EPFL

Thèse n° 9693

Combining Model Driven and Data Driven Approaches for Inverse Problems in Parameter Estimation and Image Reconstruction: From Modelling to Validation

Présentée le 20 mai 2022

Faculté des sciences et techniques de l'ingénieur
Laboratoire de traitement des signaux 5
Programme doctoral en génie électrique

pour l'obtention du grade de Docteur ès Sciences

par

Thomas YU

Acceptée sur proposition du jury

Prof. D. N. A. Van De Ville, président du jury
Prof. J.-Ph. Thiran, Dr M. Bach Cuadra, directeurs de thèse
Prof. F. Knoll, rapporteur
Prof. A. Chaudhari, rapporteur
Prof. E. Konukoglu, rapporteur

École polytechnique fédérale de Lausanne

2022

Acknowledgements

The journey culminating in the production of this document was not an individual endeavor; it required the help of many individuals, whom I will thank here. As this acknowledgement was prepared in a rush, if you are left out and feel you deserve acknowledgement, please contact me so we can negotiate a settlement. I would like to begin by acknowledging my supervisors, Professor Jean-Philippe Thiran and Dr. Meritxell Bach Cuadra. Throughout my PhD, during which I changed subjects/directions many times, they continually supported me and allowed me the opportunity to pursue my interests. I am so fortunate to have had them both as supervisors. In addition, I would like to thank Anne De Witte for all her help (administrative and otherwise) throughout my PhD, from helping me get my first paychecks when EPFL refused to send money to a German bank account to helping with arranging travel and conferences.

I would like to thank the members of my jury (Professors Akshay Chaudhari, Florian Knoll, Ender Konukoglu, and Dimitri Van De Ville) for their careful evaluation of my thesis as well as the interesting (but tough!) discussion and questioning during my private defense.

While students in general are lucky to have one good mentor, during my PhD, I would say I had three amazing mentors, who taught me so much about science, writing papers, and the business of academia: Dr. Marco, Dr. Erick, and Dr. Tom. Marco, I will always remember almost killing myself by eating so much of your mother's delicious cooking in Verona. Thank you for teaching me in Venice that conferences are for fun and meeting people, not just reading posters. Erick, you are a fountain of amazing research ideas; thank you for collaborating with me and introducing me to cool new projects. Tom, you are the only person who has ever sent me an email asking if I was still alive; thank you for teaching me so much about MR physics.

I was also lucky enough to be a part of three laboratories/groups during my PhD: LTS5, MIAL, and Siemens Healthineers. In addition to their help with scientific endeavors, we also (more importantly) had a lot of fun. To the doctors of LTS5: Dr. Francesco, always remember that pineapple pizza is the pinnacle of Italian cuisine, and that I beat you in one-on-one foosball. Dr. David, the remontada we shared in foosball will live on forever. Dr. Muhamed, back in the dark days before I met my wife, we used to meet, talk, and play foosball almost every weekend in the office; thanks for giving me good (and bad) advice on science, life, etc. Dr. Gaetan, I often think about the hike we completed only to see nothing at the summit due to clouds; it seems like a good metaphor for life. Dr. Elda, thanks for all the good memories at lunch, at

Satellite (pre-COVID), and at barbecues during the summer. Dr. Saeed, I know it was an honor for you to have worked with me; you are the second best Saeed that I know, but the only Saeed to be the godfather of my son. Dr. Gab, I will always cherish our weekly gaming sessions that preserved my sanity during the thesis preparation. Dr. Christophe, remember me when you are the CEO of Disney. Dr. Jony, my fellow Norte-Americano and senpai, thank you for all the fun and games (literally) and the lessons on the grim reality of academic life.

To the youngsters of LTS5: Juan-Luis, you are always welcome in my house, as long as you bring Mexican hot sauce; thanks for all your help and all the fun over the past few years. Remy, while I don't understand how you can be vegetarian, I think your cooking is amazing. Samuel, it's still not too late to monetize my Swiss flag, hot sauce fondue concept; we can discuss it again at the gym. Christian, thank you for teaching me about Swiss politics; remember, always follow the money.

To the doctors of MIAL: Dr. Yasser, maybe one day we can re-discuss our position on globes and massages. Dr. Helene, if you will expand to pediatric MR, I gladly volunteer my son. Dr. Emeline, we will always be on the same team.

To the youngsters of MIAL: Tomasso, I can never remember how to spell your name; pasta and eggs will always be dear to my heart as my greatest contribution to Italian-American fusion cuisine. Hamza, we have become brothers over the last two years, ever since you told me to lose weight; thank you for always being there for me, and hopefully we will eat a sheep's head together one day.

To the people of Siemens Healthineers: Gian Franco, Till, Ludovica, Cipo, Jonathan, Bene, Ricardo, Marija, Davide, Arun, Gabriele, Chloe, Constantin, Stefan, and Tobi. Thank you for welcoming me into the group (virtually) during COVID times and helping me during my internship; I hope to get to know you all better (and in person) in the future!

I will now thank miscellaneous people who did not fit into the above categories. Saleh, you are/were the first Saudi-Arabian person I have known/called a friend; please teach me more curse words so that I can expand my vocabulary. Harshal, your chicken biryani and other Indian dishes have earned you a spot on the team, above Emeline. Markiyan, Shashank, and Saeed (the best Saeed), thanks for the all the fun and support during the master, and the dark times when all of us were looking for jobs/PhD positions.

Of course, I would like to thank my parents, Annie and Jason (the original Dr. Yu). Thank you so much for supporting and loving me my whole life; now that I have a son, I can see how difficult it can be! I could never have done this without you. To my brother James, thanks for your support; until/if you finish your doctorate, please call me Dr. Yu. Thank you to my mother-in-law Karina, for her support and her visit to us in the winter, which gave me the time to actually write the thesis.

My son Joseph was born exactly three days into the last year of my PhD. Joseph, you are welcome.

Finally, I would like to thank my wife, Paola, who, during the writing of the thesis, had to take care of a cranky, demanding person who constantly needs love and attention, as well as my son, Joseph during this challenging last year of the PhD. Without her, not only would this thesis not exist, but my entire life would be impoverished. While a cliche, it is in fact very difficult to

express in words how lucky and grateful I am to have such a wonderful (but also demanding) wife; she is the light of my life. I look forward to the contractually obligatory 18 (and hopefully more) years of marriage we will have together.

Ecublens, April 9, 2022 Thomas

Abstract

Machine learning has become the state of the art for the solution of the diverse inverse problems arising from computer vision and medical imaging, e.g. denoising, super-resolution, de-blurring, reconstruction from scanner data, quantitative magnetic resonance imaging, etc, largely replacing the variational solutions of regularized optimization problems. However, between the two extremes of purely model-driven solutions, such as the solution of regularized optimization problems, and purely data-driven solutions, such as supervised deep learning, exist hybrid methods which combine aspects of both model-driven and data-driven solutions. Such hybrid methods are as manifold as the number of different inverse problems, as the particular characteristics of the inverse problem, e.g. availability of training data, complexity of the forward model, prior knowledge on solutions, etc., will understandably have a huge impact on the structure as well as the underlying techniques of the hybrid method. Furthermore, the validation of such approaches is also of utmost importance, particularly in medical imaging, where there are stringent requirements on the reliability of methods. In particular, hybrid methods are important when large, realistic training datasets are unavailable, such that one cannot immediately apply standard data-driven algorithms.

In this thesis, we examine the solution and validation of four inverse problems derived from Magnetic Resonance Imaging (MRI) and computer vision, where we address the lack of large, realistic training datasets through solutions which try to maximally take advantage of the available model-driven and data-driven resources.

We first show that self-supervised learning embedded in traditional model-driven schemes can be used to robustly solve inverse problems without ground-truth data. We conduct a rigorous validation of self-supervised methods for reconstructing MR images from raw measurement data through novel experiments on clinically relevant data, showing the importance of correct formulation of the forward model/conformity to the training data for image reconstruction quality, critically examining commonly used metrics for quantitative evaluation, and the generalization capabilities of self-supervised approaches. Furthermore, we propose embedding a neural network into a Hamiltonian Markov Chain Monte Carlo (HMCMC) sampling scheme with a self-supervised loss which improves the robustness and accuracy of solutions to a joint diffusometry/relaxometry problem, with respect to state of the art methods.

Complementarily, we show that embedding realistic modeling into standard supervised learning schemes can be used to accommodate the lack of realistic, ground truth data. We combine realistic models and priors to create an extensive synthetic dataset and train a multi-layer perceptron for reconstructing T_2 spectra from MRI data which is more accurate, robust, and orders of magnitude less computationally expensive than the state of the art. Finally, we propose parametrizing an analytically infeasible albeit realistic downsampling model in single image super-resolution through a neural network and integrating it into arbitrary deep learning pipelines which were trained on data with an unrealistic downsampling model, achieving state of the art performance in real-world super-resolution.

Key words: Inverse Problems, Machine Learning, Deep Learning, Self-Supervised Learning, Quantitative MRI, Validation, Super-Resolution, Markov Chain Monte Carlo, Relaxometry

Résumé

L'apprentissage automatique est devenu l'état de l'art dans la résolution de divers problèmes inverses issus de la vision par ordinateur et de l'imagerie médicale, par exemple le débruitage, la super-résolution, le défloutage, la reconstruction des données du scanner, l'IRM quantitative, etc., substituant largement les solutions variationnelles de problèmes d'optimisation régularisés. Cependant, entre les deux extrêmes que sont les solutions purement guidées par un modèle, comme la résolution de problèmes d'optimisation régularisés, et les solutions purement guidées par les données, comme l'apprentissage profond (deep-learning) supervisé, il existe des méthodes hybrides qui combinent des aspects des solutions guidées par le modèle et des solutions guidées par les données. Ces méthodes hybrides sont autant nombreuses que les problèmes inverses, car les caractéristiques particulières du problème inverse, par exemple la disponibilité des données d'entraînement, la complexité du modèle avant (forward model), les connaissances préalables sur les solutions, etc. auront évidemment un énorme impact sur la structure ainsi que sur les techniques sous-jacentes de la méthode hybride. En outre, la validation de ces approches est également de la plus haute importance, notamment dans le domaine de l'imagerie médicale, où la fiabilité des méthodes est soumise à des exigences strictes. En particulier, les méthodes hybrides sont importantes lorsque de grands ensembles de données d'entraînement réalistes ne sont pas disponibles, de sorte qu'il n'est pas possible d'appliquer immédiatement des algorithmes standard guidés par les données.

Dans cette thèse, nous examinons la résolution et la validation de quatre problèmes inverses dérivés de l'imagerie par résonance magnétique (IRM) et de la vision par ordinateur, où nous taclons le problème du manque de grands et réalistes ensembles de données d'entraînement par des solutions qui tentent de tirer le meilleur parti possible des ressources disponibles orientées modèle et données.

Nous montrons tout d'abord que l'apprentissage auto-supervisé (self-supervised learning) intégré dans des schémas traditionnels basés sur des modèles peut être utilisé pour résoudre de manière robuste des problèmes inverses sans données de vérification (ground truth). Nous procédons à une validation rigoureuse des méthodes autosupervisées pour la reconstruction d'images IRM à partir de données de mesure brutes par le biais d'expériences inédites sur des données cliniquement pertinentes, en montrant l'importance d'une formulation correcte du

modèle avant (forward model) et la conformité aux données d'apprentissage pour la qualité de la reconstruction d'image, en examinant scrupuleusement les métriques couramment utilisées pour l'évaluation quantitative, et les capacités de généralisation des approches autosupervisées. De plus, nous proposons d'intégrer un réseau de neurones dans un schéma d'échantillonnage Hamiltonian Markov Chain Monte Carlo (HMCMC) avec une fonction de perte (loss function) auto-supervisée qui améliore la robustesse et la précision des solutions à un problème conjoint de diffusométrie/relaxométrie, par rapport aux méthodes de l'état de l'art.

En complément, nous montrons que l'intégration d'une modélisation réaliste dans des schémas d'apprentissage supervisé standard peut être utilisée pour pallier le manque de données de vérification (ground truth). Nous combinons des modèles ainsi que des a priori réalistes afin de créer un vaste ensemble de données synthétiques et d'entraîner un perceptron multicouche (multi-layer perceptron) pour reconstruire les spectres T2 à partir de données d'IRM, ce qui est plus précis, plus robuste et beaucoup moins coûteux en termes de calcul que l'état de l'art. Enfin, nous montrons qu'un modèle de sous-échantillonnage analytiquement infaisable mais réaliste dans la super-résolution d'une seule image (single image super-resolution) peut être paramétré par un réseau de neurones et intégré dans des pipelines d'apprentissage profond arbitraires qui ont été entrainés sur des données avec un modèle de sous-échantillonnage non réaliste, ce qui permet d'atteindre des performances d'état de l'art dans la super-résolution dans des situations réelles.

