inverse Fourier transform, and finally combined the resulting multi-coil images to a single complex coil image using fast coil combination algorithm. The highly accelerated images, in the absence of any compressed sensing reconstruction and respiratory motion gating, had significant under-sampling aliasing artifacts and respiratory motion artifacts. Both the reference images and the highly accelerated images were normalized by subtracting the complex mean within the image volume

and dividing by the absolute value of twice the standard deviation of the same volume. The highly accelerated image volumes were formatted as individual 4D tensors with its complex values expressed as real and imaginary channels. The magnitude of the normalized reference images were formatted as a 4D tensor as well with a single (magnitude) channel. To minimize the background effect, 10 voxels from the edge of the tensors were cropped. To prepare for network training, paired patches, of size 64×64×64×2 from the highly accelerated images and of size 64×64×64×1 from the reference images, were extracted randomly from the cropped tensors and used as an input and target, respectively, in the training phase of the 3D U-Net, the Volumetric-GAN, and the TAV-GAN. For training the Temporal-GAN, the input was formatted as a real-valued 4D tensor with the magnitude of the three sequential cardiac phases t-1, t, t+1 in the channel dimension of the tensor, and the training target was the magnitude of the reference image corresponding to cardiac phase t. Subsequently, paired patches with sizes 64×64×64×3 for the input, and 64×64×64×1 for the target 4D tensor, was extracted randomly and used as an input and target in the training phase for the Temporal-GAN. It is worth noting that in the Temporal-GAN, to prepare the data for the first and last cardiac frames, cardiac frames were assumed cyclic. For instance, for the last cardiac frame t as the target, three aliased and respiratory corrupted cardiac frames t-1, t, 1 were stacked in the channel dimension as the Temporal-GAN input.

For 2D GAN, the reference images and the highly accelerated images were normalized slice-by-slice by subtracting the complex mean within the image slice, followed by division by the absolute value of the standard deviation of the slice. The input and target of the generator for the 2D GAN in the training stage was cropped from the slice-by-sliced normalized highly accelerated images and reference images and formatted as a complexed-valued tensor with size 320×192×2 (real and imaginary), and a magnitude-valued tensor with size 320×192×1 , respectively.

For the testing data sets in Groups B1 and B2, we reconstructed both the reference image volumes and the highly accelerated image volumes as well. Although data in these Groups were not used in network training, the reference images were used in the network performance evaluations and comparisons.

APPENDIX IV Sharpness Analysis

The normalized Tenengrad focus measure[100,101] was used to quantify the sharpness of the reconstructed respiratory motion-corrected results with different networks. In general, to compute the Tenengrad focus measure, the image is convolved with a Sobel operator, and the square of all the magnitudes greater than a threshold is reported as a focus measure. Equation (A-10) formulates the Tenengrad measure:

$$F_{Tenengrad} = \sum_{i,j}[I(i,j) ** S]^2 + [I(i,j) ** S^T]^2 , \quad \text{(A-10)}$$

Where $I(i,j)$ shows the image and S is the Sobel operator: $S = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -2 \end{bmatrix}$.

Because of the difference in the size of the testing cases, the mean of the Tenengrad focus measure without threshold was calculated and reported as a sharpness score of an image. To calculate the results' sharpness score from different methods, we first cropped the cardiac region, and then we computed the mean of the Tenengrad focus measure for each slice of the cropped region and normalized them based on the calculated mean of the Tenengrad measure for the corresponding slice of the cropped region in the reference SG CS-WV images. Then, the normalized values were averaged over the slices inside a cropped cardiac region and cardiac phases to represent a single sharpness number for each case. We excluded the 2D GAN in our sharpness analysis because of its inferior image quality with more residual artifacts than other methods.

APPENDIX V 3D spatiotemporal GAN

Purpose: The goal of this study is to compare a 3D spatiotemporal GAN against the Temporal-GAN qualitatively and quantitatively.

Method: A 3D spatiotemporal GAN was trained based on the ROCK MUSIC data in this work. To circumvent limitations in GPU memory, we first performed a Fourier Transform on the ROCK MUSIC data in the readout direction, to divide the 4D (3D spatial + cardiac phase) data into a contiguous series of 2D dynamic slices in the readout direction, each slice having two spatial dimensions and one temporal dimension. We included 9 cardiac phases for each 2D dynamic slice. The same Fourier Transform in the readout direction was performed for both the highly-accelerated motion-corrupted datasets and the SG CS-WV reference datasets. We subsequently trained a 3D spatiotemporal GAN that takes these individual 2D dynamic slices as the input such that the GPU memory is not saturated. This 3D spatiotemporal GAN takes advantage of 2D spatial information and the temporal information, i.e., redundant information through the sequential 2D cardiac frames, to recover the clean images from the aliased and respiratory motion affected images. The network structure for the 3D spatiotemporal GAN is similar to the Temporal-GAN, except that the last convolutional layer of the network has nine kernels. For the 3D spatiotemporal GAN, a combination of two-loss functions including the content loss ($\lambda = 0.5$, $\zeta = 0.3$), and adversarial loss ($\gamma = 1$) were considered. The progressive training strategy as described in the main manuscript was used to train the network. The network's trainable weights were initiated with random normal distributions with a variance of $\sigma = 0.01$ and mean $\mu=0$. For the 3D spatiotemporal GAN, the Adam optimizer was used with the momentum parameter $\beta =0.9$, mini-batch size= 16, an initial learning rate of 0.0001 for the generator, and an initial learning rate of 0.00001 for the

discriminator. To evaluate the image quality of the 3D spatiotemporal GAN, we randomly selected 7 cases from Group B1 and Group B2, and asked two blinded radiologists to rank reconstructed dynamic image volumes using either the Temporal-GAN and the new 3D spatiotemporal GAN.

Results: Figure A-5 shows the qualitative reconstruction and respiratory motion correction results for the two patient cases drawn from the datasets Group B1 and Group B2 for the three techniques, including the (a)Temporal-GAN, (b) 3D spatiotemporal GAN, and (c)2D GAN. The first row of each subpanel in Figure A-5 shows a coronal section of the results, and the second and third rows show the zoomed cardiac region and the temporal difference maps between two sequential cardiac frames. Based on the temporal difference maps in both patient cases, the flickering artifacts were substantially reduced in both Temporal-GAN and the 3D spatiotemporal GAN in comparison to the 2D GAN. Both Temporal-GAN and the 3D spatiotemporal GAN had better performance in removing the aliasing and respiratory artifacts from the image than the 2D GAN. Based on blinded evaluations of 7 cases, both radiologists ranked the Temporal-GAN higher than the 3D spatiotemporal GAN in 5 cases, and they were split in the remaining two cases (i.e. one ranked Temporal-GAN higher, and one ranked 3D spatiotemporal GAN higher in these two cases). SSIM (±SD), nRMSE (±SD)) which were calculated based on the testing dataset Group B1 for the Temporal-GAN, 3D spatiotemporal GAN, and the 2D GAN was (0.746±0.0495, 0.036±0.0072), (0.682±0.061, 0.053±0.010), and (0.481±0.0594, 0.072±0.0138), respectively. The mean of the normalized Tenengrad focus measure (±SD) for the reconstructed and respiratory motion-corrected results obtained by the Temporal-GAN and the 3D spatiotemporal GAN was 0.702±0.1408 and 0.762±0.146, respectively.

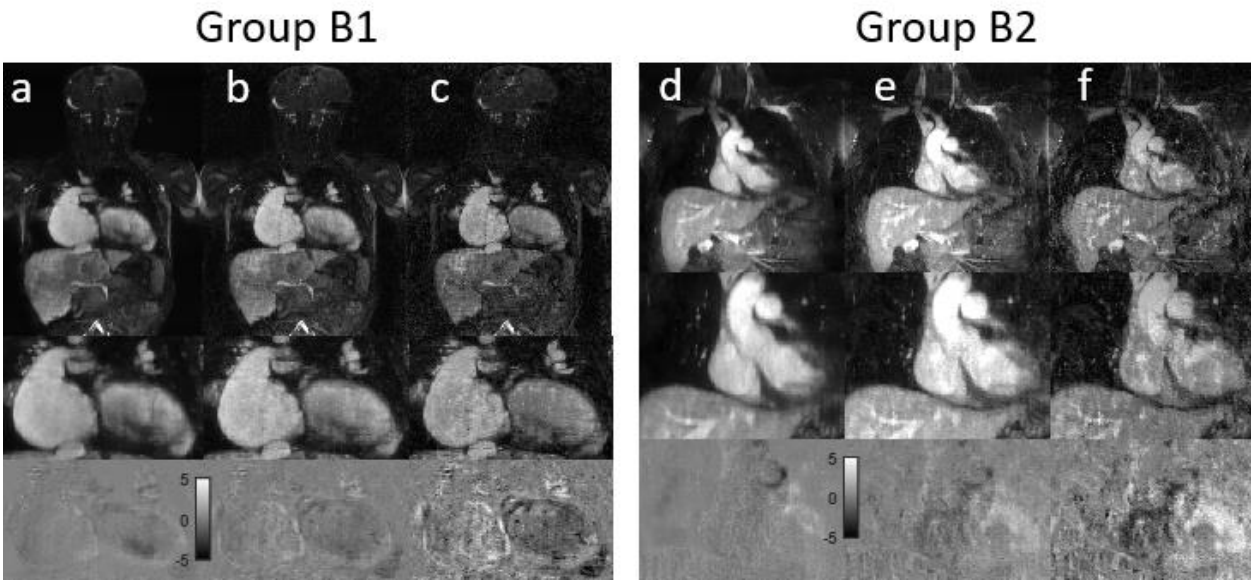

Figure A-5. Qualitative results obtained by three techniques for two patient cases selected from the testing datasets Group B1 and Group B2. It shows the reconstruction and respiratory motion correction results for the Temporal-GAN (a, d), 3D spatiotemporal GAN (b, e), and 2D GAN (c, f). The magnified heart region is shown for each image (2nd row of each panel). The bottom row of each panel shows the temporal difference maps between two sequential cardiac frames. Both Temporal-GAN and 3D spatiotemporal GAN achieved better results regarding aliasing and respiratory motion and flickering artifacts reduction than the 2D GAN.

APPENDIX VI Cardiorespiratory gated inputs vs. the cardiac gated inputs

Purpose: In this study, we sought to investigate the difference between TAV-GAN trained based on 1) cardiac-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction as a reference, and the TAV-GAN trained based on 2) cardiorespiratory-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction as a reference.

Method: As illustrated in the main manuscript, TAV-GAN was trained based on the cardiac-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction images as the target. Another TAV-GAN with the same training procedure and parameters was trained based on the cardiorespiratory-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction images as the target. The performance of two TAV-GANs was compared qualitatively with regard to regular respiratory motion artifact and irregular respiratory motion artifact.