Mots clefs : Problèmes inverses, apprentissage automatique, apprentissage profond, apprentissage autogéré, IRM quantitative, validation, super-résolution, Monte Carlo par chaîne de Markov, relaxométrie.

Contents

Acknowledgements **i**

Abstract (English/Français) **v**

List of figures **xiii**

List of tables **xvii**

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Thesis Outline and Contributions . . . 2

2 Background **5**
2.1 Inverse Problems . . . 5
2.1.1 Can you hear the shape of a drum? . . . 5
2.1.2 Theoretical Formulation . . . 7
2.1.3 Examples of Inverse Problems in Imaging . . . 7
2.1.4 Variational Framework for Solution of Ill-Posed Inverse Problems . . . 9
2.1.5 Caveats . . . 16
2.2 Machine Learning for Inverse Problems . . . 17
2.2.1 Motivation . . . 18
2.2.2 Artificial Neural Networks and Deep Learning . . . 19
2.2.3 Supervised Approaches which Implicitly Embed M . . . 22
2.2.4 Supervised Approaches which Explicitly Embed M . . . 23
2.2.5 Comparison of Supervised Approaches . . . 24
2.2.6 Comparison of Machine Learning and Traditional Approaches . . . 24
2.2.7 Our Approach . . . 25
2.3 Closing Remarks . . . 26

3 Introduction to Magnetic Resonance Imaging **27**
3.1 Nuclear Magnetic Resonance for a Single Spin . . . 27
3.1.1 Magnetic Moment . . . 27
3.1.2 Magnetic Moment in a Static Field . . . 29
3.1.3 Magnetic Moment in a Static Field with an RF Perturbation . . . 31
3.1.4 Quantum Note . . . 33

3.2 Nuclear Magnetic Resonance for an Ensemble of Spins 34
3.2.1 Net Magnetization 34
3.2.2 Bloch Equation 35
3.2.3 Bloch-Torrey Equation 36
3.3 Signal Detection 37
3.4 Basic Imaging Model 38
3.5 Image Contrast 41

4 Validation of Self-Supervised, Undersampled MRI Reconstruction 45
4.1 Introduction 45
4.1.1 Parallel, Undersampled MRI Reconstruction 45
4.1.2 Motivation 47
4.1.3 Contributions 47
4.2 Theory 48
4.2.1 DeepDecoder 49
4.2.2 Self-supervised learning via data under-sampling 50
4.3 Methods 51
4.3.1 Training Data and Hyperparameter Tuning 52
4.3.2 Validation using Prospectively Accelerated and Fully Sampled Data . . . 53
4.3.3 Generalizability of Self-Supervised Reconstruction Methods 53
4.3.4 Statistical Significance 55
4.4 Results 55
4.4.1 Validation Using Prospectively Accelerated and Fully Sampled Data . . . 55
4.4.2 Generalizability 59
4.5 Discussion 60
4.5.1 Validation using Prospectively Accelerated and Fully Sampled Data . . . 60
4.5.2 Generalizability 67
4.5.3 Ranking Methods through Quantitative Metrics 68
4.5.4 Future of Validation 69
4.6 Conclusion 69

5 Neural Network Enhanced MCMC 71
5.1 Introduction 71
5.1.1 Probabilistic Framework and MCMC for Inverse Problems 71
5.1.2 Multi-Compartment T_2 Relaxometry/Diffusometry 73
5.1.3 Contributions 76
5.2 Related Work 76
5.2.1 Hamiltonian Markov Chain Monte Carlo 76
5.2.2 L2HMC 78
5.3 Methods 79
5.3.1 Neural Network Enhanced Hamiltonian MC (NNEHMC) 79
5.3.2 Biophysical Parameter Estimation 81
5.4 Results and Discussion 82

5.4.1 Strongly Correlated Gaussian . . . 82
5.4.2 Multi Echo Spherical Mean Technique (MESMT) . . . 83
5.5 Discussion and Conclusion . . . 84

6 Model Informed Machine Learning 87
6.1 Introduction . . . 87
6.1.1 Related Work . . . 88
6.1.2 Contributions . . . 93
6.2 Methods . . . 93
6.2.1 Synthetic Dataset Generation . . . 94
6.2.2 Mapping the MR Signal to the T_2 Distribution . . . 97
6.3 Evaluation . . . 99
6.3.1 Synthetic Data . . . 100
6.3.2 Real Data . . . 100
6.4 Results . . . 102
6.4.1 Synthetic Data . . . 102
6.4.2 Real Data . . . 106
6.4.3 Computation Time . . . 116
6.5 Discussion . . . 116
6.6 Conclusion . . . 122

7 Using Realistic and Bicubic Downsampling for Super-Resolution 123
7.1 Introduction . . . 123
7.2 Contributions . . . 126
7.3 Related Work . . . 127
7.3.1 Real-World SR through real data . . . 127
7.3.2 Real World SR through extended models . . . 128
7.4 Methodology . . . 129
7.4.1 Overall pipeline . . . 129
7.4.2 Bicubic look-alike image generator . . . 129
7.4.3 SR generator . . . 132
7.4.4 Training parameters . . . 132
7.5 Experimental results . . . 133
7.5.1 Test images . . . 133
7.5.2 Quantitative results . . . 134
7.5.3 Qualitative comparison . . . 135
7.5.4 User study . . . 135
7.6 Additional Experiments . . . 136
7.6.1 Generalizability of RBSR . . . 136
7.6.2 Ablation study . . . 136
7.6.3 Computational cost . . . 137
7.7 Conclusion . . . 137

8 Discussion and Conclusion **141**
8.1 General Discussion 141
8.1.1 Self-Supervised vs. Supervised Approaches 141
8.1.2 Validation 141
8.1.3 Properties of Measurement Data and Solutions 142
8.2 Future Work and Conclusion 143

Bibliography **163**

Publications **165**
Articles in peer-reviewed journals 165
Articles in proceedings of international conferences 166
Abstracts in proceedings of international conferences 167

Curriculum Vitae **169**

List of Figures

2.1 Isospectral shapes 6
2.2 Example Inverse Problems 8
2.3 Noise Unstable reconstruction from only data consistency 10
2.4 Degeneracy of the In-painting Inverse Problem 11
2.5 Example Denoising Reconstructions 17

3.1 Example of Orbital and Spin Angular Momentum 28
3.2 Current Loops in a Magnetic Field 29
3.3 Geometrical Derivation of Larmor Precession 30
3.4 Magnetic Moment in a Rotating Reference Frame 33
3.5 Ensemble of Spins 35
3.6 Example K-space with Corresponding Magnitude Image 41
3.7 Examples of Contrast Image 43

4.1 Method Overview 48
4.2 Prospective/Retrospective Reconstructions of a Multi-Purpose Phantom 56
4.3 MPRAGE: Prospective/Retrospective Reconstructions of Fruits/Vegetables . . . 57
4.4 MPRAGE: Axial Brain Slices 61
4.5 MPRAGE: Closeups of the cerebellum and an axial slice 62
4.6 PD SPACE: Sagittal slice of the knee with sagittal/axial closeups 63
4.7 MPRAGE: Axial brain slices from different perturbations 64
4.8 SPACE: Axial brain slices and a sagittal knee slice 65
4.9 Barplots of No-reference image metrics and human ratings 66

5.1 Sequence of T_2 Images 74
5.2 Flowchart of the training algorithm for neural network parametrization of HMCMC 80
5.3 Plot of the average autocorrelation of 200 chains of length 2000 for L2HMC and NNEHMC with the corresponding effective sample size. 82
5.4 Box plots of relative absolute errors from ground truth using least squares (blue), NUTS (orange), L2HMC (green), and NNEHMC (red). 83
5.5 Representative plots of the marginal probability distributions for each parameter, where the black vertical line denotes the ground truth value. 84

6.1 An overview of our Model-Informed Machine Learning(MIML) for multicomponent T_2 relaxometry where we learn a mapping from the multi-echo MR signal to the corresponding T_2 distribution. 89
6.2 Reconstructions from two different SNRs (40,1000) using NNLS with Laplacian regularization 91
6.3 Wasserstein Distance, KL Divergence, and the MSE changes between two, non-intersecting lobes, as one is shifted closer to the other. 98
6.4 Plots of the mean distribution over all of the ground truth distributions in the test split of the synthetic dataset, as well as the mean reconstructed distribution over the test split 103
6.5 Boxplots of the MSE and Wasserstein Distance between the ground truth distributions in the test set of the synthetic dataset and the corresponding, reconstructed distributions from each method over a range of different SNRs. 104
6.6 Mean reconstructed distributions (ground truth and from each method) over a range of SNRs as well as boxplots of the MSE and Wasserstein distance between the ground truth and reconstructed distributions from the results on the realistic, synthetic case. 107
6.7 Distribution reconstructions from different noise realizations for SNRs 200 and 1000 on the realistic, synthetic case. 108
6.8 MWF maps from each method, the histology map, and the reconstructed distributions for each method on the ex-vivo data. 109
6.9 Example MWF maps produced from each method, in the axial, coronal, and sagittal planes of two healthy subjects. 113
6.10 Reconstructed distributions (in color) in the WM voxels of the axial slices of two healthy subjects for each method. 114
6.11 Boxplots of the mean MWF (left) and the standard deviation of the MWF (right) for each method over all the WM ROIs for each subject in the cohort of healthy subjects. 114
6.12 Anatomical FLAIR map where the white matter is hypointense (first column), maps of the MWF (second column), and maps of the geometric mean T_2 in the range corresponding to the IE space (50-200ms) (third column) for an axial slice in a subject with MS. 117
6.13 Zoom in on the lesions as well as the corresponding patches in the MWF map . 118
6.14 T_2 distributions within the lesion mask and the same mask translated to the normal appearing region contralateral to the lesion for each lesion and method. 119

7.1 An example SR from RBSR and RealSR on a real-world LR image 124
7.2 Downsampling kernels estimated from bicubically and realistic downsampling 125
7.3 Two-step pipeline for real world SR 128
7.4 Schematic diagram of the bicubic-alike decoder. 130
7.5 Illustration of the effectiveness of using bicubic perceptual loss 138

7.6 Example images generated with and without the copying mechanism during training 138
7.7 Qualitative results of ×4 SR on a variety of datasets 139
7.8 Results of the user study 139
7.9 Example screenshot from the performed online survey 140
7.10 Comparing results from using RBSR with RCAN and ESRGAN 140

List of Tables

4.1 PSNR/SSIM/PD with respect to Ground Truth 58
4.2 Sequence Parameters 70

6.1 The ranges for the possible mean (μ) and the standard deviations (σ) of simulated water pools 94
6.2 Spatial Pearson correlations (with p-values) between the MWF maps constructed from each method and the histology map of the myelin in a white matter mask. 109
6.3 The regions of interest (ROI) in the brain used for analysis 111
6.4 The mean and standard deviation of the absolute difference between the mean MWF values of the scan and rescan in white matter ROIs for each method and for each healthy subject. 112
6.5 The spatial Pearson correlation and the linear regression coefficients (slope and intercept) between the mean MWF values of the scan and rescan in white matter ROIs for each method and each healthy subject. 112
6.6 The average computation time for whole brain reconstructions for each method on the healthy subjects 116

7.1 Quantitative comparison of different methods 134
7.2 Ablation study of the components of RBSR 136

1 Introduction

Image reconstruction in computer vision and medical imaging can often be formulated as an inverse problem, where one reconstructs an image from measurements for which the model relating the measurements to the image is well or approximately known [1]–[3]. If algebraic or explicit solutions are not feasible, inverse problems are traditionally solved through the solution of optimization problems, though Markov Chain Monte Carlo (MCMC) methods have also been used. In particular, regularized optimization schemes, where a prior on the reconstructed image, such as sparsity in a certain signal domain, is imposed were until recently the state of the art for image reconstruction. We denote these approaches as model-driven, as they explicitly leverage knowledge of the measurement model as well as priors on plausible solutions. With the advent of deep learning, approaches based on end-to-end, supervised machine learning, where, for example, mappings from the measurements directly to the desired image were learned over large datasets, quickly overtook regularized optimization schemes as the state of the art [4]. As these approaches mainly or solely leverage information from the training datasets, we denote these approaches as data-driven.

1.1 Motivation

Supervised machine learning generally require large amounts of realistic training data for learning accurate and robust mappings [5]. For example, problems arise when supervised methods are trained on insufficient data or unrealistic data, such as in cases where ground truth data is difficult or infeasible to obtain. Consequently, problems with domain shifts and generalization often arise when data-driven methods are used in the real world. Furthermore, validating methods also become difficult without realistic data or ground truth. Ideally, reconstruction methods will leverage all the information available; both model-driven information (i.e. realistic modelling, knowledge of a-priori constraints/characteristics of plausible solutions, the use of model-driven schemes, etc.) and data-driven information (i.e. using machine learning on the available data). In practice, the optimal ratio of model-driven and data-driven components of a given reconstruction method depends significantly on the specifics of the inverse problem considered; for example, whether the forward model is known/efficiently

computable or not, the feasibility of generating a sufficient amount of realistic, synthetic data, the desired tradeoff between computational complexity and performance, the amount of training data available, etc.

Therefore the main questions explored in this thesis are as follows.

- How can data-driven methods, e.g. machine learning, be used to help solve inverse problems in the setting where **no or very limited, realistic datasets are available which could be used for standard, supervised learning**?
- How can data-driven and model-driven methods be combined to form hybrid methods which leverage all available information?
- How can methods for solving inverse problems be quantitatively or qualitatively validated in the setting where little to no ground truth data is available?

The main chapters of this thesis focus on answering these questions in the context of 4 different inverse problems which arise from Magnetic Resonance Imaging (MRI) and computer vision, with different chapters emphasizing certain questions more than others. **These inverse problems are all connected by a lack of large, realistic training datasets that could be used for simple application of supervised learning or validation.**

In particular, we proposed and studied two orthogonal strategies:

- Embedding **self-supervised** (i.e., learning only from measurement data) neural networks into traditional, model-driven schemes.
- Embedding **realistic modelling** into the pipelines of standard, supervised machine learning approaches.

1.2 Thesis Outline and Contributions

The thesis is organized as follows:

- **Chapter 2** consists of a general, non-rigorous introduction to inverse problems, particularly as it relates to image reconstruction. We start from the formulation of generic inverse problems, define a traditional, variational framework for solving generic inverse problems through convex optimization, and conclude with a description of machine learning for solving inverse problems. This chapter gives context for understanding the contributions of this thesis from a methodological point of view.
- **Chapter 3** consists of an introduction to Magnetic Resonance Imaging (MRI) in order to give context for the inverse problems in MRI which are addressed in the subsequent chapters (4, 5, and 6).

In Chapters 4 and 5, we validate (Chapter 4) and propose (Chapter 5) novel methods where self-supervised learning is embedded into different model-driven schemes for more accurate and robust solution of inverse problems where no ground truth data is generally available. Specifically:

- In **Chapter 4**, we begin with a rigorous validation of self-supervised methods for reconstructing MR images from raw, undersampled measurements. This chapter first illustrates the importance of validating on realistic, undersampled datasets by showing significant differences between reconstructions performed on prospectively vs. retrospectively undersampled data in an MR phantom and an assortment of fruits and vegetables. It also draws inspiration from the computer vision literature to critique and suggest alternatives to image metrics commonly used for validating MR image reconstruction. Finally, it showcases the potential for generalizability of self-supervised methods using an extensive dataset of different sequences, as well as showing the potential for no-reference image metrics to be used for quantitative evaluation when no ground truth is available. The results of this chapter are a first step toward realizing standardized, realistic validation of machine learning methods for undersampled MR reconstruction which is necessary for future deployment in the clinic.
- In **Chapter 5**, we continue with self-supervised learning by proposing to solve inverse problems in a probabilistic framework through embedding a self-supervised neural network into a Hamiltonian Markov Chain Monte Carlo (HMCMC) sampler; we extend an existing approach by modifiying the self-supervised loss function to enforce more conformity to ideal Hamiltonian dynamics, showing that this results in faster mixing and greater robustness. We test the efficacy of our proposed method on a complex, extremely nonlinear inverse problem in MRI derived from joint relaxometry and diffusometry, showing greater robustness and accuracy in comparison to the state of the art.

In Chapters 6 and 7, we show how to integrate realistic training data, through either judicious use of a small quantity of physically acquired data or the synthetic generation of a large dataset of realistic data, into supervised methods to solve inverse problems for which large, realistic training datasets do/did not exist. We show that this results in improvements in all dimensions of inverse problem solving compared to the state of the art: accuracy, robustness, and speed. Specifically:

- In **Chapter 6**, we pivot to supervised learning for multi-component T_2 relaxometry, where no physical, ground truth data is available; we focus on relaxometry in the white matter of the brain/nervous system, where there is much prior information. We show that realistic forward modelling and realistic priors on the structure of the solutions can be combined to generate a large, realistic, and synthetic dataset for training a simple multi-layer perceptron, achieving better accuracy, robustness, and orders of magnitude difference in speed in comparison to state of the art methods. Furthermore, we show

that a Wasserstein loss function tailored to the specific solutions, in this case probability distributions, can significantly help in the accuracy of solutions in the estimation of the contribution from myelin. The results of Chapters 5 and 6, in providing new methods for diffusometry and relaxometry, contribute to the analysis of MRI biomarkers for pathology and research on neurodevelopment.

- In **Chapter 7**, we conclude with supervised learning for real-world single image super-resolution (SR), where again, usually little to no physical, ground truth data is available; most approaches use artificial, bicubic downsampling to generate unrealistic datasets for training, resulting in poor performance on "realistic" low-resolution images. We show that a small amount of realistic, ground truth data can be leveraged to train a network which, roughly, maps the space of "realistic", low resolution images to the space of bicubically downsampled images. This network can then be used in conjunction with any standard network trained on bicubically downsampled data to generate a high resolution image. Hence, we propose a method which leverages both a small amount of realistic data and a large amount of unrealistic data such that one can robustly perform super-resolution on realistic images while still being able to reuse virtually any previous, pretrained SR network. Our method facilitates efficient and easy adaptation of existing SR networks for real-world super-resolution, with applications ranging from television to cell-phone images.

- **Chapter 8** first discusses the contributions of this thesis in a general context, followed by a conclusion with an eye toward future work.

2 Background

2.1 Inverse Problems

We note that the mathematical foundations of inverse problem theory are generally framed in terms of functional analysis, with particular emphasis on theoretical estimates/bounds for stability, convergence, and error of proposed solutions. In this background section, we provide a brief and high-level overview of inverse problems, leaning more towards accessibility and relevance to later sections of the thesis rather than an extensive, rigorous mathematical exposition; readers interested in the latter are directed to the following references which were used to guide this chapter: [1], [2], [6], [7].

2.1.1 Can you hear the shape of a drum?

We begin with a decidedly non-imaging related, canonical example of an inverse problem, popularized by the mathematician Mark Kac, which can be summarized in a single question: can you hear the shape of a drum [8]?

More precisely, consider a clamped drum in 2D which is modeled by a domain Ω with a condition on its boundary, $\partial\Omega$.

Then the frequencies of the normal modes of the drum, λ, can be determined from the solution of the following partial differential equation, derived from a separation of variables approach to solving the wave equation for the height of the drum:

$$\Delta U + \lambda U = 0 \tag{2.1}$$

$$U \restriction_{\partial\Omega} = 0 \tag{2.2}$$

where $U : \mathbb{R}^2 \to \mathbb{R}$.

We note that given the domain Ω, it is straightforward to calculate, analytically or computationally, the normal mode frequencies; the above model which takes as input the domain Ω

and provides the frequencies, λ, is called the **forward model**.

The question then arises: does the spectrum of frequencies uniquely determine the domain Ω? That is, if the domain Ω is unknown, can we cover it from measurement of the spectrum? This is called the **inverse problem**, as we are seeking to invert the forward model. In general the answer is no, as proven by counterexample in [9]. In Fig. 2.1, are two shapes which have the same spectrum of frequencies, constructed by the authors in [9].

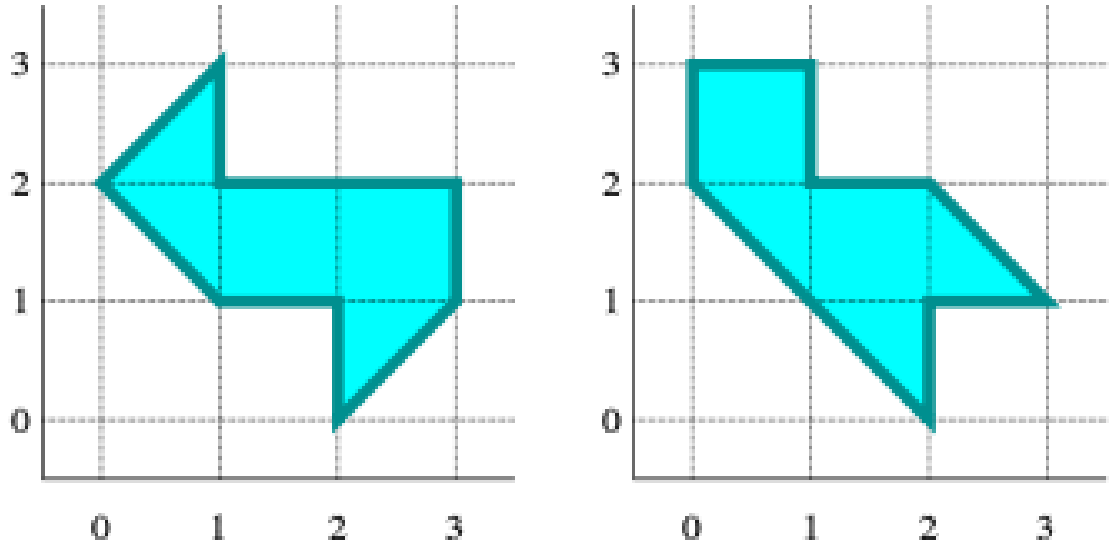


Figure 2.1: Here are two shapes which generate the same spectrum of frequencies [9] [10]

However, while recovering the shape is not generally possible, it **is** possible to recover the area of the drum from its spectrum as shown in [11]. Let $N(\lambda)$ be the number of frequencies less than λ, A the area of Ω. Then Weyl's law in 2D states that:

$$A = 4\pi \lim_{\lambda \to \infty} \frac{N(\lambda)}{\lambda} \tag{2.3}$$

Furthermore, in [12], the authors show that in 2D, one **can** hear whether a drum has corners or not, i.e. whether the boundary of Ω is smooth or not fundamentally changes the spectrum of frequencies.

This problem nicely illustrates fundamental problems, issues, and questions to consider for the solution of inverse problems in general, particularly from a practical viewpoint.

- **Modelling**:
  - **How realistic is the model?**: As this is a toy model, it is not clear whether modelling the system as 2D/with the above PDE is realistic enough.
  - **How computationally expensive is the forward model?**: Depending on the domain/dimension, simulating the PDE above can be quite expensive.

- **Measurement**
  - **How accurate/noisy are the measurements?**: It is not clear how accurately the spectrum of frequencies can be measured in real life, whether due to the impact of bias or random noise.
  - **How many measurements do you need?**: Weyl's law applies asymptotically; therefore, it not clear how well characterized the spectrum must be for an accurate estimation of A.
- **Theory**:
  - **What are the limits of what can be recovered?**: [9] gives a fundamental limit on what we can recover from measurements of the spectrum.
  - **If recovery is possible, how can it be accomplished?**: While there is a formula for calculating A, we note that there is no formula for determining whether a shape has corners; it was only proved that it was possible.

2.1.2 Theoretical Formulation

In the following, we will only consider finite dimensional inverse problems for accessibility/conceptual understanding; however, proper theoretical formulation takes place between infinite dimensional function spaces (Hilbert/Banach spaces), e.g. images are modeled as continuous functions on $\mathbb{R}^2$. However, for computational solutions, generally one passes to a discretized version of the continuous inverse problem, e.g. pixelization of images.

Therefore, let $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}$, and a function $M : \mathbb{R}^n \to \mathbb{R}^m$. The goal of inverse problems is to recover $\mathbf{x}$ from the measurements $\mathbf{y}$, given knowledge of a model function M such that

$$\mathbf{y} = M(\mathbf{x}) + n \tag{2.4}$$

where n indicates measurement noise. Note that if M has a well-defined inverse and there is no measurement noise, then the solution is straightforward. For interesting inverse problems, this is not true or simple inversion is insufficient for good quality/unstable; however, the notion of inverting M is where the term "inverse problems" originates. This formulation encompasses a wide variety of applications in engineering and physics; however, in this chapter, we focus on applications in imaging as they are relevant to the thesis and easy to understand.