Results: Figure A-6 shows the qualitative results obtained by SG CS WV (a, d), TAV-GAN trained based on the cardiac-gated zero-filled images as input (b, e), and TAV-GAN trained based on cardiorespiratory-gated zero-filled images as input (c, f) for two representative cases selected from Group B1 and Group B2. For the patient with regular breathing, there was no apparent difference between the two TAV-GANs – both of them provided good image qualities. For the Group B2 patient, the TAV-GAN trained based on the cardiac-gated zero-filled images as input provided better overall image quality.

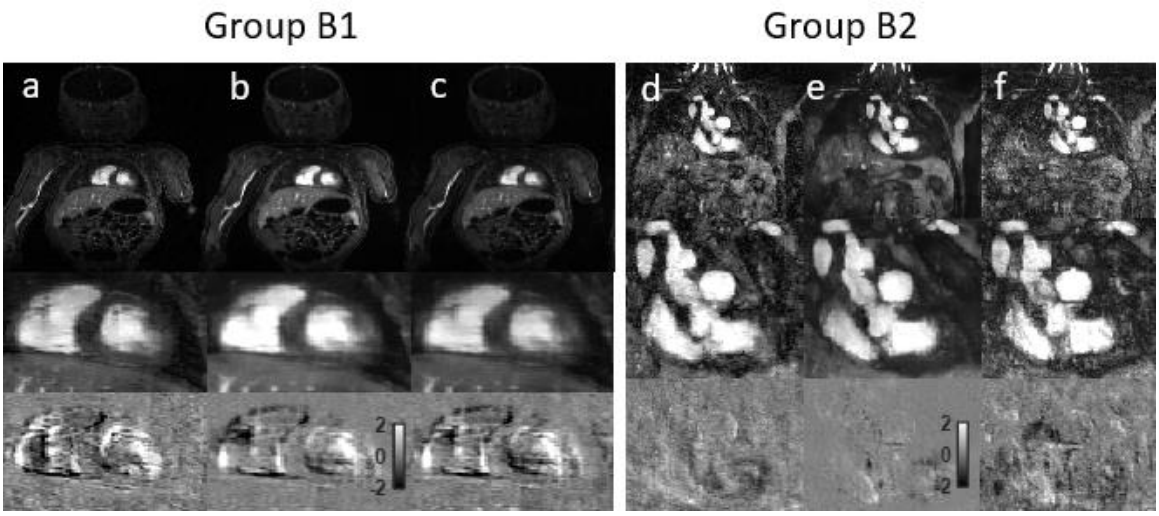

Figure A-6. Qualitative representative results of two unseen cases from Group B1 and Group B2. (a-c) show the un-aliased and respiratory artifact-corrected images from a patient with a regular respiratory pattern during scanning, obtained by SG CS-WV, TAV-GAN (trained based on cardiac-gated zero-filled images as the input), and TAV-GAN (trained based on cardiorespiratory gated zero-filled images as the input), respectively. (d-f) show images using the same techniques from a patient with irregular respiratory motion. The TAV-GAN trained based on the cardiorespiratory gated zero-filled images as the input would reduce the respiratory and aliasing artifacts in the case with regular breathing, but it seems in the case with irregular breathing, its performance dropped substantially. In each panel, the 2nd rows are amplified images of the heart region, and the third rows are temporal difference maps for two sequential cardiac phases.

Discussion: For the presented test results (See Fig. A-6) from Group B1, which had regular breathing and was similar to the data in training datasets (Group A), both TAV-GANs showed similar performance. However, the TAV-GAN trained based on the cardiorespiratory gated zero-filled images, could not provide satisfactory results in the presence of irregular breathing (See Fig. A-6; Group B2). The TAV-GAN trained based on the cardiac gated zero-filled images as the input shows more robustness in the testing stage on the data with irregular breathing. We speculate that when the TAV-GAN is trained on cardiorespiratory-gated zero-filled images as the input, it would only learn how to remove under-sampling aliasing artifacts, which is easier than removing the aliasing and respiratory artifacts simultaneously. This drawback may compromise the network's ability in removing any residual motion after respiratory self-gating in the testing stage.

APPENDIX VII Comparison of different networks, hyper-parameters and learning rate annealing methods

Methods: For the proposed model, we evaluated the performance of the model with different hyper-parameters. While keeping the other hyper-parameters the same as those described in the chapter 6, three groups of comparison experiments were performed, based on the radial T1-T2 sequence: 1) we compared different numbers of ResNet blocks (R_n = 0, 2, 4, 6 while keeping stride=2 for the last two convolutional layers); 2) we compared different stride sizes (stride = 1, 2, 3 while keeping R_n=4) used in the last two convolutional layers, and 3) we compared two different learning rate annealing methods (step decay and exponential decay) during the training. In addition, the dense network used in the DRONE network[168] was also evaluated against the proposed network.

For the learning rate annealing method comparison, the performance of two other learning rate annealing methods were compared with the proposed methods. The first one was a step decay method by reducing the learning rate by half every 100 epochs. The second one was the exponential decay method which we set the learning rate (epoch) = initial learning rate × exp ((1-epoch) / 100). For both methods compared, we used a relatively optimized initial learning rate = 0.001 and a total of 600 epochs.

To adapt to the dense network used in DRONE for our comparison, the acquisition time stamps and the signal were combined into a vector of 121 (=11×10+11) nodes as the input layer, followed by two 300 × 300 hidden layers and the output layer.

We use the same simulated training set (SNR = 20) and validation set (SNR = 20) as described in the chapter 6 to train different networks. An independent testing set (SNR = 20) was used to compare results of various networks or hyper-parameters. The mean square errors (MSEs) for T1

and T2 from the testing result were used as the evaluation metric. For the radial T1-T2 sequence, the proposed network (e.g. R_n = 4, stride = 2) was trained and evaluated 5 times to establish the range of MSEs for better comparison. The other networks or hyper-parameters were trained and evaluated once. For MOLLI, the models with R_n = 0, 2, 4 and 6 were trained and evaluated once.

Results: Based on the simulated data of the radial T1-T2 sequence, the testing set MSE and the number of trainable parameters for different networks or hyper-parameters were listed in Table A-1. In the testing set (SNR = 20), the mean MSE by DeepBLESS trained using the proposed network was 489.23 ± 0.23 ms^2 (trained 5 times, minimum MSE = 489.02 ms^2 and maximum MSE = 489.60 ms^2) for T1 and was 8.32 ± 0.03 ms^2 (trained 5 times, minimum MSE = 8.28 ms^2 and maximum MSE = 8.36 ms^2) for T2. In comparison, the MSE by the network used in DRONE[168] was larger (503.58 ms^2 for T1 and 8.49 ms^2 for T2). The proposed network with different number of ResNet blocks (R_n = 0, 2, 6) generated MSE of 508.78 ms2, 492.36 ms^2 and 490.86 ms^2 for T1 and 9.71 ms^2, 8.42 ms^2, 8.34 ms^2 for T2, respectively, all higher than that using Rn = 4 except the model with R_n = 6, which generated MSE similar to that using R_n = 4 for T2 but still generated higher MSE for T1. For the last two convolutional using different strides, using stride =3 generated similar MSE (489.13 ms^2 for T1 and 8.32 ms^2 for T2) compared to stride = 2 (proposed), and using stride = 1 generated greater MSEs for both T1 (490.49 ms^2) and T2 (8.37 ms^2). For different learning rate annealing methods, the proposed approach generated better results compared to the step decay (MSE = 489.78 ms^2 for T1 and 8.39 ms^2 for T2) and exponential decay (MSE = 497.52 ms^2 for T1 and 8.41 ms^2 for T2). Examples of the training and validation loss against the number of epochs for the three learning rate annealing approaches are shown in Figure A-7.

Based on the simulated data of the MOLLI sequence (Table A-2), the proposed network with R_n = 4 and R_n=6 achieved lower MSE compared to models with fewer layers.

In conclusion, for the networks or hyper-parameters compared, the proposed network with the proposed hyper-parameters provided the lowest MSE.

Table A-1: The mean square error (MSE) in the validation set (SNR = 20) of the radial T1-T2 sequence using different networks, hyper-parameters and learning rate annealing methods for DeepBLESS.

Model	Trainable parameters	Mean square error (MSE)	
		T1 (ms^2)	T2(ms^2)
Proposed	32,225	489.23 ± 0.23	8.32 ± 0.03
DRONE	127,501	503.58	8.49
$R_n = 0$	7,393	508.78	9.71
$R_n = 2$	19,809	492.36	8.42
$R_n = 6$	44,641	490.86	8.34
step decay	same as proposed	489.78	8.39
exponential decay	same as proposed	497.52	8.41
stride = 1	32,481	490.49	8.37
stride = 3	32,193	489.13	8.32

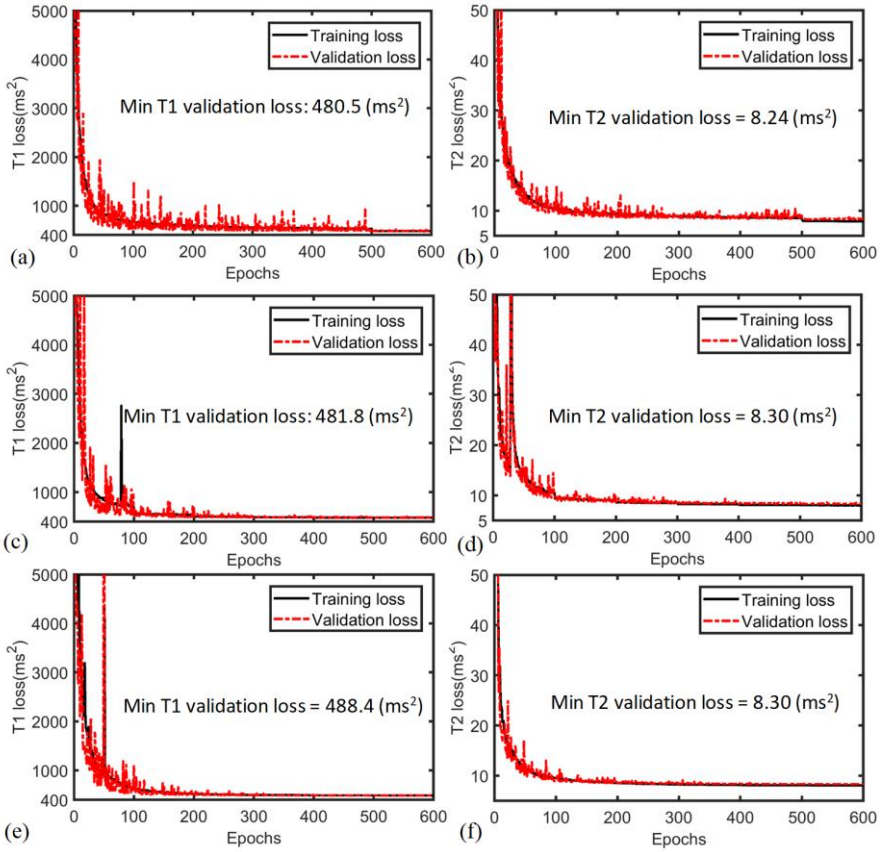

Figure A-7: The training and validation loss against the number of epochs using the proposed learning rate strategy (a for T1, b for T2), conventional learning rate step decay (c for T1, d for T2) and learning rate exponential decay (e for T1, f for T2). The proposed learning rate strategy achieved the best validation loss.