2.1.3 Examples of Inverse Problems in Imaging

In many inverse problems related to images, $\mathbf{y}$ is a distorted image, $\mathbf{x}$ the undistorted image,and M is the model for distorting $\mathbf{x}$. Below we list some simple inverse problems in imaging where we identify M (see Fig. 2.2).

- **Denoising**: In this inverse problem, we want to remove the noise from an image. Here M is the identity function, with $\mathbf{y} = \mathbf{x} + n$.

- **Deblurring**: In this inverse problem, we want to deblur a blurred image. Here M can be modeled as a convolution with a blur kernel, e.g. a Gaussian kernel.

- **In-painting**: In this inverse problem, we want to fill in missing portions of an image, using only the visible portions of the image. Here M can be modeled as a masking function which sets portions of an input image to zero.

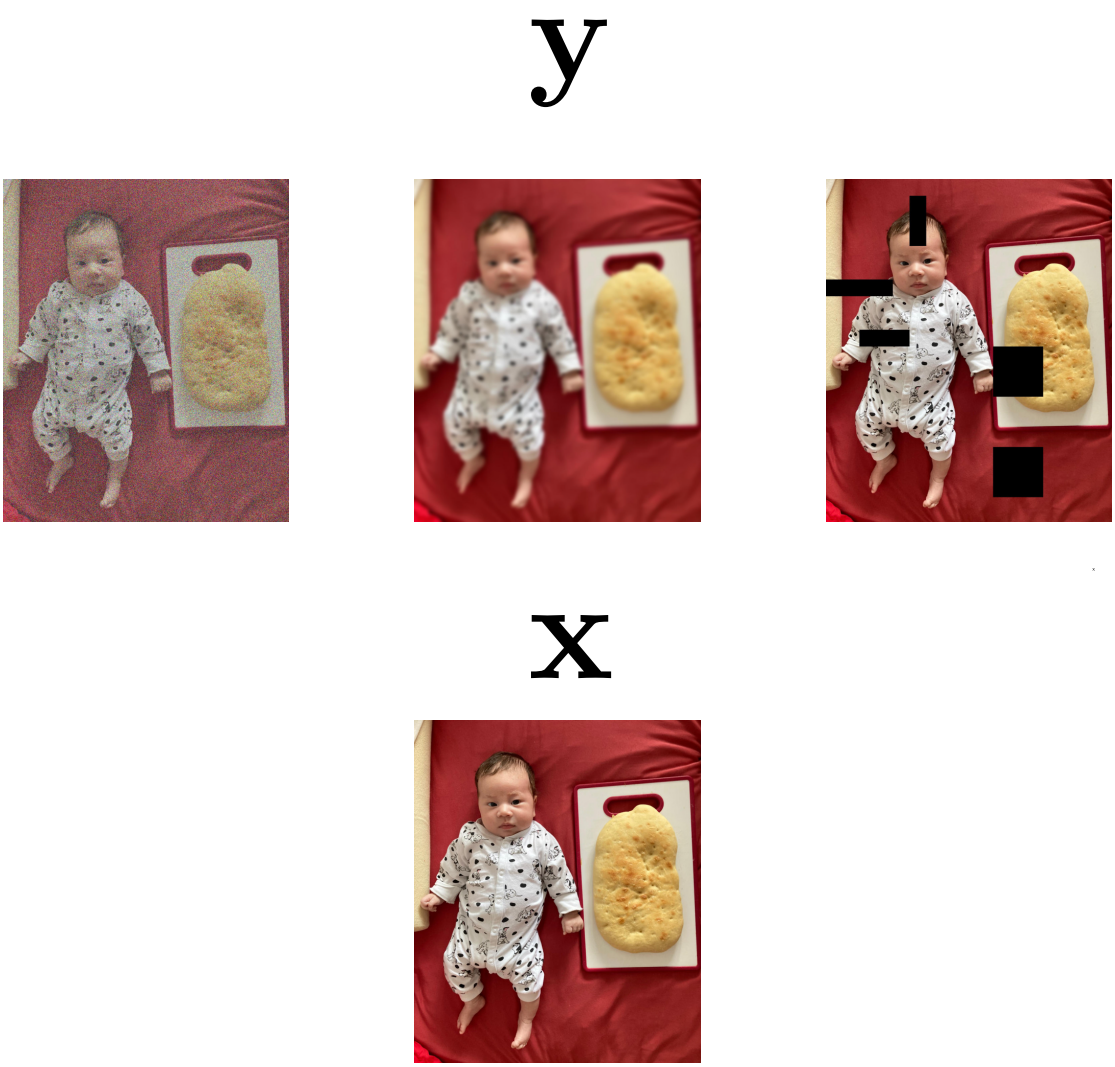


Figure 2.2: Here are three example inverse problems: denoising, deblurring, and in-painting, from left to right. On the top row, we show the measurements $\mathbf{y}$, and the bottom row shows the ground truth signal $\mathbf{x}$.

2.1.4 Variational Framework for Solution of Ill-Posed Inverse Problems

We can examine the solution of inverse problems in light of Hadamard's famous criteria for a well-posed mathematical problem [13]:

- That a solution to the inverse problem exists for any given $\mathbf{y}$.
- That given measured data, $\mathbf{y}$, the solution to the inverse problem is unique.
- That the solution to the inverse problem is stable or depends continuously on $\mathbf{y}$.

The solution of interesting inverse problems are generally ill-posed, in the above sense, due to violating the second and third conditions: non-uniqueness of solutions and instability with respect to the input data. In the following, we outline a generic framework for solving ill-posed inverse problems.

Given no other information other than the triple $(M, \mathbf{x}, \mathbf{y})$, one might naively solve Equation 2.4 by

$$\mathbf{x} = \arg\min_{\mathbf{x}'} D(M(\mathbf{x}'), \mathbf{y}) \tag{2.5}$$

where we are simply optimizing for the vector which, upon applying the forward model, best fits to the data, using some function D to compare the predicted and measured data. For example, $D(M(\mathbf{x}), \mathbf{y}) = \|M(\mathbf{x}) - \mathbf{y}\|_2^2$ is commonly used.

However, this solution already exemplifies the ill-posedness discussed above.

- As noise is usually present in the measurements, it may overfit to the noise or the model function may be unstable with respect to noise, leading to suboptimal solutions.
- It can be unstable/inaccurate due to degeneracy of solutions.

As an example of the first, consider Fig. 2.3, where the forward model M is multiplication by a matrix with a high condition number, meaning small changes in the input vector lead to large changes in the output vector. We can see that even simple inverse problems (solving a system of linear equations) can demonstrate high instability with respect to noise.

As an example of the second, consider Fig. 2.4, where we show that the in-painting inverse problem is inherently degenerate. Given an image with missing patches, there are an infinite number of images which, upon applying the forward model i.e. masking, returns the original image with missing patches. Each of the candidate images in Fig. 2.4 yields the exact same error with respect to the measured data, i.e. the masked image: zero. This degeneracy prevents the solution of Eq. 2.5 from being useful.

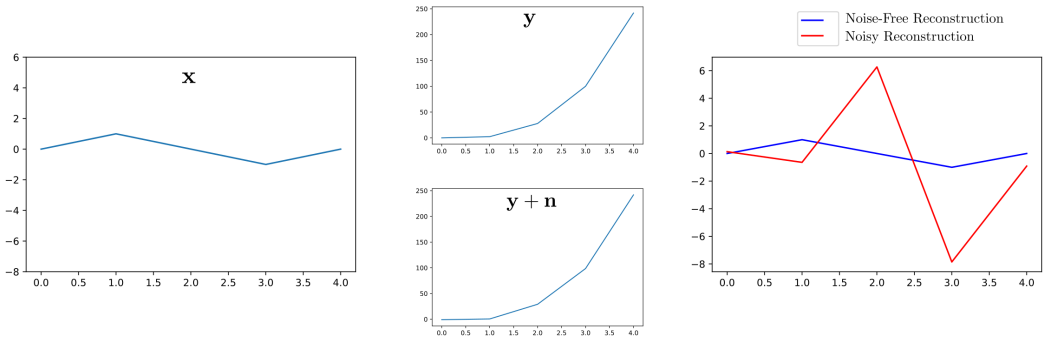


Figure 2.3: Let M be a linear function corresponding to multiplication by a Vandermonde matrix from a given vector. These matrices are known to be highly ill-conditioned. To the left, we show the ground truth signal. In the middle, we show the corresponding measurement vectors, without (top) and with (bottom) noise; note that the ratio of the L_2 norm of the noise to that of the measurement vector is less than 1 percent. To the right, we show the reconstruction from solving Eq. 2.5 using $D(M(\mathbf{x}),\mathbf{y}) = \|M(\mathbf{x}) - \mathbf{y}\|_2^2$, for both noise-free and noisy cases. We can see that the addition of even a small amount of noise to a measurement vector can drastically alter the resulting reconstruction, demonstrating the problem's inherent noise instability.

Therefore, for ill-posed problems, we can see that there must be additional criteria/assumptions other than best fit to noisy data to robustly solve inverse problems.

In Fig. 2.4, perceptually, we have an idea of what the missing patches of the images should look like, even if we had not seen the original image. This is because we have prior knowledge, having seen these kinds of images before. We know that the bottom-most patch is most likely a continuation of the red background. We know that the missing patch near the bread is most likely a continuation of the bread, etc. In fact, one general guess would be that missing patches should be similar to the image content near them. Therefore, we could reject all but the ground truth image simply based on the implausibility of the proposed images. This idea of using prior knowledge or assumptions on $\mathbf{x}$ as an additional criterion for judging the quality of solutions is called **regularization**, and can be implemented as follows:

$$\mathbf{x} = \arg\min_{\mathbf{x}'} D(M(\mathbf{x}'),\mathbf{y}) + \lambda R(\mathbf{x}'). \tag{2.6}$$

Here $R : \mathbb{R}^n \to \mathbb{R}$ is a regularization function which encodes prior knowledge/assumptions on $\mathbf{x}$ and penalizes deviation from these during optimization. λ is the regularization parameter which determines the weight given to R vs. the data consistency term during optimization. We note that while we motivated R as injecting prior knowledge, it can also be used to stabilize solutions to Equation 2.6. We further note that this framework for solving inverse problems is called a **variational** framework as the optimization is over classes of functions, e.g. images. While there are other frameworks for solving inverse problems, e.g. spectral frameworks, which are concerned with the spectrum of M when it is a linear operator, variational frameworks

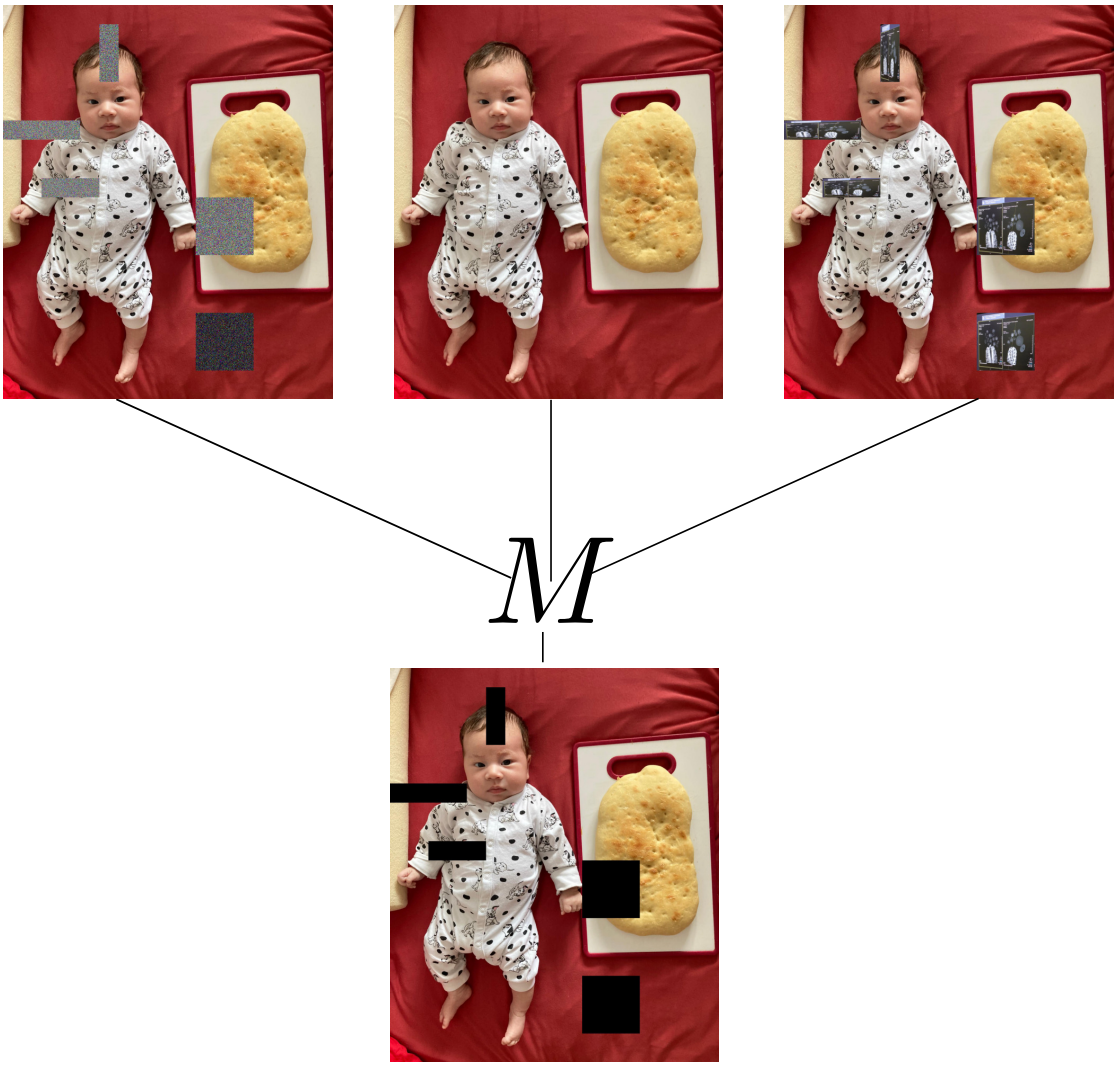


Figure 2.4: Let us consider the in-painting inverse problem, where M masks select patches from an image and sets them to zero. In the top row we show three candidate solutions, along with their image upon operation by M. From this example, we can see with respect to Eq. 2.5, that for any reasonable function D, all three of the images in the top row will have zero error in fitting to the data since their masked version are all the same/equal to the initial measurement. As these three images include both implausible images (left and right) as well as the actual ground truth (middle), we can see that the degeneracy of the model makes the solution of Eq. 2.5 unhelpful as it will have infinite global minima.

dominate in inverse problems in imaging. Hence, with different choices of R, Equation 2.6 is a generic framework for solving ill-posed inverse problems.

Below we list some common types of regularization in inverse problems:

- **Tikhonov Regularization [14], [15]**: $R(\mathbf{x}) = \|T\mathbf{x}\|_2^2$. Here T is a linear operator which can

be, for example, the identity, which encourages $\mathbf{x}$ with minimal L_2 norm, or difference operators which approximate the derivative, encouraging smoothness of $\mathbf{x}$. This is motivated by the fact that, for example, using T with the identity operator and $\lambda > 0$ guarantees a unique solution to 2.6 when M is linear; in this case, this choice can be interpreted as a filter which removes the effects of small singular values of M, which are one of the sources of instability in solving Equation 2.5.

- **Total Variation Regularization [16]**: Let $\mathbf{v}$ be a discretized 2D image with indices i, j. Then the total variation, $TV(\mathbf{v}) = \sum_{i,j} \sqrt{\|v_{i+1,j} - v_{i,j}\|^2 + \|v_{i,j+1} - v_{i,j}\|^2}$. This is the discretization of the integral of the norm of the gradient of an image. If $\mathbf{x}$ is the flattened vector corresponding to $\mathbf{v}$, then $R(\mathbf{x}) = TV(\mathbf{v})$. This regularization is motivated by the observation that in noisy images, there are abrupt shifts in image intensity across the whole image, meaning the norm of the gradient of the image is high over the whole image. This implies noisy images will have a high total variation, since TV is the spatial integral of the norm of the gradient of the image. Hence, using TV regularization should penalize noisy instances of $\mathbf{x}$ during optimization.

- **Compressed Sensing Regularization [17], [18]**: Let W denote the transformation matrix to an arbitrary orthonormal basis. Then a compressed sensing regularization has the form $R(\mathbf{x}) = \|W\mathbf{x}|\|_1$, which is the L_1 norm of $\mathbf{x}$ expressed in the orthonormal basis. This regularization is motivated by two observations. First, is that generally, signals (such as natural images) are sparse when expressed in certain bases; e.g. when a natural image is expressed in a wavelet basis, only relatively few elements of the image vector in this basis will be significant, with others being small/insignificant. This is in contrast to the usual spatial (pixel) basis of images, in which the images are dense. Second, is that while it is difficult/impossible to optimize the sparsity of $W\mathbf{x}$, optimizing the L_1 norm, $\|\mathbf{x}\|_1$, is feasible and can be shown to be a good proxy for the sparsity of $\mathbf{x}$ under some assumptions. Combining these observations, compressed sensing (CS) regularization promotes the sparsity of $\mathbf{x}$ in a convenient domain, encoding the assumption that desirable solutions are sparse in that domain. The practical importance of CS is that with this regularization/assumption, one can reconstruct sparse signals from far fewer measurements than expected by, for example, the Shannon-Nyquist limit [19]; this can greatly accelerate processes where measurements are time-consuming, such as in magnetic resonance imaging.

As can be seen above, the variational framework in Equation 2.6 is extremely flexible, allowing for arbitrary assumptions/prior knowledge/constraints to be imposed on $\mathbf{x}$ through R, as long as R can be formulated in a mathematically tractable way; indeed the vast majority of inverse problems in imaging (and all the inverse problems addressed in this thesis) can be formulated and efficiently solved in this framework.

Within this framework, given an inverse problem, one must select three things to proceed:

- what regularization function R to use,
- what optimization algorithm is used to solve Equation 2.6,
- what strategy to use to set λ.

With all three choices, there are a wide variety of options to choose from, with an enormous amount of literature covering all three aspects. As we have already given several examples of regularization functions, we briefly give some examples for the latter two.

Setting the Optimization Algorithm

In order to solve Equation 2.6, one must select an optimization algorithm; a suitable algorithm depends on the relevant mathematical properties of D, M and R; i.e., the linearity/non-linearity, degree of differentiability/smoothness, convexity, etc. In general depending on these properties, one chooses an algorithm with some theoretical proof of convergence (given the assumed properties of D, M and R), an algorithm which works empirically, or preferably both. From a practical viewpoint, the data consistency term $D(M(\mathbf{x}),\mathbf{y})$ is usually one which is smooth, convex, etc, enjoying nice properties (e.g. when M is linear and using the L_2 norm for D). Therefore, selection of algorithms generally hinge on the choice of R, whose mathematical properties can vary greatly. Furthermore, depending on the scale of the problem (e.g. if the discretization of M is a prohibitively large matrix to store/compute with), iterative algorithms which only require matrix-vector multiplications rather than storing matrices may be preferred/required. Here we briefly describe two classes of algorithms relevant for the problems in this thesis, with more detail given as necessary in the subsequent chapters of the thesis.

If D,M and R are differentiable/smooth: then simple gradient descent/other iterative algorithms which rely on taking derivatives will work; for instance, if we let $D(M(\mathbf{x}),\mathbf{y}) = \|M(\mathbf{x})-\mathbf{y}\|_2^2$, $R = Id$ as in Tikhonov regularization, and M is also linear, the resulting optimization problem reduces to solving a system of linear equations as the first order optimality condition can be written explicitly as such. Then algorithms for solving these systems can be used, e.g. the conjugate gradient algorithm [20]. Furthermore, if the entire optimization problem can be recast into either a linear or nonlinear least squares form, one can use the Newton/Gauss-Newton/Levenberg-Marquardt algorithms [21].

In many inverse problems in imaging, M is linear, $D(M(\mathbf{x}),\mathbf{y}) = \|M(\mathbf{x})-\mathbf{y}\|_2^2$ (hence smooth and convex), and R is not differentiable but is convex. For example, in compressed sensing, the L_1 norm is necessary for R, since it serves as a proxy for sparsity; however, since it is not differentiable, gradient based methods cannot be used. In this case, there are a broad class of closely related algorithms (alternating direction method of multipliers [22], forward-backward splitting [23], Douglas Rachford Splitting [24], Proximal Gradient Descent [25], Primal Dual

Splitting [26], etc.) which can solve optimization problems of the form

$$\mathbf{x} = \arg\min_{\mathbf{x}'} D(M(\mathbf{x}'), \mathbf{y}) + R(\mathbf{x}') \tag{2.7}$$

where D is a smooth/convex function of $\mathbf{x}'$ and R is proper, lower semi-continuous, and convex. Very imprecisely, these are iterative methods which have update steps which alternate between optimizing D and R (hence the splitting) in order to converge to a solution; as R is not differentiable, global update steps using the gradient of the sum of the terms is not possible. We will define proximal gradient descent as it is the simplest but conveys the general idea behind these algorithms. For all the algorithms mentioned above, the following notion is important; given a function $h : \mathbb{R}^n \to \mathbb{R}$ and a point $\mathbf{x}$, the proximal operator [27] of h evaluated at $\mathbf{x}$ is defined by

$$prox_{h,t}(\mathbf{x}) = \arg\min_{\mathbf{z}} \frac{1}{2t}\|\mathbf{x}-\mathbf{z}\|_2^2 + h(\mathbf{z}). \tag{2.8}$$

Note that if h is closed, lower-semicontinuous, and convex, then this definition is well-defined; $\forall \mathbf{x}$, the solution always exists and is unique. In the case where h is the indicator function of of a closed and convex set, then the proximal operator is the projection operator onto that set. This motivates the definition, as somehow the proximal operator is trying to "project" a given point onto the set of minima of h, without having to take the gradient of h. Though the definition requires solving an optimization problem for each $\mathbf{x}$, analytical expressions for the proximal operator are known for a wide variety of h, including those commonly used for regularization in inverse problems. Then proximal gradient descent for solving Equation 2.7 is an iterative algorithm where beginning with an initial guess $\mathbf{x}^{(0)}$, the kth iteration is defined by

$$\mathbf{x}^{(k)} = prox_{R,t_k}(\mathbf{x}^{(k-1)} - t_k \nabla D(M(\mathbf{x}^{(k-1)}), \mathbf{y}). \tag{2.9}$$

Note that both ∇D and $prox_{R,t_k}$ are well-defined from the assumptions defined before. One can interpret this algorithm as basically alternating/splitting the optimization between D and R; that is, in each iteration, a standard gradient descent step with respect to D is performed, then the result is "projected" onto the set of minima of R through the proximal operator. Therefore, this, and the aforementioned related algorithms, allow for the solution of inverse problems in the common case where the objective function is a sum of a smooth, convex function (the data consistency term) and a convex but non-differentiable function (the regularizer).

Setting the Regularization Parameter

The regularization parameter can have an enormous impact on the final solution; e.g., setting the regularization parameter of total variation too high can completely oversmooth the resulting image. However, in general, it is difficult to select the optimal regularization parameter without access to the ground truth, which is of course, usually unavailable. Below we list some

examples of strategies for setting the regularization parameter, with different strategies being appropriate depending on the amount of information available; for example, some strategies require information on the noise in the data. Heuristic strategies require no information other than the residuals during optimization.

- **Hope for Generalization**: Suppose we have a dataset where the ground truth is available; i.e., we have access to N pairs $(\mathbf{y_i}, \mathbf{x_i})$ for a given model M. Let $\lambda^i_{opt} = \arg\min_{\lambda'} \|\mathbf{x}^i_{\lambda'} - \mathbf{x_i}\|$, where $\mathbf{x}^i_{\lambda'}$ is the solution of Equation 2.6 using $\mathbf{y_i}, \lambda'$ as inputs. Then we can set $\lambda = \sum_i \frac{\lambda^i_{opt}}{N}$; that is, we set λ as the average over all pairs of the optimal regularization parameter which minimizes deviation to the ground truth. Then, one can hope that for similar datasets (for which no ground truth exists), that this parameter will also work well.

- **Morozov Discrepancy Principle [28]**: Suppose we have some knowledge of the noise level present in the measurement $\mathbf{y}$. That is, let $\mathbf{y}_T$ be the measurement with no noise; assume that we know that $\|\mathbf{y} - \mathbf{y}_T\|_2 \approx \sigma$. Let $\mathbf{x}_\lambda$ denote the solution of Equation 2.6 using regularization parameter λ. Then Morozov's discrepancy principle dicates that one should choose the largest possible λ such that $\|M(\mathbf{x}_\lambda) - \mathbf{y}\|_2 \approx \sigma$; i.e., to choose λ such that the discrepancy between the simulated measurement of the proposed solution and the noisy, measured data is the same as the known discrepancy between the noise-free and noisy, measured data. This principle can help prevent overfitting to the noise.