Table A-2: The mean square error (MSE) in the testing set (SNR = 20) of the MOLLI sequence using 0-6 Resnet blocks (Rn =0, 2 ,4 and 6) for DeepBLESS.

Model	# of Trainable parameters	T1 MSE (ms^2)
Proposed (R_n=4)	31,329	117.9
R_n = 0	6,497	130.6
R_n = 2	18,913	119.3
R_n = 6	43,705	117.9

Table A-3: The size of the intermediate features after each of the 11 convolutional layers of DeepBLESS network.

Layers	Output Shape for different layers		
	Radial T1	Radial T2	MOLLI T1
Input layer	(B*, 11, 11)	(B, 11, 11)	(B, 8,2)
Convolution layer 1	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 2	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 3	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 4	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 5	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 6	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 7	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 8	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 9	(B, 11, 32)	(B, 11, 32)	(B, 11, 32)
Convolution layer 10	(B, 6, 32)	(B, 6, 32)	(B, 6, 32)
Convolution layer 11	(B, 3, 32)	(B, 3, 32)	(B, 3, 32)
Flatten layer	(B, 96)	(B, 96)	(B, 96)
Output	(B, 1)	(B, 1)	(B, 1)

* B indicates the batch size in training

Table A-4: The mean percentile absolute T1 and T2 reconstruction error at different testing data noise level (SNR = 10 - 100)

Testing data noise	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%	Average
SNR of testing data	100.0	50.0	33.3	25.0	20.0	16.7	14.3	12.5	11.1	10.0	
Training data SNR	Mean T1 percent error (%)										
SNR = 100	0.39	0.76	1.17	1.57	2.01	2.46	2.91	3.38	3.87	4.36	2.29
SNR =20.0	0.44	0.73	1.04	1.36	1.70	2.03	2.36	2.68	3.05	3.37	1.88
SNR =11.1	0.67	0.88	1.14	1.43	1.74	2.05	2.38	2.67	3.03	3.35	1.93
SNR = 10 -100	0.42	0.72	1.05	1.37	1.71	2.04	2.37	2.69	3.05	3.37	1.88
BLESSPC	0.39	0.68	1.02	1.35	1.69	2.03	2.37	2.69	3.06	3.39	1.87
Training data SNR	Mean T2 percent error (%)										
SNR = 100	0.63	1.18	1.75	2.36	2.99	3.63	4.31	5.05	5.87	6.71	3.45
SNR =20.0	0.80	1.18	1.63	2.11	2.59	3.08	3.55	4.07	4.59	5.11	2.87
SNR =11.1	1.62	1.8	2.06	2.39	2.76	3.17	3.59	4.04	4.51	4.96	3.09
SNR = 10 - 100	0.78	1.19	1.67	2.16	2.65	3.13	3.60	4.10	4.59	5.08	2.90
BLESSPC	0.52	1.21	1.67	2.16	2.64	3.15	3.64	4.16	4.68	5.19	2.90

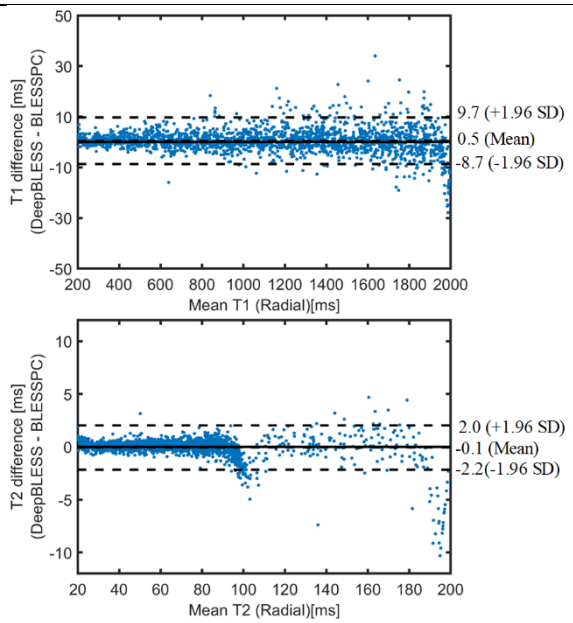

Figure A-8: Bland Altman analysis (2000 data points) between DeepBLESS and DeepBLESS for testing data with at least 1 missed heartbeat (SNR = 20).

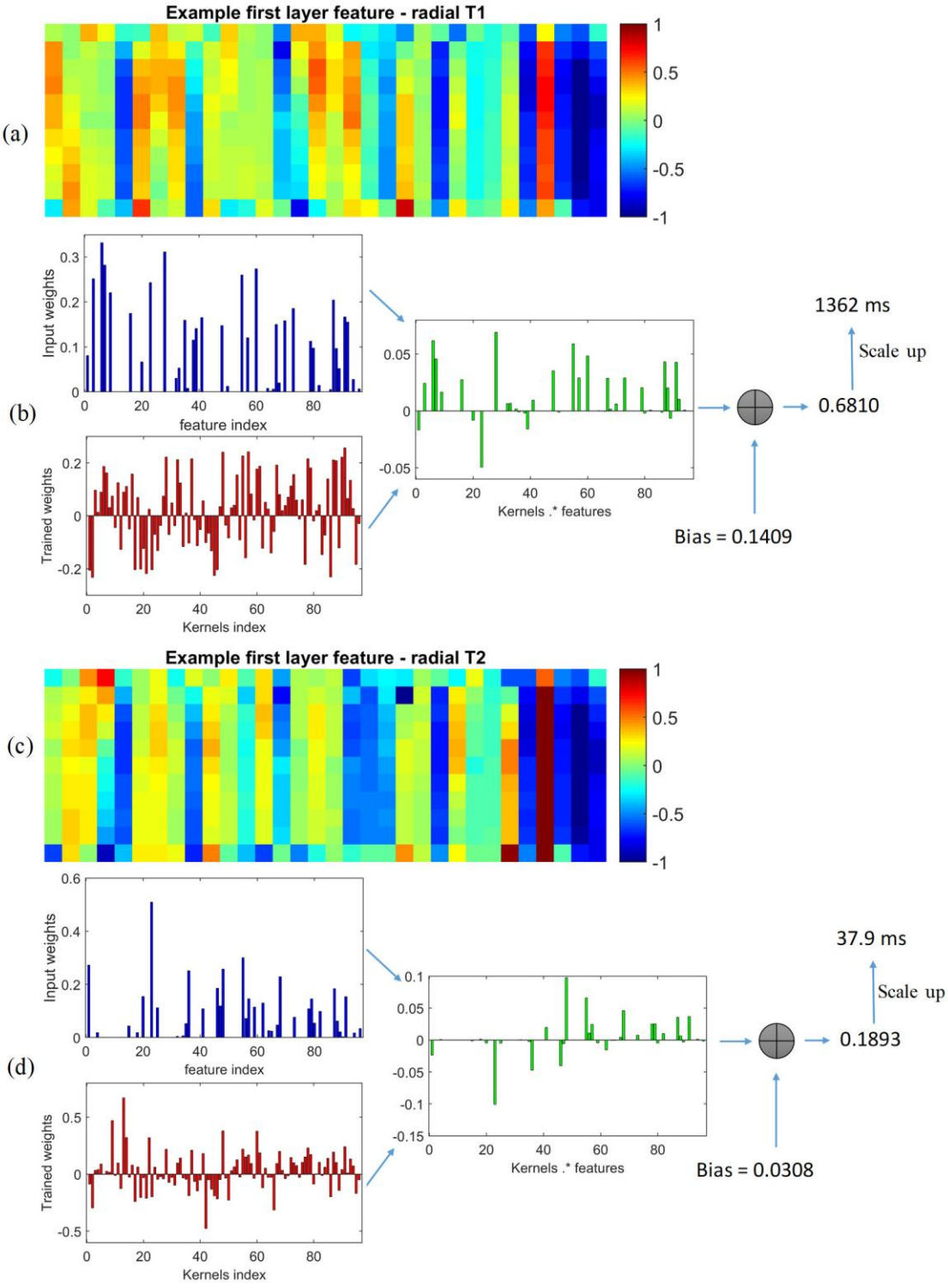

Figure A-9: Example features of DeepBLESS T1 and T2 models for a sample (BLESSPC T1 = 1361 ms, T2 = 37.7 ms) of the testing set (SNR = 20) simulated based on the radial T1-T2 sequence: (a, c) First

layer feature map for DeepBLESS T1 and T2, respectively; (b,d)The last layer's input feature, kernels and the final predication results for DeepBLESS T1 and T2, respectively.

APPENDIX VIII Comparison Study

Purpose: The aim of this study is comparing our proposed artery and vein segmentation approach with the Fuzzy based method which has been proposed by Lei et al[194]. The comparison was performed in the calf region based on the 7 test cases.

Method: Lei et al. used Fuzzy logic to perform artery and vein segmentation in the thigh region for high-resolution Gd-MRA images[194]. Their proposed approach includes two sequential stages. In the first stage, they use absolute fuzzy (scaled-based) connectedness1 to extract the whole vasculature from the high-resolution Gd-MRA images in the thigh region. In the second stage, they use relative fuzzy connectedness[194] to separate the arteries from the veins in the extracted vasculature. To separate the arteries from the veins in the second stage, competition is set up via the relative fuzzy connectedness for seed voxels specified in the arterial and venous branches to grab voxels based on their relative strengths of connectedness with the two sets of seed voxels. The two main stages of their algorithm are summarized as the followings:

Stage 1. Extraction of the entire vessel structure from the other clutters and background via the absolute fuzzy connectedness in a given CE-MRA image:

a. Seeds specification for the blood vessels.

b. Blood vessel segmentation using absolute fuzzy connectedness (κFOE)[235].

Stage 2. Separation of the arteries from the veins via the iterative relative fuzzy connectedness in the segmented blood vessel structure:

a. Seeds specification for both arteries and veins.

b. Artery-vein separation by applying the iterative relative fuzzy connectedness (κIRFOE)[194].

All 7 test datasets in our study were first normalized between 0 to 1, and then a manual seeding process was performed. For the manual seeding, we put 2 seeds in each branch of the arteries and the veins and then used those initial seeds to calculate the absolute fuzzy connectedness values (Stage 1). For stage 2, we used the previous seeds and calculated the centerlines for the arterial and venous branches, and used the arterial and venous centerlines as the initial seed points to calculate the relative fuzzy connectedness values (Stage 2). Figure A-10 (a) shows examples of these initial points. Since the Fuzzy-based approach's output is not a binary map, we applied proper thresholding for each test case to convert them to the segmentation masks for the sake of quantitative and qualitative comparisons.