- **L-Curve Principle [29]**: In this method, one observes the 2 dimensional graph corresponding to $(\log(\|\mathbf{x}_\lambda\|), \log(\|M(\mathbf{x}_\lambda) - \mathbf{y}\|_2))$ as a function of λ. One then chooses the λ corresponding to the corner of this curve; i.e. where there is an abrupt shift. The motivation for this principle is that when λ is too high, the regularization term will dominate, and the norm of the residual will be high; in contrast, when λ is too low, then the data consistency term will dominate, and the noise instability will tend to make the norm of the proposed solution high, in comparison to well-regularized solutions. Therefore, the chosen λ should balance the residual norm and the norm of the proposed solution, corresponding to the point in the graph where one switches from one regime to the other. The name comes from the fact that the curve will roughly look like an L.

Probabilistic Formulation

We note that the Equation 2.6 also admits a probabilistic interpretation as a maximum a-posteriori (MAP) estimate; indeed, this is one method within the probablistic framework for solutions to inverse problems, a more complicated version of which is proposed in Chapter 5. As probablistic solutions are restricted to this chapter, we will explain this framework in detail there.

2.1.5 Caveats

We note that our discussion so far of inverse problems has a very modular, convenient form. That is, given some model M, we simply pick an appropriate regularization function(s), a way to select the regularization parameter, and an optimization algorithm which makes sense given the mathematical properties of M and R. Indeed, several popular software libraries for the computational solution of inverse problems (GlobalBioIm [30], ODL [31], Pycsou [32] etc.) more or less follow this modular route, in particular for inverse problems in imaging as described in the optimization section. However, while this abstract/modular framework works well for conceptual understanding and accessibility, we emphasize that we have omitted a great deal of the machinery/theory underlying this framework, including proofs of existence/-convergence, making precise/rigorous the notion of ill-posedness, the usual formulation of inverse problems with reference to Hilbert/Banach spaces as the domain for $\mathbf{x}$, error estimates when passing from continuous to discretized inverse problems, etc. We encourage the reader to peruse the supplied references for these important details.

2.2 Machine Learning for Inverse Problems

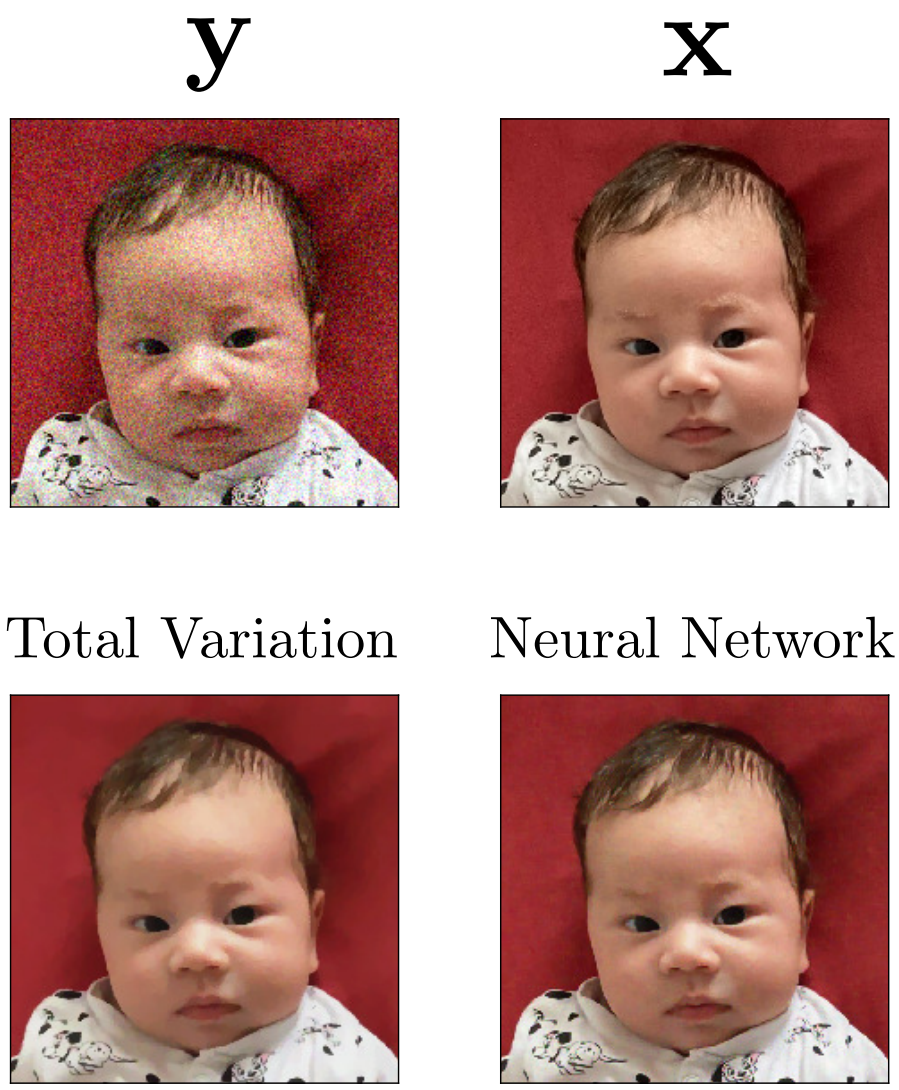


Figure 2.5: Here we show example reconstructions for denoising. In the top row, we show the noisy image (left) and the corresponding ground truth image (right). In the bottom row, we compare a reconstruction using total variation regularization (left) and a neural network reconstruction called DNCNN [33] (right). While the reconstruction quality is similar, note that the total variation reconstruction oversmooths considerably (see texture of the hair, skin, red background) in comparison to the neural network reconstruction, which looks more realistic. Furthermore, using the same CPU, the neural network reconstruction took 5 sec, while the total variation reconstruction took 5min. This example illustrates the power of using neural networks to solve inverse problems.

2.2.1 Motivation

Given an inverse problem with model M, we have outlined a flexible framework for solving it, the most important part being the selection of a regularization function which encodes prior knowledge on the solution, as all other choices hinge on this. However, thus far the regularization functions we have addressed have been **handcrafted**; i.e., manually constructed functions involving tractable operations (differentiation, scaling, change of basis) composed with tractable norms (L_1, L_2), such that they approximately impose our intuition/prior knowledge. This handcraftedness is also convenient for the subsequent optimization problems as, for example, the algorithms we outlined for optimizing problems with non-differentiable regularizers hinge on the regularizers having certain, nice mathematical properties/a tractable proximal operator in order to prove convergence/existence of solutions.

However, among others, handcrafted regularizers have three major problems:

- while the mathematical form of the regularizer may have been constructed to impose a given prior, this same form may also induce other, undesirable results,
- there is no adaptivity to the input data, since handcrafting implies that a given regularizer is the same for all input data, and
- the regularization parameter requires tuning for each problem instance.

To illustrate the first point, consider total variation regularization. While effective for denoising, it has been shown that the non-differentiability of the regularizer tends to cause staircasing [34]: i.e. favoring piece-wise constant, cartoonish reconstructions in highly oscillating regions of the ground truth. This implicit prior is undesirable in medical imaging, for example, as it does not reflect real images.

To illustrate the second point, consider compressed sensing regularization. In our example, we used sparsity in a wavelet basis i.e. sparsity of $W\mathbf{x}$ as the regularization. However, the wavelet basis is a handcrafted basis; the justification that natural images are sparse in the wavelet basis is empirical. This raises the question of whether the wavelet basis or, indeed, any given handcrafted basis is optimal for a particular set of input data, particularly because image content can vary widely. For instance, a priori, it is not clear whether a wavelet transform or, for example, a discrete cosine transform is better for reconstructing medical images vs. cellphone images.

To the third point, even if the solution of Eq. 2.6 can be found efficiently, tuning the regularization parameter could require the computation of many solutions corresponding to different values of the regularization parameter; in addition, many effective tuning methods require information on the measurement noise which may not be available.

There are of course remedies for these problems. Instead of total variation regularization, one can use another handcrafted regularizer called total generalized variation [35], which has

been shown to eliminate the staircasing effects. Instead of a handcrafted transform, one can consider compressed sensing with a learned sparsifying transform [36] i.e., learn from the input data the optimal basis in which the input data is most sparse while maintaining data consistency. However, in the first case, we have simply replaced one handcrafted regularizer for another, which could have other undesirable effects. In the latter case, the optimization becomes more difficult as now both $\mathbf{x}$ and the change of basis need to be optimized simultaneously, requiring more computation time as well as less theoretical assurance. Furthermore, in both cases, we are still only imposing a single prior. It is not clear whether a single prior is sufficient for the best possible signal reconstruction. It is possible to encode more/different priors by using multiple regularizers; however, depending on the number/nature of priors, this can become intractable for optimization. Furthermore, the regularization parameters of all priors need to be tuned as well.

For simplicity, let us consider the problem of denoising cell-phone images. The problem is that it is difficult/impossible to handcraft a regularizer which somehow perfectly captures the essence of what it means for a cell-phone image to be noise-free; to put it another way, total variation seems to capture a necessary condition (penalizing oscillations in the image gradient over the whole image which come from noise), but not a sufficient condition to be a denoised image, as the image content should not be affected, but can be by regularization. In addition, even if we add together multiple different handcrafted priors, it seems that capturing this ”perfect” prior is impossible.

Therefore, if possible, it would be desirable to construct a regularizer which enforces a stronger, more comprehensive prior, qualitatively different from simply using a single or multiple handcrafted priors. As such a regularizer most likely could not be applied universally due to the incredible diversity in the set of signals desirable for reconstructions, it is clear that it would have to be tailored or adapted for a specific set of data.

Learning a sparsifying transform for compressed sensing was a step toward realizing such a regularizer: using machine learning to solve inverse problems. Using machine learning, one can implicitly or explicitly learn a regularizer from the data itself. In recent years, machine learning approaches to solving inverse problems, particularly using neural networks/deep learning, have become the state of the art for the solution of a wide variety of inverse problems in computer vision and medical imaging (e.g. denoising, deblurring, dehazing, inpainting, super-resolution, image reconstruction from physical measurements, etc.). For an example in denoising, see Figure 2.5.

We begin with a brief introduction to artificial neural networks and deep learning:

2.2.2 Artificial Neural Networks and Deep Learning

Abstractly, artificial neural networks (ANNs) are parametrized functions whose parameters are tuned to approximate other functions; their name derives from the fact that their structure

was designed to mimic models of biological neural networks. Given sufficient data, they have become the state of the art for solving many problems (image classification, image segmentation, etc.) including the solution of inverse problems. Here we briefly describe two types of neural networks which are relevant for this thesis: multi-layer perceptrons [37], [38] and convolutional neural networks [39], [40]. Deep learning generally refers to artificial neural networks which are "deep", i.e. have multiple layers.

Multi-layer Perceptron

We will start by first defining the prototypical ANN: the multilayer perceptron. Concretely, fix an input domain $\mathbb{R}^n$ and an output domain $\mathbb{R}^m$; For an input $\mathbf{x} \in \mathbb{R}^n$, the mapping of the multilayer perceptron, $MP_\theta : \mathbb{R}^n \to \mathbb{R}^m$ corresponds to an iterated output of n functions or layers: $f_n \ldots (f_2(f_1(\mathbf{x}))\ldots)$. The layers $f_i : \mathbb{R}^p \to \mathbb{R}^q$ have the form

$$f_i(\mathbf{y}) = g(W_i\mathbf{y} + \mathbf{c}_i) \tag{2.10}$$

where W is a linear transformation $W : \mathbb{R}^p \to \mathbb{R}^q$, $\mathbf{c} \in \mathbb{R}^q$ is a bias vector, and $g : \mathbb{R}^q \to \mathbb{R}^q$ is a nonlinear function which is applied elementwise; e.g., the hyperbolic tangent. Note that the input dimension of f_1 and the output dimension of f_n are constrained to be $\mathbb{R}^n$ and $\mathbb{R}^m$ respectively due the desired input and output domains of the network; otherwise, all other input/output dimensions of intermediate spaces can be set freely. The parameters θ of the multi-layer perceptron are the set of linear transformations and bias vectors: $(W_i, \mathbf{c}_i)$. The structure of the multilayer perceptron was inspired by the hierarchical and iterative connections between neurons and the transmission of electrical impulses along neurons. Each component of the output of each layer can be described as a linear combination over **all** the input components followed by a nonlinearity; these layers are thus called **fully-connected** layers since each output component is connected to each input component. **Note that the use of the nonlinearity is crucial**; otherwise, the application of all layers could be reduced to a single linear transformation and bias vector addition.

Suppose one is given a dataset of pairs $(\mathbf{y}, \mathbf{x})$, where there is an underlying function h such that $h(\mathbf{y}) = \mathbf{x}$. Then multi-layer perceptrons can be used to **learn** the function h by solving for the optimal parameters as follows, for example

$$MP_\theta = \frac{1}{N} \arg\min_{\theta'} \sum_i \|\mathbf{x_i} - MP_{\theta'}(\mathbf{y_i})\|^2, \tag{2.11}$$

This optimization can be done efficiently using gradient or stochastic gradient descent efficiently through the backpropagation algorithm, which takes advantage of the iterative structure of the neural network layers for computing the derivatives with respect to the network parameters.

In the universal approximation theorem [41], [42], it was shown that a neural network of this type composed of a single layer followed by a linear output could approximate any continuous

and bounded function between finite dimensonal, real spaces to an arbitrary degree of error, provided that the output dimension of the single layer is large enough. **However we note that this gives no guarantee that any function can be learned by the optimization in Equation 2.11, merely that the expressive power of the neural network is sufficient to represent any function, given sufficiently wide layers.** In practice, the number of layers, the dimensionality of the inputs and outputs, and the type of nonlinear function used must be tuned for the specific mapping being learned.

Convolutional Neural Networks

We note that while multi-layer perceptrons could, in principle, be used for images and other higher dimensional signals, the size of the network and number of parameters of the resulting neural network can quickly become intractable for training, due to the large dimension of typical vectorized images. Furthermore, multi-layer perceptrons do not take advantage of structure in the data since the output of each neuron comes from the linear combination of all neurons in the previous layer.

For example, consider 2D, grayscale images as an input. Consider the output of a fully connected layer acting on a vectorized image; each output component will come from a linear combination over all the pixels in the 2D image, followed by element-wise application of a nonlinear function. In contrast, instead of generating outputs through linear combinations over all the pixels, we could generate outputs by convolving the image with a 2D kernel matrix followed by element-wise application of a nonlinear function. In this case, each output component of this so-called **convolutional layer** will come from a linear combination of the pixels in a localized patch (determined by the kernel size) of the 2D image, rather than all the pixels as with a fully connected layer. Furthermore, within a single convolutional layer, one can stack many different kernels which are independently convolved with the input image, such that the final output of the convolutional layer is a set of maps each coming a from different convolution. In this case, the learnable parameters of a convolutional layer are the kernels and (potentially) bias vectors. Convolutional layers have the advantage of much sparser connections (since outputs are only connected to localized patches of inputs) and take into account the structure of the input; in this example, natural images have plenty of local structure (e.g. locally, image patches tend to be similar) which can be exploited with a convolutional structure. Furthermore, generating maps by convolution with a kernel can be understood as extending traditional computer vision techniques for edge detection and analysis of local structure with linear filters.

Convolutional neural networks, as a whole, are neural networks which use different combinations/arrangements of convolutional layers to generate outputs; they can be trained for learning the mapping between training pairs as in Equation 2.11.

Practical Application

There is a bewildering large array of theoretical/practical concerns for which we have omitted a general discussion concerning the application of neural networks as they are outside the scope of this thesis; for example, algorithms for optimizing the neural network parameters, different topologies of neural networks used, strategies for determining the optimal structure of neural networks, etc. We address relevant concerns for practical application in each chapter of the thesis for the specific problem involved; for more general discussion, we point the reader to [5].

Application to Inverse Problems

Having described two types of neural networks relevant for this thesis, in the following subsection we broadly classify supervised, deep learning approaches to solving inverse problems into two categories: approaches which implicitly incorporate the model M, and approaches which explicitly incorporate the model M. In the following, given a model M, consider a dataset of N pairs $(\mathbf{y_i}, \mathbf{x_i})$, where $\mathbf{y_i}$ is the ith measurement sample and $\mathbf{x_i}$ the ith ground truth signal corresponding to $\mathbf{y_i}$. This dataset could be collected from real life or generated artificially.

2.2.3 Supervised Approaches which Implicitly Embed M

At the advent of deep learning in inverse problems, many approaches followed the following generic framework: given a generic neural network $f_\theta : \mathbb{R}^m \to \mathbb{R}^n$ with parameters θ, one can train the neural network to take as input $\mathbf{y}$ and output $\mathbf{x}$ by empirical risk minimization:

$$f_\theta = \arg\min_{\theta'} \sum_i \frac{1}{N} \mathbb{L}(\mathbf{x_i}, f_{\theta'}(\mathbf{y_i})), \tag{2.12}$$

where $\mathbb{L}$ is a metric which measures the difference between ground truth and predicted signals; e.g. $\mathbb{L}(\mathbf{x_i}, f_{\theta'}(\mathbf{y_i})) = \|\mathbf{x_i} - f_{\theta'}(\mathbf{y_i})\|_2^2$. Therefore, machine learning is used to learn an approximate inverse to M over the dataset, with the hope that this would generalize to other data. Examples include denoising [43] (mapping noisy images directly to noise-free images), super-resolution [44] (mapping low-resolution images directly to high-resolution images), and reconstruction of magnetic resonance images [45] (mapping Fourier measurements directly to contrast images),

Note that in practice, memory issues require training to be done in mini-batches of the dataset rather than all at once or reconstructing the signal in patches, etc. Furthermore, there are a variety of modifications that can be made: e.g. approaches input the adjoint reconstruction $M^T\mathbf{y_i}$ and learn the residual $\mathbf{x_i} - M^T\mathbf{y_i}$. Nevertheless, the main idea of learning the inverse from training data is captured by Equation 2.14. In some sense, f_θ has to implicitly learn both data consistency and regularization from the distribution of training data; in particular, the regularization is not handcrafted but learned by observing examples of high quality, ground truth signals. Here the model M is implicitly embedded in the training pairs.

2.2.4 Supervised Approaches which Explicitly Embed M

Particularly in medical imaging but also in computer vision, approaches to supervised learning for solving inverse problems were developed which explicitly embed the model M into the reconstruction pipeline. A generic framework for explicitly embedding M goes back to the variational framework for solving inverse problems introduced before. In essence, the handcrafted regularizer is replaced by the neural network; consequently, one can construct so-called **unrolled networks** which mimic the iterations of a generic, iterative optimization algorithm, albeit with steps involving the handcrafted regularizer replaced by a neural network. There are a wide variety of such approaches as any iterative algorithm can be modified in this way; here, we give a simple example using proximal gradient descent which illustrates the main idea. Consider T iterations of the proximal gradient descent from before; then one can let the kth iteration be defined by

$$\mathbf{x}^{(k)} = f_\theta^k(\mathbf{x}^{(k-1)} - t_k \nabla D(M(\mathbf{x}^{(k-1)}), \mathbf{y})). \tag{2.13}$$

where now instead of using the proximal operator of a handcrafted regularizer for projection, one uses a neural network, f_θ^k. Note that in this formulation, a different neural network is trained for each iteration; however, one can also use the same network for each iteration. The parameters of the (these) neural network regularizers can then be trained again through empirical risk minimization:

$$(\theta_1, \theta_2, \ldots, \theta_T) = \underset{\theta_1', \theta_2', \ldots, \theta_T'}{\arg\min} \frac{1}{N} \sum_i \mathbb{L}(\mathbf{x_i}, \mathbf{f}^T(\mathbf{y_i})) \tag{2.14}$$

where $\mathbf{f}^T(\mathbf{y_i})$ is the Tth iteration of Equation 2.13. As the model M is explicitly used in the reconstruction, data consistency is explicitly enforced. Furthermore, the neural network is more interpretable than in implicit approaches as it can really be identified as acting as a regularization function, albeit learned from the training data rather than handcrafted. We note, in passing, that this framework is also compatible with generative/probabilistic frameworks which use, neural networks to, roughly speaking, parametrize the prior probability distribution of $\mathbf{x}_i$. One can, for example, solve a maximum a-posteriori problem, i.e. optimize for the signal which maximizes a weighted sum of the prior probability and the data consistency; This devolves back to an iterative framework as well.

This type of approach has been used extensively in MRI [46]–[48]. However, we note that the emulation of an iterative, splitting approach has the advantage that the neural network component is separate from the data consistency step; therefore, for image restoration tasks such as denoising, deblurring, super-resolution, etc. where the same regularization is desired, i.e. regularization that encodes being a "high quality", natural image, one can learn a single regularizer (over natural images) and use this for solving multiple different inverse problems. This kind of approach has been used in [49], [50], where the authors used a single regularizer for image deblurring, denoising, etc.

2.2.5 Comparison of Supervised Approaches

Having broadly outlined different approaches to using machine learning for solving inverse problems, we note that machine learning approaches which implicitly and explicitly embed M have their advantages and disadvantages.

While implicit approaches were dominant for a period of time, there are immediate problems with generalizability/robustness. First, there is no enforcement/guarantee that the resulting prediction is consistent with the model/input data. Second, as the model is only implicitly embedded in the training pairs, data generated from a slight variation of the model could result in a completely different solution. For example, a network trained to deblur images blurred by a particular blur kernel could fail completely when presented with images blurred by a slightly different blur kernel. Third, depending on the complexity of the task, very large datasets of training pairs could be required for training, which is not always feasible to acquire, for example, in medical imaging. In addition, implicit methods can require an extremely large number of parameters in the network as it must learn both the model and the regularizer. In contrast, explicit approaches are more robust to variations in M due to explicitly enforcing data consistency, and the decoupling between data consistency and regularization. In addition, explicit approaches generally require much training data in comparison to implicit methods, again due to the added information from explicitly including the model. Finally, as explicit methods include the data consistency, they tend to require smaller networks than in implicit methods.

However, implicit approaches, at inference, generally require a single forward pass through the trained network. Furthermore, while they require more training data, they are still effective if the model M is unknown, difficult to compute analytically, or difficult to integrate into an optimization framework; in contrast, explicit approaches can be impossible or infeasible when faced with these issues. Furthermore, if the model M is a poor approximation to the true, underlying model, then this can cause biased solutions. In addition, explicit approaches generally require more computation time at inference than implicit approaches as they iteratively apply data consistency and neural network steps.

2.2.6 Comparison of Machine Learning and Traditional Approaches

In many cases, machine learning approaches to solving inverse problems outperform traditional solutions using the variational framework on virtually all dimensions: accuracy, inference speed, etc. In principle, this is because machine learning solutions can learn a similar or better prior than any single or combination of handcrafted regularizer; furthermore, implicit approaches can produce solutions, once trained, orders of magnitude faster than traditional approaches which require the use of iterative algorithms. However, with machine learning, one still has a variety of impactful choices to make in terms of practical application, just as in traditional approaches. One must select:

- what kind of neural network to use/ what kind of network architecture,

- what optimization algorithm is used to solve Equation 2.14, and

- how to tune the hyperparameters of the selected network

Furthermore, a big disadvantage of machine learning approaches compared to traditional approaches is the general lack of proofs of convergence or stability estimates, in particular for supervised approaches which implicitly embed M. However, for supervised approaches which explicitly embed M through unrolled networks, there is a growing literature which theoretically analyze convergence and stability [51]–[53]; nonetheless, these analyses depend on the specific network architecture/unrolled algorithm used.

2.2.7 Our Approach

Having broadly outlined the evolution of solutions to inverse problems, from variational optimization to supervised machine learning, we briefly outline and motivate the approaches to inverse problems examined in this thesis. **In the supervised paradigm, broadly speaking, the problem is that the training dataset should be realistic and of sufficient quantity for learning the desired inverse mapping.** However, in the inverse problems addressed in this thesis, realistic training datasets are either unavailable or scarce. Therefore, we consider two broad approaches to address this issue: self-supervised learning and realistic modelling.

Self-Supervised Learning

In self-supervised approaches to inverse problems, machine learning is still used, but no ground truth is required: only the measurement data $\mathbf{y_i}$ and the model M are required. These being more specialized to each inverse problem, we defer their description to later chapters of the thesis, where we propose (Chapter 5) and validate (Chapter 4) different self-supervised approaches.

Integration of Realistic Training Data through Realistic Modelling

In Chapters 6 and 7, we integrate realistic training data into supervised methods, in inverse problems for which realistic training data is nonexistent or scarce, by either synthetically generating a large realistic dataset by combining realistic models with realistic priors on desirable solutions or leveraging a small quantity of realistic training data in conjunction with a large amount of unrealistic training data.