Results: Figure A-10(b-d) qualitatively compared the artery-veins segmentation results on the three exemplary coronal slices of the fuzzy-based approach (arterial branches: yellow contours; venous branches: green contours) against our proposed method (arterial branches: red contours; venous branches: blue contours). Figure A-10(e, f) shows the volume-rendered images for the segmented artery and veins obtained by the Fuzzy-based approach. Figure A-10(g, h) displays the volume-rendered images for the segmented artery and veins obtained by our proposed approach. As evident in Figure A-10(e-h), our proposed method captured more arterial and venous branches than the Fuzzy-based approach. Quantitatively, for the 7 test datasets, the Fuzzy-based approach and our proposed method achieved mean F1 (±SD) 0.7321±0.0921 and 0.8274±0.0152 for the segmentation of the arteries and 0.7863±0.0643 and 0.7405±0.1061 for the venous segmentation.

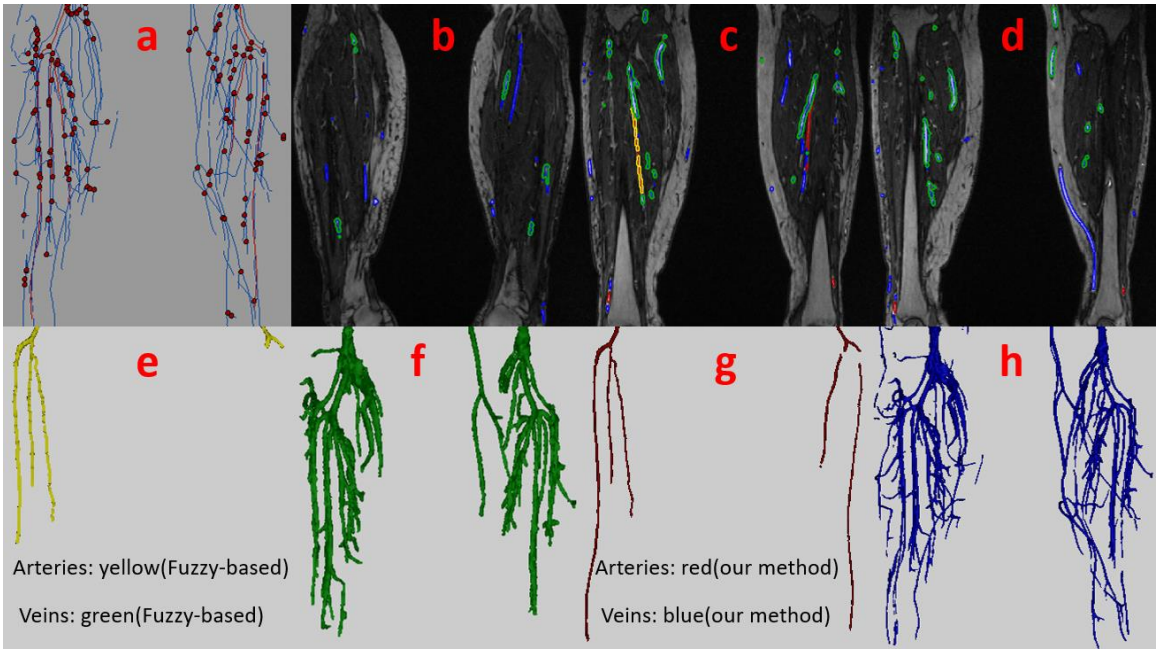

Figure A-10. Fuzzy-based approach by Lei et al vs. our proposed method: a) manual seeds for the first and second stage of the Fuzzy-based approach. Red spheres are the initial seeds used to perform the first stage of the Fuzzy-based segmentation (absolute fuzzy connectedness). Blue and red lines, which present the centerline of the arteries and veins, were used to complete the second stage of the Fuzzy-based approach (relative fuzzy connectedness). (b-d) shows the arteries and veins segmentation performance of the Fuzzy based approach (arterial branches: yellow contours; venous branches: green contours) and the proposed method (arterial branches: red contours; venous branches: blue contours) on three coronal views of the calf region. (e, f) and (g, h) shows the volume-rendered image of the arteries and veins for the Fuzzy-based approach and our proposed method, respectively. Qualitatively, our proposed method has superior performance over the fuzzy-based approach with respect to the artery and veins segmentation in the calf region.

Discussion: This mini-study showed that the proposed method achieved better qualitative and quantitative artery and vein segmentation in the calf region than the Fuzzy-based approach proposed by Lei et al. Accessing the initial seed points are essential for both stages of the Fuzzy-based approach. For the first stage, such initial seeds can be easily provided by users, but knowing the arteries and veins is required for the second stage, which is very challenging in the calf region. It worth mentioning that the fuzzy-based approach originally proposed to segment the arteries and veins in the thigh region in particular pelvic region where the user can easily recognize the arteries

and veins, and initialization of those branches are relatively more straightforward than the calf region.

REFERENCE

1. Sodickson DK, Manning WJ. Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays. Magn. Reson. Med. 1997; 38:591–603.

2. Griswold MA, Jakob PM, Nittka M, Goldfarb JW, Haase A. Partially Parallel Imaging with Localized Sensitivities (PILS). Magn. Reson. Med. 2000; 44:602–609.

3. Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. Magn. Reson. Med. 1999; 42:952–62.

4. Griswold MA, Jakob PM, Heidemann RM, Nittka M, Jellus V, Wang J, Kiefer B, Haase A. Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn. Reson. Med. 2002; 47:1202–10.

5. Lustig M, Donoho D, Pauly JM. Sparse MRI: The application of compressed sensing for rapid MR imaging. Magn. Reson. Med. 2007; 58:1182–95.

6. Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed Sensing MRI. IEEE Signal Process. Mag. 2008; 25:72–82.

7. Zhou Z, Han F, Ghodrati V, Gao Y, Yin W, Yang Y, Hu P. (2019), Parallel imaging and convolutional neural network combined fast MR image reconstruction: Applications in low-latency accelerated real-time imaging. Med. Phys., 46: 3399-3413.

8. Ghodrati V, Shao J, Bydder M, Zhou Z, Yin W, Nguyen KL, Yang Y, Hu P. MR image reconstruction using deep learning: evaluation of network structure and loss functions. Quant Imaging Med Surg 2019; 9(9):1516-1527.

9. Ghodrati V, Bydder M, Ali F, et al. Retrospective respiratory motion correction in cardiac cine MRI reconstruction using adversarial autoencoder and unsupervised learning. NMR in Biomedicine. 2021; 34:e4433.

10. Ghodrati V, Bydder M, Bedayat A, et al. Temporally aware volumetric generative adversarial network-based MR image reconstruction with simultaneous respiratory motion compensation: Initial feasibility in 3D dynamic cine cardiac MRI. Magn Reson Med. 2021; 86: 2666– 2683. https://doi.org/10.1002/mrm.28912

11. Shao J, Ghodrati V, Nguyen K-L, Hu P. Fast and accurate calculation of myocardial T1 and T2 values using deep learning Bloch equation simulations (DeepBLESS). Magn Reson Med. 2020; 84: 2831– 2845. https://doi.org/10.1002/mrm.28321

12. Ghodrati V, Rivenson Y, Prosper A, et al. Automatic segmentation of peripheral arteries and veins in ferumoxytol-enhanced MR angiography. Magn Reson Med. 2021; 00: 1– 15. doi:10.1002/mrm.29026

13. Bloch F, Hansen WW, Packard M. The nuclear induction experiment. Phys Rev 1946; 70:474-485.

14. Purcell EM, Torrey HC, Pound RV. Resonance absorption by nuclear moments in a solid. Phys Rev 1946; 69:37-38.

15. LAUTERBUR P. Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance. Nature 242, 190–191 (1973).

16. Damadian R. Tumor detection by nuclear magnetic resonance. Science. 1971 Mar 19; 171(3976):1151-3.

17. Grannell PK, Mansfield P. Microscopy in vivo by nuclear magnetic resonance. Phys Med Biol. 1975 May; 20(3):477-82.

18. Mansfield P, Guilfoyle DN, Ordidge RJ, Coupland RE. Measurement of T1 by echo-planar imaging and the construction of computer-generated images. Phys Med Biol. 1986 Feb; 31(2):113-24.

19. Mansfield P, Maudsley AA. Medical imaging by NMR. Br J Radiol. 1977 Mar; 50(591):188-94. doi: 10.1259/0007-1285-50-591-188. PMID: 849520.

20. Edelman RR. The history of MR imaging as seen through the pages of radiology. Radiology. 2014 Nov; 273(2 Suppl):S181-200.

21. Becker E. D. (1993) A brief history of nuclear magnetic resonance. Amdyt. Chem. 65, 295A 302A.

22. Jung Y, Jashnani Y, Kijowski R, Block WF. Consistent non-cartesian off-axis MRI quality: calibrating and removing multiple sources of demodulation phase errors. Magn. Reson. Med. 2007; 57:206–212.

23. Brodsky EK, Samsonov AA, Block WF. Characterizing and correcting gradient errors in non-cartesian imaging: Are gradient errors linear time-invariant (LTI)? Magn. Reson. Med. 2009; 62:1466–76.

24. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 521, 436–444 (2015).

25. Hassoun MH. Fundamentals of Artificial Neural Networks. 1st ed. Cambridge, MA, USA: MIT Press; 1995.

26. Nair V, Hinton GE. Rectified Linear Units Improve Restricted Boltzmann Machines. In: Furnkranz J, Joachims T, eds. ICML. Omnipress; 2010:807-814.

27. Bottou L. Stochastic gradient learning in neural networks. In Proceedings of Neuro-Nimes 91, 1991.

28. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778.

29. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of ICML, pages 448–456, 2015.

30. Sonoda S, Murata N. Neural network with unbounded activation functions is universal approximator. Applied and Computational Harmonic Analysis, 43(2):233–268, 2017.

31. Tsao J, Boesiger P, Pruessmann KP. k-t BLAST and k-t SENSE: dynamic MRI with high frame rate exploiting spatiotemporal correlations. Magn Reson Med. 2003 Nov;50(5):1031-42.

32. Shao J, Liu D, Sung K, Nguyen K-L, Hu P. Accuracy, precision, and reproducibility of myocardial T1 mapping: A comparison of four T1 estimation algorithms for modified look-locker inversion recovery (MOLLI). Magn. Reson. Med. 2017;78.

33. Jung H, Sung K, Nayak KS, Kim EY, Ye JC. k-t FOCUSS: a general compressed sensing framework for high resolution dynamic MRI. Magn Reson Med. 2009 Jan;61(1):103-16.

34. Ravishankar S, Bresler Y. MR Image Reconstruction From Highly Undersampled k-Space Data by Dictionary Learning. IEEE Trans Med Imaging 2011;30:1028-41.

35. Dong C, Loy CC, He K, Tang X. Image Super-Resolution Using Deep Convolutional Networks. IEEE Trans Pattern Anal Mach Intell 2016;38:295-307.

36. Kim J, Lee JK, Lee KM. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Available online: https://ieeexplore.ieee.org/document/7780551/

37. Wang Z, Liu D, Yang J, Han W, Huang T. Deep Networks for Image Super-Resolution with Sparse Prior. Proceedings of IEEE International Conference on Computer Vision (ICCV). Available online: https://ieeexplore.ieee.org/document/7410407

38. Cui Z, Chang H, Shan S, Zhong B, Chen X. Deep Network Cascade for Image Super-resolution. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T. editors. ECCV 2014. Springer Nature, 2014:49-64.

39. Xie J, Xu L, Chen E. Image Denoising and Inpainting with Deep Neural Networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ. editors. Proccedings of Advances in Neural Information Processing Systems-volume 1. 2012:341-9.

40. Jain V, Seung HS. Natural Image Denoising with Convolutional Networks. In: Koller D, Schuurmans D, Bengio Y, Bottou L. editors. Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference. 2008:769-76.

41. Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Trans Image Process 2017; 26:3142-55

42. Chen Y, Pock T. Trainable Nonlinear Reaction Diffusion: A Flexible Framework for Fast and Effective Image Restoration. IEEE Trans Pattern Anal Mach Intell 2017; 39:1256-72.

43. Kyong Hwan Jin. McCann MT, Froustey E, Unser M. Deep Convolutional Neural Network for Inverse Problems in Imaging. IEEE Trans Image Process 2017; 26:4509-22.

44. Chen H, Zhang Y, Kalra MK, Lin F, Chen Y, Liao P, Zhou J, Wang G, Low-Dose CT. With a Residual Encoder-Decoder Convolutional Neural Network. IEEE Trans Med Imaging 2017;36:2524-35.

45. Majumdar A. Real-time Dynamic MRI Reconstruction using Stacked Denoising Autoencoder. arXiv:1503.06383.

46. Sandino CM, Dixit N, Cheng JY, Vasanawala SS. Deep convolutional neural networks for accelerated dynamic magnetic resonance imaging. Available online: http://cs231n.stanford.edu/reports/2017/pdfs/513.pdf

47. Wang S, Su Z, Ying L, et al. ACCELERATING MAGNETIC RESONANCE IMAGING VIA DEEP LEARNING. Proc IEEE Int Symp Biomed Imaging. 2016; 2016:514-517. doi:10.1109/ISBI.2016.7493320.

48. Hammernik K, Klatzer T, Kobler E, Recht MP, Sodickson DK, Pock T, Knoll F. Learning a Variational Network for Reconstruction of Accelerated MRI Data. Magn Reson Med 2018; 79:3055-71.

49. Schlemper J, Caballero J, Hajnal JV, Price AN, Rueckert D. A Deep Cascade of Convolutional Neural Networks for Dynamic MR Image Reconstruction. IEEE Trans Med Imaging 2018; 37:491-503.

50. Hyun CM, Kim HP, Lee SM, Lee S, Seo JK. Deep learning for undersampled MRI reconstruction. arXiv: 1709.02576.

51. Zhu B, Liu JZ, Cauley SF, Rosen BR, Rosen MS. Image reconstruction by domain-transform manifold learning. Nature 2018; 555:487-92.

52. Yang Y, Sun J, Li H, Xu Z. ADMM-CSNet: a deep learning approach for image compressive sensing. IEEE Trans Pattern Anal Mach Intell. 2018;1–1.

53. Zhu J-Y, Krahenbuhl P, Shechtman E, Efros AA. Generative visual manipulation on the natural image manifold. In: Computer Vision – ECCV 2016. Cham, Switzerland: Springer; 2016:597–613

54. Yeh. RA, Chen C, Lim TY, Schwing AG, Hasegawa-Johnson M, Do MN. Semantic image inpainting with deep generative models. https:// arxiv.org/abs/1607.07539; 2016.

55. Sønderby CK, Caballero J, Theis L, Shi W, Huszar F. Amortised MAP inference for image super-resolution. arXiv:1610.04490; 2016.

56. Ledig C, Theis L, Huszar F, et al. Photo-realistic single image super-resolution using a generative adversarial network. arXiv:1609.04802; 2016.

57. Mardani M, Gong E, Cheng JY, et al. Deep generative adversarial neural networks for compressive sensing MRI. IEEE Trans Med Imaging. 2019; 38:167–179.

58. Yang G, Yu S, Dong H, et al. DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction. IEEE Trans Med Imaging. 2018; 37:1310–1321.

59. Dar SUH, Cukur T. A Transfer-Learning Approach for Accelerated MRI using Deep Neural Networks. arXiv: 1710.02615; 2017.

60. Han Y, Yoo J, Kim HH, Shin HJ, Sung K, Ye JC. Deep learning with domain adaptation for accelerated projection-reconstruction MR. Magn Reson Med. 2018; 80:1189–1205.

61. Kressler B, Spincemaille P, Nguyen TD, Cheng L, Xi Hai Z, Prince MR, Wang Y. Three-dimensional cine imaging using variable-density spiral trajectories and SSFP with application to coronary artery angiography. Magnetic resonance in medicine. 2007;58(3):535-543.

62. Wetzl J, Schmidt M, Pontana F, Longère B, Lugauer F, Maier A, Hornegger J, Forman C. Single-breath-hold 3-D CINE imaging of the left ventricle using Cartesian sampling. Magnetic Resonance Materials in Physics, Biology and Medicine 2018;31(1):19-31.

63. Barkauskas KJ, Rajiah P, Ashwath R, Hamilton JI, Chen Y, Ma D, Wright KL, Gulani V, Griswold MA, Seiberlich N. Quantification of left ventricular functional parameter values using 3D spiral bSSFP and through-time non-Cartesian GRAPPA. Journal of Cardiovascular Magnetic Resonance 2014;16:65.

64. T Küstner, A Bustin, R Neji, R Botnar, C Prieto. 3D Cartesian Whole-heart CINE MRI Exploiting Patch-based Spatial and Temporal Redundancies. ESMRMB 2019.

65. Weiger M, Pruessmann KP, Boesiger P. Cardiac Real-Time Imaging Using SENSE. Magn Reson Med. 2000; 43: 177-184.

66. Cui C, Yin G, Lu M, et al. Retrospective Electrocardiography-Gated Real-Time Cardiac Cine MRI at 3T: Comparison with Conventional Segmented Cine MRI. Korean J Radiol. 2019;20(1):114–125.

67. Uecker M, Zhang S, Voit D, Karaus A, Merboldt K and Frahm J. Realtime MRI at a resolution of 20 ms. NMR Biomed. 2010; 23: 986-994.

68. Feng X, Salerno M, Kramer CM, Meyer CH. Non-Cartesian Balanced Steady-State Free Precession Pulse Sequences for Real-Time Cardiac MRI. Magn Reson Med. 2016; 75:1546-1555.

69. Stehning C, Bornert P, Nehrke K, Eggers H, Stuber M. Free-Breathing Whole-Heart Coronary MRA With 3D Radial SSFP and Self-Navigated Image Reconstruction. Magn Reson Med. 2005; 54:476-480.

70. Feng L, Axel L, Chandarana H, Block KT, Sodickson D.K, Otazo R. XD-GRASP: Golden-angle radial MRI with reconstruction of extra motion-state dimensions using compressed sensing. Magn Reson Med. 2016; 75:775–788

71. Yuan Q, Axel L, Hernandez EH, et al. Cardiac-Respiratory Gating Method for Magnetic Resonance Imaging of the Heart. Magn Reson Med. 2000; 43:314-318.

72. Wang Y, Christy PS, Korosec FR, et al. Coronary MRI with a Respiratory Feedback Monitor : The 2D Imaging Case. Magn Reson Med. 1995; 33:116-121.

73. Santelli C, Nezafat R, Goddu B, et al. Respiratory Bellows Revisited for Motion Compensation : Preliminary Experience for Cardiovascular MR. Magn Reson Med. 2011; 65:1097-1102.

74. Ehman RL, Felmlee JP. Adaptive Technique for High-Definition MR Imaging of Moving Structures. Radiology. 1989; 173(1): 255-263.

75. Larson AC, Kellman P, Arai A, et al. Preliminary investigation of respiratory self‐gating for free‐breathing segmented cine MRI. Magn Reson Med. 2005; 53:159-168.

76. Wang Y, Riederer SJ, Ehman RL. Respiratory Motion of the Heart : Kinematics and the Implications for the Spatial Resolution in Coronary Imaging. Magn Reson Med. 1995; 33:713-719.

77. Wang Y, Ehman RL. Retrospective Adaptive Motion Correction for Navigator-Gated 3D Coronary MR Angiography. Magn Reson Imaging. 2000; 11:208-214.

78. Atkinson D, Hill D.LG, Stoyle P.NR, Summers PE, Keevil S. Automatic correction of motion artifacts in magnetic resonance images using an entropy focus criterion. IEEE Trans Med Imaging. 1997; 16(6): 903-910.

79. Cheng JY, Alley MT, Cunningham CH, Vasanawala SS, Pauly JM, Lustig M. Nonrigid Motion Correction in 3D Using Autofocusing With Localized Linear Translations. Magn Reson Med. 2012; 68:1785-1797.

80. Cruz G, Atkinson D, Buerger C, Schaeffter T, Prieto C. Accelerated Motion Corrected Three-Dimensional Abdominal MRI Using Total Variation Regularized SENSE Reconstruction. Magn Reson Med. 2016; 75:1484-1498.

81. Chen H, Zhang Y, Kalra MK, et al. Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network. IEEE Trans Med Imaging. 2017; 36(12):2524-2535.

82. Xu L, Ren JS, Liu C, Jia J. Deep Convolutional Neural Network for Image Deconvolution. In: Proccedings of Advances in Neural Information Processing Systems (NIPS). Montreal, Canada: Curran Associates, Inc; 2014: 1790–1798.

83. Nishio M, Nagashima C, Hirabayashi S, et al. Convolutional auto-encoder for image denoising of ultra-low-dose CT. Heliyon. 2017; 3(8):e00393.

84. Küstner T, Schick F, et al. Retrospective correction of motion - affected MR images using deep learning frameworks. Magn Reson Med. 2019; 82: 1527–1540.

85. Pawar K, Chen Z, Shah NJ, Egan GF. Motion Correction in MRI using Deep Convolutional Neural. arXive: arXive: 1807.10831, preprint, 2018.

86. Duffy B.A, Zhang W, Tang H, Zhao L, Law M, Toga A.W, Kim H. Retrospective correction of motion artifact affected structural MRI images using deep learning of simulated motion. In: Medical Imaging with Deep Learning (MIDL). Amsterdam, Netherlands; 2018.

87. Lv J, Yang M, Zhang J, Wang X. Respiratory motion correction for free-breathing 3D abdominal MRI using CNN-based image registration: a feasibility study. Br J Radiol. 2018; 91(1083):20170788.

88. Haskell MW, Cauley SF, Bilgic B, et al. Network Accelerated Motion Estimation and Reduction (NAMER): Convolutional neural network guided retrospective motion correction using a separable motion model. Magn Reson Med. 2019; 82: 1452– 1461.

89. Johnson J, Alahi A, Li F-F. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. arXive: arXive: 1603.08155, preprint, 2016.

90. Armanious K, Tanwar A, Abdulatif S, Kustner T, Gatidis S, Yang B. Unsupervised Adversarial Correction of Rigid MR Motion Artifacts. arXive: arXive: 1910.05597, preprint, 2019.

91. Makhzani A. Implicit Autoencoders. arXive: arXive: 1805.09804, preprint, 2019.

92. Makhzani A, Frey B, Goodfellow I. Adversarial Autoencoders. arXive: arXive:1511.05644, preprint, 2016.

93. Zhu J-Y, Park T, Isola P, Efros AA. Unpaired image-to-image translation using cycle-consistent adversarial networks. arXive: arXive: 1703.10593, preprint, 2016.

94. Goodfellow I, Bengio Y, Courville A, Bengio Y. Autoencoders. Deep Learning. Vol 1. MIT press Cambridge; 2016:493-515.

95. Odena A, Olah C, Shlens J. Conditional Image Synthesis with Auxiliary Classifier GANs. arXive: arXive: 1610.09585, preprint, 2017.

96. Karras T, Aila T, Laine S, Lehtinen J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. arXive: arXive: 1710.10196, preprint, 2018.

97. Radford A, Metz L, Chintala S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXive: arXive: 1511.06434, preprint, 2016.

98. Li C, Wand M. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. arXive: arXive: 1604.04382, preprint, 2016.

99. Xanthis CG, Venetis IE, Aletras AH. High performance MRI simulations of motion on multi-GPU systems. J Cardiovasc Magn Reson 16, 48 (2014).

100. Krotkov E. Focusing. Int J Comput Vision 1, 223–237 (1988).

101. Mir H, Xu P, Van Beek P, "An extensive empirical evaluation of focus measures for digital photography," Proc. SPIE 9023, Digital Photography X, 90230I (7 March 2014).

102. Nguyen KL, Khan SN, Moriarty JM, et al. High-field MR imaging in pediatric congenital heart disease: initial results. Pediatr Radiol. 2015; 45(1):42–54.

103. Gwet KL. Computing inter-rater reliability and its variance in the presence of high agreement. British Journal of Mathematical and Statistical Psychology. 2008; 61(1) 29-48.

104. Forthover RN, Lee SL, Hernandez M. Biostatistics: A Guide to Design, Analysis, and Discovery. 2nd ed. San Diego: Academic Press, 2007: 249-68.

105. Zaitsev M, Maclaren J, Herbst M. Motion artifacts in MRI: A complex problem with many partial solutions. J Magn Reson Imaging. 2015; 42(4):887–901.

106. Bydder M, Larkman D, Hajnal J. (2002), Detection and elimination of motion artifacts by regeneration of k -space. Magn. Reson. Med., 47: 677-686.

107. Samsonov AA, Velikina J, Jung Y, Kholmovski EG, Johnson CR, Block WF. POCS-enhanced correction of motion artifacts in parallel MRI. Magn Reson Med. 2010; 63(4):1104-1110.

108. Earls JP, Ho VB, Foo TK, Castillo E, Flamm SD. Cardiac MRI: Recent progress and continued challenges. J Magn Reson Imaging. 2002;16(2):111-127. doi:10.1002/jmri.10154

109. Yang RK, Roth CG, Ward RJ, deJesus JO, Mitchell DG. Optimizing Abdominal MR Imaging: Approaches to Common Problems. RadioGraphics. 2010; 30(1):185-199. doi:10.1148/rg.301095076

110. Zand KR, Reinhold C, Haider MA, Nakai A, Rohoman L, Maheshwari S. Artifacts and pitfalls in MR imaging of the pelvis. J Magn Reson Imaging. 2007; 26(3):480-497. doi:10.1002/jmri.20996

111. Deshmane A, Gulani V, Griswold MA, Seiberlich N. Parallel MR imaging. J Magn Reson Imaging. 2012; 36(1):55-72. doi:10.1002/jmri.23639

112. Feng L, Axel L, Chandarana H, Block KT, Sodickson DK, Otazo R. XD-GRASP: Golden-angle radial MRI with reconstruction of extra motion-state dimensions using compressed sensing. Magn Reson Med. 2016; 75(2):775-788. doi:10.1002/mrm.25665

113. Tariq U, Hsiao A, Alley M, Zhang T, Lustig M, Vasanawala SS. Venous and arterial flow quantification are equally accurate and precise with parallel imaging compressed sensing 4D phase contrast MRI. J Magn Reson Imaging. 2013; 37(6):1419-1426. doi:10.1002/jmri.23936

114. Zhang T, Chowdhury S, Lustig M, et al. Clinical performance of contrast enhanced abdominal pediatric MRI with fast combined parallel imaging compressed sensing reconstruction. J Magn Reson Imaging. 2014; 40(1):13-25. doi:10.1002/jmri.24333

115. Zucker EJ, Cheng JY, Haldipur A, Carl M, Vasanawala SS. Free-breathing pediatric chest MRI: Performance of self-navigated golden-angle ordered conical ultrashort echo time acquisition. J Magn Reson Imaging. 2018; 47(1):200-209. doi:10.1002/jmri.25776

116. Han F, Zhou Z, Han E, et al. Self‐gated 4D multiphase, steady‐state imaging with contrast enhancement (MUSIC) using rotating cartesian K‐space (ROCK): Validation in children with congenital heart disease. Magn Reson Med. 2017; 78(2):472-483

117. Cheng JY, Zhang T, Ruangwattanapaisarn N, et al. Free-breathing pediatric MRI with nonrigid motion correction and acceleration. J Magn Reson Imaging. 2015; 42(2):407-420. doi:10.1002/jmri.24785

118. Forman C, Piccini D, Grimm R, Hutter J, Hornegger J, Zenge MO. Reduction of respiratory motion artifacts for free-breathing whole-heart coronary MRA by weighted iterative reconstruction. Magn Reson Med. 2015; 73(5):1885-1895. doi:10.1002/mrm.25321

119. Jiang W, Ong F, Johnson KM, et al. Motion robust high resolution 3D free-breathing pulmonary MRI using dynamic 3D image self-navigator. Magn Reson Med. 2018; 79(6):2954-2967. doi:10.1002/mrm.26958

120. Johnson KM, Block WF, Reeder SB, Samsonov A. Improved least squares MR image reconstruction using estimates of k-space data consistency. Magn Reson Med. 2012;67(6):1600-1608. doi:10.1002/mrm.23144

121. Taylor AM, Keegan J, Jhooti P, Firmin DN, Pennell DJ. Calculation of a subject-specific adaptive motion-correction factor for improved real-time navigator echo-gated magnetic

resonance coronary angiography. J Cardiovasc Magn Reson. 1999; 1(2):131-138. doi:10.3109/10976649909080841

122. Zhang T, Cheng JY, Potnick AG, et al. Fast pediatric 3D free-breathing abdominal dynamic contrast enhanced MRI with high spatiotemporal resolution. J Magn Reson Imaging. 2015; 41(2):460-473. doi:10.1002/jmri.24551

123. Christodoulou AG, Shaw JL, Nguyen C, et al. Magnetic resonance multitasking for motion-resolved quantitative cardiovascular imaging. Nat Biomed Eng. 2018; 2(4):215-226. doi:10.1038/s41551-018-0217-y

124. Sandino CM, Lai P, Vasanawala SS, Cheng JY. Accelerating cardiac cine MRI using a deep learning-based ESPIRiT reconstruction. Magn Reson Med. 2020; 00:1-16. doi:10.1002/mrm.28420

125. Fuin N, Bustin A, Küstner T, et al. A multi-scale variational neural network for accelerating motion-compensated whole-heart 3D coronary MR angiography. Magn Reson Imaging. 2020; 70:155-167. doi:https://doi.org/10.1016/j.mri.2020.04.007

126. Kofler A, Dewey M, Schaeffter T, Wald C, Kolbitsch C. Spatio-Temporal Deep Learning-Based Undersampling Artefact Reduction for 2D Radial Cine MRI With Limited Training Data. IEEE Trans Med Imaging. 2020;39(3):703-717. doi:10.1109/TMI.2019.2930318

127. Han Y, Sunwoo L, Ye JC. k -Space Deep Learning for Accelerated MRI. IEEE Trans Med Imaging. 2020; 39(2):377-386. doi:10.1109/TMI.2019.2927101

128. Knoll F, Hammernik K, Zhang C, et al. Deep-Learning Methods for Parallel Magnetic Resonance Imaging Reconstruction: A Survey of the Current Approaches, Trends, and Issues. IEEE Signal Process Mag. 2020; 37(1):128-140. doi:10.1109/MSP.2019.2950640

129. Wang S, Cheng H, Ying L, et al. DeepcomplexMRI: Exploiting deep residual network for fast parallel MR imaging with complex convolution. Magn Reson Imaging. 2020;68:136-147. doi:https://doi.org/10.1016/j.mri.2020.02.002

130. Chen F, Cheng JY, Taviani V, et al. Data-driven self-calibration and reconstruction for non-cartesian wave-encoded single-shot fast spin echo using deep learning. J Magn Reson Imaging. 2020; 51(3):841-853. doi:10.1002/jmri.26871

131. Hammernik K, Knoll F. Chapter 2 - Machine learning for image reconstruction. In: Zhou SK, Rueckert D, Fichtinger GBT-H of MIC and CAI, eds. Academic Press; 2020:25-64. doi:https://doi.org/10.1016/B978-0-12-816176-0.00007-7

132. Sun L, Fan Z, Fu X, Huang Y, Ding X, Paisley J. A Deep Information Sharing Network for Multi-Contrast Compressed Sensing MRI Reconstruction. IEEE Trans Image Process. 2019;28(12):6141-6153. doi:10.1109/TIP.2019.2925288.

133. Akçakaya M, Moeller S, Weingärtner S, Uğurbil K. Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: Database-free deep learning for fast imaging. Magn Reson Med. 2019; 81(1):439-453. doi:10.1002/mrm.27420.

134. Hauptmann A, Arridge S, Lucka F, Muthurangu V, Steeden JA. Real-time cardiovascular MR with spatio-temporal artifact suppression using deep learning-proof of concept in congenital heart disease. Magn Reson Med. 2019; 81(2):1143-1156.

135. Aggarwal HK, Mani MP, Jacob M. MoDL: Model-Based Deep Learning Architecture for Inverse Problems. IEEE Trans Med Imaging. 2019; 38(2):394-405.

136. Qin C, Schlemper J, Caballero J, Price AN, Hajnal J V, Rueckert D. Convolutional Recurrent Neural Networks for Dynamic MR Image Reconstruction. IEEE Trans Med Imaging. 2019; 38(1):280-290. doi:10.1109/TMI.2018.2863670.

137. Chen F, Taviani V, Malkiel I, et al. Variable-Density Single-Shot Fast Spin-Echo MRI with Deep Learning Reconstruction by Using Variational Networks. Radiology. 2018; 289(2):366-373. doi:10.1148/radiol.2018180445.

138. Eo T, Jun Y, Kim T, Jang J, Lee H-J, Hwang D. KIKI-net: cross-domain convolutional neural networks for reconstructing undersampled magnetic resonance images. Magn Reson Med. 2018; 80(5):2188-2201. doi:10.1002/mrm.27201.

139. Lee D, Yoo J, Tak S, Ye JC. Deep Residual Learning for Accelerated MRI Using Magnitude and Phase Networks. IEEE Trans Biomed Eng. 2018; 65(9):1985-1995.

140. Quan TM, Nguyen-Duc T, Jeong W. Compressed Sensing MRI Reconstruction Using a Generative Adversarial Network with a Cyclic Loss. IEEE Trans Med Imaging. 2018; 37(6):1488-1497. doi:10.1109/TMI.2018.2820120.

141. Yan Y, Sun J, Li H, Xu Z. Deep ADMM-Net for Compressive Sensing MRI. In: Lee DD, Sugiyama M, Luxburg U V, Guyon I, Garnett R, eds. Advances in Neural Information Processing Systems 29. Curran Associates, Inc.; 2016:10-18.

142. Küstner T, Fuin N, Hammernik K, et al. CINENet: deep learning-based 3D cardiac CINE MRI reconstruction with multi-coil complex-valued 4D spatio-temporal convolutions. Sci Rep. 2020;10(1):13710. doi:10.1038/s41598-020-70551-8.

143. Tamada D, Kromrey ML, Ichikawa S, Onishi H, Motosugi U. Motion Artifact Reduction Using a Convolutional Neural Network for Dynamic Contrast Enhanced MR Imaging of the Liver. Magn Reson Med Sci. 2020; 19(1):64-76. doi:10.2463/mrms.mp.2018-0156.

144. Lv J, Yang M, Zhang J, Wang X. Respiratory motion correction for free-breathing 3D abdominal MRI using CNN-based image registration: a feasibility study. Br J Radiol. 2018; 91(1083):20170788. doi:10.1259/bjr.20170788.

145. Zhao H, Gallo O, Frosio I, Kautz J. Loss Functions for Image Restoration With Neural Networks. IEEE Trans Comput Imaging. 2017; 3(1):47-57. doi:10.1109/TCI.2016.2644865.

146. Menchón-Lara RM, Simmross-Wattenberg F, Casaseca-de-la-Higuera P, Martín-Fernández M, Alberola-López C. Reconstruction techniques for cardiac cine MRI. Insights Imaging. 2019; 10(1):100. doi:10.1186/s13244-019-0754-2.

147. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Nets. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, eds. Advances in Neural Information Processing Systems 27. Curran Associates, Inc.; 2014:2672-2680. http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

148. Arjovsky M, Chintala S, Bottou L. Wasserstein GAN. arXiv: arXiv: 1701.07875, preprint, 2017.

149. Mao X, Li Q, Xie H, Lau RYK, Wang Z, Smolley SP. Least Squares Generative Adversarial Networks. arXiv: arXiv: 1611.04076, preprint, 2016.

150. Van Belle G, Fisher LD, Heagerty PJ. and Lumley T. Multiple Comparisons. In: W. A. Shewhart, S. S. Wilks, G. Van Belle, L. D. Fisher PJH and TL, ed. Biostatistics. Second. John Wiley & Sons, Ltd; 2004:520-549. doi:10.1002/0471602396.ch12.

151. Knoll F, Murrell T, Sriram A, et al. Advancing machine learning for MR image reconstruction with an open competition: Overview of the 2019 fastMRI challenge. Magn Reson Med. 2020; 84: 3054– 3070. https://doi.org/10.1002/mrm.28338

152. Messroghli DR, Moon JC, Ferreira VM, et al. Clinical recommendations for cardiovascular magnetic resonance mapping of T1, T2, T2* and extracellular volume: a consensus statement by the society for cardiovascular magnetic resonance (SCMR) endorsed by the European association for cardiovascular Imaging (EACVI). J Cardiovasc Magn Reson. 2017;19:75.

153. Kim PK, Hong YJ, Im DJ, et al. Myocardial T1 and T2 mapping: techniques and clinical applications. Korean J Radiol. 2017;18:113-131.

154. Bohnen S, Radunski UK, Lund GK, et al. Performance of T1 and T2 mapping cardiovascular magnetic resonance to detect ctive myocarditis in patients with recent-onset heart failure. Circ Cardiovasc Imaging. 2015;8:e003073.

155. Messroghli DR, Radjenovic A, Kozerke S, Higgins DM, Sivananthan MU, Ridgway JP. Modified look-locker inversion recovery (MOLLI) for high-resolution T1 mapping of the heart. Magn Reson Med. 2004;52:141-146.

156. Huang T-Y, Liu Y-J, Stemmer A, Poncelet BP. T2 measurement of the human myocardium using a T2-prepared transient-state true-FISP sequence. Magn Reson Med. 2007;57:960-966.

157. Karamitsos TD, Piechnik SK, Banypersad SM, et al. Noncontrast T1 mapping for the diagnosis of cardiac amyloidosis. JACC Cardiovasc Imaging. 2013;6:488-497.

158. Giri S, Chung Y-C, Merchant A, et al. T2 quantification for im-proved detection of myocardial edema. J Cardiovasc Magn Reson. 2009;11:56.

159. Kellman P, Hansen MS. T1-mapping in the heart: Accuracy and precision. J Cardiovasc Magn Reson. 2014;16:2.

160. Kvernby S, Warntjes MJB, Haraldsson H, Carlhäll C-J, Engvall J, Ebbers T. Simultaneous three-dimensional myocardial T1 and T2 mapping in one breath hold with 3D-QALAS. J Cardiovasc Magn Reson. 2014;16:102.

161. Chow K, Flewitt JA, Green JD, Pagano JJ, Friedrich MG, Thompson RB. Saturation recovery single-shot acquisition (SASHA) for myo-cardial T(1) mapping. Magn Reson Med. 2014; 71:2082-2095.

162. Shao J, Zhou Z, Nguyen KL, Finn JP, Hu P. Accurate, precise, simultaneous myocardial T1 and T2 mapping using a radial se-quence with inversion recovery and T2 preparation. NMR Biomed. 2019;32:e4165. https://doi.org/10.1002/nbm.4165.

163. Xanthis CG, Bidhult S, Kantasis G, Heiberg E, Arheden H, Aletras AH. Parallel simulations for QUAntifying RElaxation magnetic resonance constants (SQUAREMR): An example towards ac-curate MOLLI T1 measurements. J Cardiovasc Magn Reson. 2015;17:104.

164. Hamilton JI, Jiang Y, Chen Y, et al. MR fingerprinting for rapid quantification of myocardial T1, T2, and proton spin density. Magn Reson Med. 2017;77:1446-1458.

165. Hamilton JI, Jiang Y, Ma D, et al. Investigating and reducing the effects of confounding factors for robust T1 and T2 map-ping with cardiac MR fingerprinting. Magn Reson Imaging. 2018;53:40-51.

166. Cruz G, Jaubert O, Botnar RM, Prieto C. Cardiac magnetic reso-nance fingerprinting: Technical developments and initial clinical validation. Curr Cardiol Rep. 2019;21:91.

167. Ma D, Gulani V, Seiberlich N, et al. Magnetic resonance finger-printing. Nature. 2013;495:187-192.

168. Cohen O, Zhu B, Rosen MS. MR fingerprinting Deep RecOnstruction NEtwork (DRONE). Magn Reson Med. 2018;80:885-894.

169. Hoppe E, Körzdörfer G, Würfl T, et al. Deep learning for magnetic resonance fingerprinting: A new approach for predicting quanti-tative parameter values from time series. Stud Health Technol Inform. 2017;243:202-206.

170. Hamilton JI, Seiberlich N. Machine learning for rapid magnetic resonance fingerprinting tissue property quantification. Proc IEEE. 2019;108:69-85.

171. Shao J, Rapacchi S, Nguyen K-L, Hu P. Myocardial T1 mapping at 3.0 tesla using an inversion recovery spoiled gradient echo readout and Bloch equation simulation with slice profile correc-tion (BLESSPC) T1 estimation algorithm. J Magn Reson Imaging. 2016;43:414-425.

172. Shao J, Rashid S, Renella P, Nguyen KL, Hu P. Myocardial T1 mapping for patients with implanted cardiac devices using wide-band inversion recovery spoiled gradient echo readout. Magn Reson Med. 2017;77:1495-1504.

173. Xue H, Greiser A, Zuehlsdorff S, et al. Phase-sensitive inversion recovery for myocardial T1 mapping with motion correction and parametric fitting. Magn Reson Med. 2013; 69:1408-1420.

174. Kellman P, Wilson JR, Xue H, Ugander M, Arai AE. Extracellular volume fraction mapping in the myocardium. I: Evaluation of an automated method. J Cardiovasc Magn Reson. 2012;14:63.

175. Kellman P, Herzka DA, Hansen MS. Adiabatic inversion pulses for myocardial T1 mapping. Magn Reson Med. 2014;71:1428-1434.

176. Rodgers CT, Piechnik SK, Delabarre LJ, et al. Inversion recovery at 7 T in the human myocardium: Measurement of T(1), inversion efficiency and B1+. Magn Reson Med. 2013;70:1038-1046.

177. Blume U, Lockie T, Stehning C, et al. Interleaved T1 and T2 re-laxation time mapping for cardiac applications. J Magn Reson Imaging. 2009;29:480-487.

178. Santini F, Kawel-Boehm N, Greiser A, Bremerich J, Bieri O. Simultaneous T1 and T2 quantification of the myocardium using cardiac balanced-SSFP inversion recovery with interleaved sam-pling acquisition (CABIRIA). Magn Reson Med. 2015;74:365-371.

179. Akçakaya M, Weingärtner S, Basha TA, Roujol S, Bellm S, Nezafat R. Joint myocardial T1 and T2 mapping using a combina-tion of saturation recovery and T2 -preparation. Magn Reson Med. 2016;76:888-896.

180. Xanthis CG, Bidhult S, Greiser A, et al. Simulation-based quanti-fication of native T1 and T2 of the myocardium using a modified MOLLI scheme and the importance of Magnetization Transfer. Magn Reson Imaging. 2018;48:96-106.

181. Christodoulou AG, Shaw JL, Nguyen C, et al. Magnetic resonance multitasking for motion-resolved quantitative cardiovascular imaging. Nat Biomed Eng. 2018;2:215-226.

182. Liu Y, Hamilton J, Rajagopalan S, Seiberlich N. Cardiac magnetic resonance fingerprinting: Technical overview and initial results. JACC Cardiovasc Imaging. 2018; 11:1837-1853.

183. Eo T, Jun Y, Kim T, Jang J, Lee H, Hwang D. KIKI -net: cross-domain convolutional neural networks for reconstructing undersampled magnetic resonance images. Magn Reson Med. 2018;80:2188-2201.

184. Robson MD, Piechnik SK, Tunnicliffe EM, Neubauer S. T1 measurements in the human myocardium: The effects of magnetization transfer on the SASHA and MOLLI sequences. Magn Reson Med. 2013;70:664-670.

185. Dabir D, Child N, Kalra A, et al. Reference values for healthy human myocardium using a T1 mapping methodology: Results from the International T1 multicenter cardiovascular magnetic res-onance study. J Cardiovasc Magn Reson. 2014;16:69.

186. Versluis B, Backes WH, van Eupen MG, et al. Magnetic resonance imaging in peripheral arterial disease: reproducibility of the assessment of morphological and functional vascular status. Invest Radiol. 2011 Jan;46(1):11-24.

187. Pollak AW, Kramer CM. MRI in Lower Extremity Peripheral Arterial Disease: Recent Advancements. Curr Cardiovasc Imaging Rep. 2013;6(1):55-60.

188. Pollak AW, Norton PT, Kramer CM. Multimodality imaging of lower extremity peripheral arterial disease: current role and future directions. Circ Cardiovasc Imaging. 2012;5(6):797-807.

189. Mathew RC, Kramer CM. Recent advances in magnetic resonance imaging for peripheral artery disease. Vasc Med. 2018 Apr;23(2):143-152.

190. Nielsen YW, Thomsen HS. Contrast-enhanced peripheral MRA: technique and contrast agents. Acta Radiol. 2012 Sep 1;53(7):769-777.

191. Rofsky NM, Adelman MA. MR angiography in the evaluation of atherosclerotic peripheral vascular disease. Radiology. 2000 Feb;214(2):325-338.

192. Leiner T, Carr JC. Noninvasive Angiography of Peripheral Arteries. 2019 Feb 20. In: Hodler J, Kubik-Huch RA, von Schulthess GK, editors. Diseases of the Chest, Breast, Heart and Vessels 2019-2022: Diagnostic and Interventional Imaging [Internet]. Cham (CH): Springer; 2019. Chapter 20. Available from: https://www.ncbi.nlm.nih.gov/books/NBK553864/ doi: 10.1007/978-3-030-11149-6_20

193. Finn JP, Nguyen KL, Han F, Zhou Z, Salusky I, Ayad I, Hu P. Cardiovascular MRI with ferumoxytol. Clin Radiol. 2016 Aug; 71(8):796-806.

194. Lei T, Udupa JK, Saha PK, Odhner D. Artery-vein separation via MRA--an image processing approach. IEEE Trans Med Imaging. 2001 Aug;20(8):689-703.

195. Shahzad R, Dzyubachyk O, Staring M, et al. Automated extraction and labelling of the arterial tree from whole-body MRA data. Med Image Anal. 2015;24(1):28-40.

196. Lesage D, Angelini ED, Bloch I, Funka-Lea G. A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. Med Image Anal. 2009;13(6):819-845.

197. Kirbas C, Quek FKH. A Review of Vessel Extraction Techniques and Algorithms. ACM Comput Surv. 2000;36:81-121.

198. Klein AK, Lee F, Amini AA. Quantitative coronary angiography with deformable spline models. IEEE Trans Med Imaging. 1997;16(5):468-482.

199. Forkert ND, Schmidt-Richberg A, Fiehler J, et al. 3D cerebrovascular segmentation combining fuzzy vessel enhancement and level-sets with anisotropic energy weights. Magn Reson Imaging. 2013;31(2):262-271.

200. Law MWK, Chung ACS. Three Dimensional Curvilinear Structure Detection Using Optimally Oriented Flux BT - Computer Vision – ECCV 2008. In: Forsyth D, Torr P, Zisserman A, eds. Berlin, Heidelberg: Springer Berlin Heidelberg; 2008:368-382.

201 Frangi AF, Niessen WJ, Vincken KL, Viergever MA. Multiscale vessel enhancement filtering BT - Medical Image Computing and Computer-Assisted Intervention — MICCAI'98. In: Wells WM, Colchester A, Delp S, eds. Berlin, Heidelberg: Springer Berlin Heidelberg; 1998:130-137.

202. Schneider M, Hirsch S, Weber B, Székely G, Menze BH. Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters. Med Image Anal. 2015;19(1):220-249.

203. Merkow J, Marsden A, Kriegman D, Tu Z. Dense Volume-to-Volume Vascular Boundary Detection BT - Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016. In: Ourselin S, Joskowicz L, Sabuncu MR, Unal G, Wells W, eds. Cham: Springer International Publishing; 2016.

204. Soomro TA, Afifi AJ, Zheng L, et al. Deep Learning Models for Retinal Blood Vessels Segmentation: A Review. IEEE Access. 2019;7:71696-71717.

205. Fu H, Xu Y, Lin S, Kee Wong DW, Liu J. DeepVessel: Retinal Vessel Segmentation via Deep Learning and Conditional Random Field BT - Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016. In: Ourselin S, Joskowicz L, Sabuncu MR, Unal G, Wells W, eds. Cham: Springer International Publishing; 2016:132-139.

206. Kitrungrotsakul T, Han X-H, Iwamoto Y, Foruzan AH, Lin L, Chen Y-W. Robust hepatic vessel segmentation using multi deep convolution network. In: Proc.SPIE. Vol 10137. ; 2017. https://doi.org/10.1117/12.2253811.

207. Phellan R, Peixinho A, Falcão A, Forkert ND. Vascular Segmentation in TOF MRA Images of the Brain Using a Deep Convolutional Neural Network BT - Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis. In: Cardoso MJ, Arbel T, Lee S-L, et al., eds. Cham: Springer International Publishing; 2017:39-46.

208. Maninis KK, Pont-Tuset J, Arbeláez P,Van Gool L. (2016) Deep Retinal Image Understanding BT - Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016. In: Ourselin S, Joskowicz L, Sabuncu MR, Unal G, Wells W, eds. Cham: Springer International Publishing; 2016, pp. 140–148.

209. Hesamian MH, Jia W, He X, Kennedy P. Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges. J Digit Imaging. 2019;32(4):582-596.

210. Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation BT - Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016. In: Ourselin S, Joskowicz L, Sabuncu MR, Unal G, Wells W, eds. Cham: Springer International Publishing; 2016:424-432.

211. Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation BT- Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. In: Navab N, Hornegger J, Wells WM, Frangi AF, eds. Cham: Springer International Publishing; 2015:234-241.

212. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. ; 2015:3431-3440.

213. Milletari F, Navab N, Ahmadi S. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV). ; 2016:565-571. doi:10.1109/3DV.2016.79

214. Chen H, Dou Q, Yu L, Qin J, Heng P-A. VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. Neuroimage. 2018;170:446-455.

215. Dou Q, Yu L, Chen H, et al. 3D deeply supervised network for automated segmentation of volumetric medical images. Med Image Anal. 2017;41:40-54.

216. Schlemper J, Oktay O, Schaap M, et al. Attention gated networks: Learning to leverage salient regions in medical images. Med Image Anal. 2019;53:197-207.

217. Todorov MI, Paetzold JC, Schoppe O, et al. Machine learning analysis of whole mouse brain vasculature. Nat Methods. 2020;17(4):442-449.

218. Sanchesa P, Meyer C, Vigon V, Naegel B. Cerebrovascular Network Segmentation of MRA Images With Deep Learning. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019). ; 2019:768-771. doi:10.1109/ISBI.2019.8759569

219. Wang Y, Yan G, Zhu H, et al. VC-Net: Deep Volume-Composition Networks for Segmentation and Visualization of Highly Sparse and Noisy Image Data. IEEE Trans Vis Comput Graph. 2020:1. doi:10.1109/TVCG.2020.3030374

220. Tetteh G, Efremov V, Forkert ND, et al. DeepVesselNet: Vessel Segmentation, Centerline Prediction, and Bifurcation Detection in 3-D Angiographic Volumes. arXiv:1803.09340 (2018).

221. Abraham N, Khan NM. A Novel Focal Tversky Loss Function With Improved Attention U-Net for Lesion Segmentation. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019). ; 2019:683-687. doi:10.1109/ISBI.2019.8759329

222. Zhao S, Wang Y, Yang Z, Cai D. Region Mutual Information Loss for Semantic Segmentation. In: Wallach H, Larochelle H, Beygelzimer A, d\textquotesingle Alché-Buc F, Fox E, Garnett R, eds. Advances in Neural Information Processing Systems. Vol 32. Curran Associates, Inc.; 2019:11117-11127.

223. Lee C-Y, Xie S, Gallagher P, Zhang Z, Tu Z. Deeply-supervised nets. In: Artificial Intelligence and Statistics. 2015:562-570.

224. Lin T-Y, Goyal P, Girshick R, He K, Dollar P. Focal Loss for Dense Object Detection. In: The IEEE International Conference on Computer Vision (ICCV). 2017.

225. Isensee F, Kickingereder P, Wick W, Bendszus M, Maier-Hein KH. Brain Tumor Segmentation and Radiomics Survival Prediction: Contribution to the BRATS 2017 Challenge. In: Crimi A, Bakas S, Kuijf H, Menze B, Reyes M, eds. Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Cham: Springer International Publishing; 2018:287-297.

226. Salehi SSM, Erdogmus D, Gholipour A. Tversky Loss Function for Image Segmentation Using 3D Fully Convolutional Deep Networks BT - Machine Learning in Medical Imaging. In: Wang Q, Shi Y, Suk H-I, Suzuki K, eds. Cham: Springer International Publishing; 2017:379-387.

227. Jerman T, Pernuš F, Likar B, Špiclin Ž. Enhancement of Vascular Structures in 3D and 2D Angiographic Images. IEEE Trans Med Imaging. 2016;35(9):2107-2118.

228. Ersoy H, Rybicki FJ. MR Angiography of the Lower Extremities. Am J Roentgenol. 2008;190(6):1675-1684.

229. Miyazaki M, Takai H, Sugiura S, Wada H, Kuwahara R, Urata J. Peripheral MR Angiography: Separation of Arteries from Veins with Flow-spoiled Gradient Pulses in Electrocardiography-triggered Three-dimensional Half-Fourier Fast Spin-Echo Imaging. Radiology. 2003;227(3):890-896.

230. Wang Y, Yu Y, Li D, et al. Artery and vein separation using susceptibility-dependent phase in contrast-enhanced MRA. J Magn Reson Imaging. 2000;12(5):661-670.

231. Finn JP, Nguyen KL, Hu P. Ferumoxytol vs. Gadolinium agents for contrast-enhanced MRI: Thoughts on evolving indications, risks, and benefits. J Magn Reson Imaging. 2017 Sep;46(3):919-923. doi: 10.1002/jmri.25580. Epub 2017 Feb 3. PMID: 28160356.

232. Toth GB, Varallyay CG, Horvath A, et al. Current and potential imaging applications of ferumoxytol for magnetic resonance imaging. Kidney Int. 2017 Jul;92(1):47-66. doi: 10.1016/j.kint.2016.12.037. Epub 2017 Apr 20. PMID: 28434822; PMCID: PMC5505659.

233. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process. 2004 Apr;13(4):600-12. doi: 10.1109/tip.2003.819861. PMID: 15376593.

234. Zhou Z, Han F, Yoshida T, Nguyen KL, Finn JP, Hu P. Improved 4D cardiac functional assessment for pediatric patients using motion-weighted image reconstruction. MAGMA. 2018 Dec;31(6):747-756. doi: 10.1007/s10334-018-0694-8. Epub 2018 Jul 24. PMID: 30043124.

235. Lei T, Udupa JK, Saha PK, Odhner D. Artery-vein separation via MRA--an image processing approach. IEEE Trans Med Imaging. 2001 Aug;20(8):689-703.