2.3 Closing Remarks

In this section, we engaged in a whirlwind tour of inverse problems and their solutions. We introduced the traditional variational framework as well as machine learning frameworks for solving inverse problems; in doing so, we tried to stay in as generic a setting as possible to show the flexibility/wide applicability of such approaches, while showing specific examples for visualization and understanding. However, two aspects of inverse problems are difficult to discuss in a generic way: **what it means for the training data for a model M to be realistic and how to rigorously validate algorithms for solving inverse problems with a model M.** This is because, of course, models used in inverse problems can be wildly different; in this thesis alone, the models used range from the effect of changing the focal length of a digital camera to sampling the magnetic moments of hydrogen atoms in the brain. Furthermore, validation of algorithms depends on the use case of the resulting solutions. As the reader will see, these crucial aspects are central to the contributions of this thesis.

3 Introduction to Magnetic Resonance Imaging

In this chapter, we give a quick overview of Magnetic Resonance Imaging (MRI) in order to give context for the models/inverse problems introduced in the following chapters. MRI is one of the foundational modalities in medical imaging, enabling detailed and high resolution imaging of the human anatomy without using ionizing radiation. This overview is largely adapted from [3]. Most of the figures were inspired by analogous figures in [3].

3.1 Nuclear Magnetic Resonance for a Single Spin

The basic physics underlying MRI is the phenomenon of nuclear magnetic resonance (NMR), in which nuclei, exposed to a constant magnetic field, can absorb radio-frequency (RF) energy when the RF pulse has a specific frequency related to the magnetic field and properties of the nuclei itself. This process is not unlike, for example, the resonance that occurs when one pushes someone on a swing at the right frequency. **In MRI, one primarily probes the magnetic resonance of the protons of the constituent hydrogen atoms of water molecules.** This section will explain this phenomenon from a semi-classical picture.

3.1.1 Magnetic Moment

From quantum mechanics, we know that the angular momentum of nuclei are a sum of two terms: the orbital angular momentum (angular momentum related to the orbital motion of the nuclei) and the spin angular momentum (angular momentum analogous to a body rotating about itself). A macroscopic example of this is a planet rotating on its own axis (spin angular momentum) which is also in orbit (orbital angular momentum) (see Figure 3.1). However, the spin angular momentum of nuclei is a purely quantum phenomenon; while the name is inspired by thinking of the nuclei as spinning on its own axis, it is purely an intrinsic property of the nuclei.

The fact that elementary particles have an intrinsic, quantized magnetic moment/spin angular momentum was first discovered in the Stern-Gerlach experiment [54], [55] in 1922, which

showed, by sending a beam of silver atoms through an inhomogeneous magnetic field, that the silver atoms split into two distinct beams, rather than a continuous fan of beams; this implied that the outer electrons of silver atoms had to have a quantized, intrinsic angular momentum, later called spin.

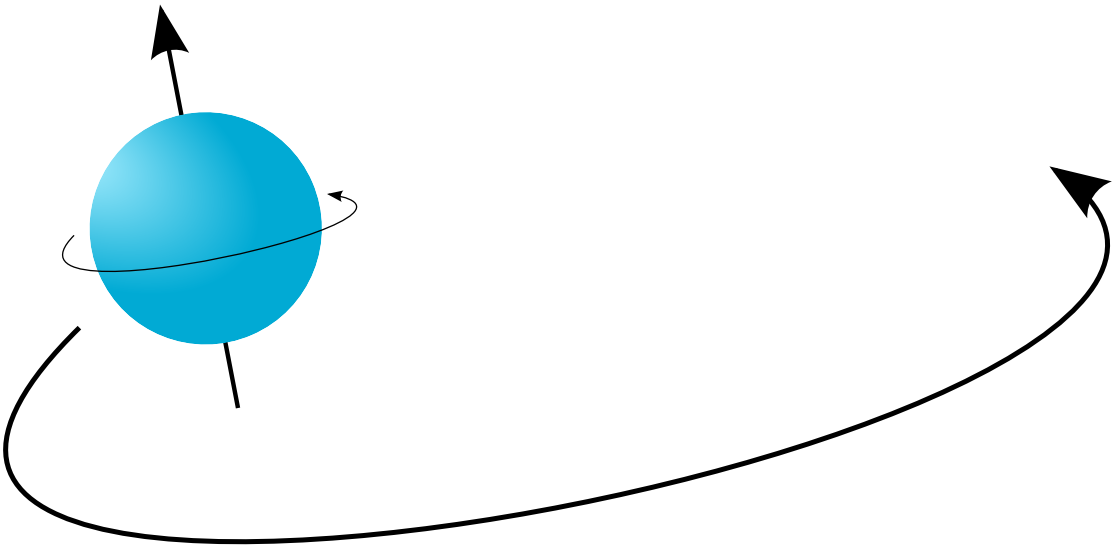


Figure 3.1: Here is an example of a macroscopic system with spin and orbital angular momentum: the ball is spinning around its own axis and also moving in a circular path. Quantum mechanics shows that this picture also applies to microscopic systems, e.g. nuclei, albeit the spin angular momentum derives from inherent properties of the nuclei rather than physical spinning.

This intrinsic spin angular momentum leads to an intrinsic magnetic moment for elementary particles with mass/electric charge which can be motivated by thinking of the charged particle as an infinitesmally small current loop (i.e. a charged particle spinning).

To see the interaction between the magnetic moment of a particle and a fixed magnetic field, consider a simple loop of current, with current I in a homogeneous, invariant magnetic field $\mathbf{B}$ in the z direction. From the Lorentz force law, we have that the differential of the force exerted on a point of the loop due to the magnetic field is

$$d\mathbf{F} = I d\mathbf{l} \times \mathbf{B} \tag{3.1}$$

The differential torque exerted by a differential force is

$$d\tau = \mathbf{r} \times d\mathbf{F} \tag{3.2}$$

Now consider two cases in Fig. 3.2, where the normal of a current loop is parallel to the magnetic field and at an angle. While there is no net torque in the first case, in the second case, from the above equations, we can see there is a net torque which rotates the loop into the x-y plane. If one take the cross product of the unit normals of the loops in both cases with the

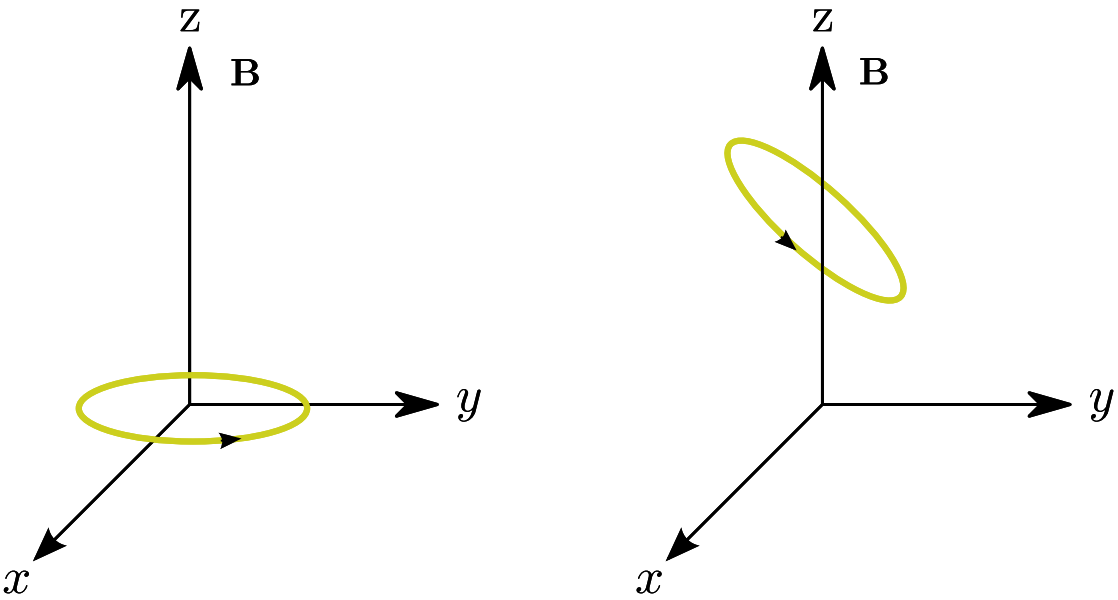


Figure 3.2: Here are two situations: to the left, a current loop whose unit normal is parallel to the magnetic field. To the right, a current loop whose unit normal is at an angle with respect to the magnetic field. Basic force calculations show that in the case to the right, there is a torque exerted on the current loop that pushes the unit normal of the current loop to align with the magnetic field. We will see later that this phenomenon also occur with nuclei in a magnetic field.

magnetic field, one can see that this is proportional to the torque from calculation. In fact, this motivates defining the magnetic moment of a current loop μ in relation to the torque by

$$\tau = \mu \times \mathbf{B}. \tag{3.3}$$

3.1.2 Magnetic Moment in a Static Field

By definition, the angular momentum **J** obeys

$$\frac{\mathrm{d}\mathbf{J}}{\mathrm{d}t} = \tau. \tag{3.4}$$

One can show that the spin angular momentum is proportional to the magnetic moment by a factor called the gyromagnetic ratio, γ, which is different for each particle

$$\mu = \gamma \mathbf{J}. \tag{3.5}$$

Combining these equations, we see that

$$\frac{\mathrm{d}\mu}{\mathrm{d}t} = \gamma \mu \times \mathbf{B}. \tag{3.6}$$

Equation 3.6 gives the time evolution of the magnetic moment of a single particle exposed to a constant magnetic field B.

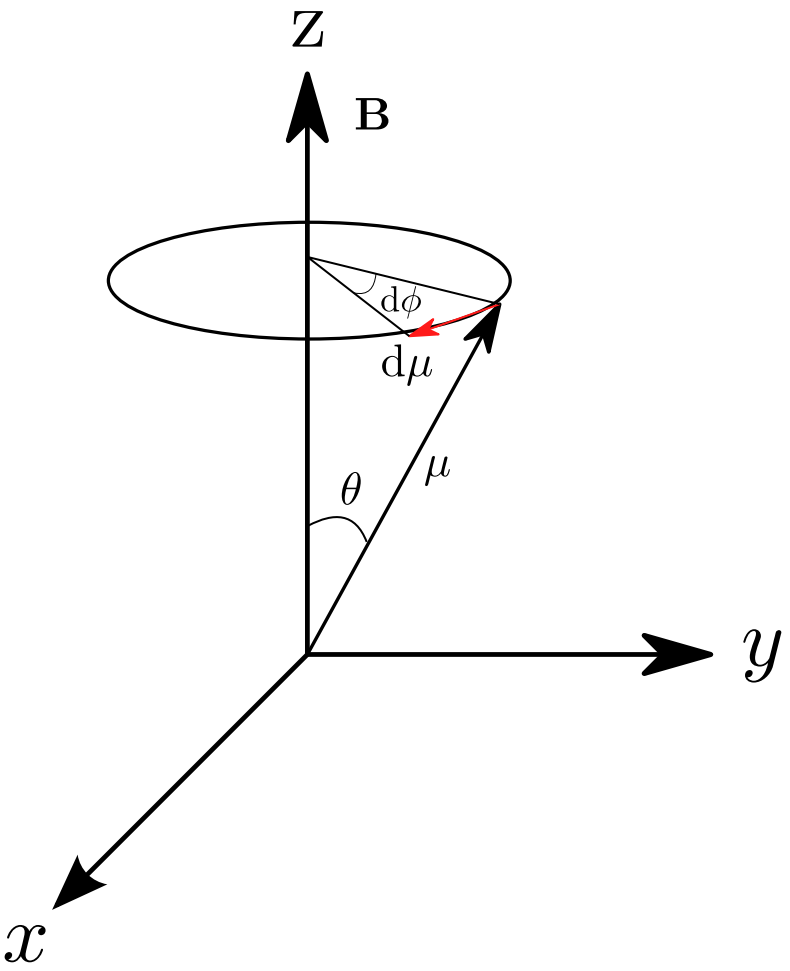


Figure 3.3: Here we show the geometry of the evolution of the magnetic moment μ in a magnetic field. In red, we show the differential $d\mu$ which can be derived from the form of the time evolution equations. By expressing the magnitude of the differential in two ways, both derived from the geometry of the system, we can derive that the magnetic moment precesses around the magnetic field at a rate depending on the magnitude of the magnetic field, called the Larmor Frequency.

While Equation 3.6 is far from complete for a realistic description of the magnetic moment, we can already derive an important concept: precession at the Larmor frequency. Consider a magnetic field, **B** in the z direction, and a magnetic moment μ at an angle θ with respect to **B**; see Figure 3.3 for the geometry. The magnetic moment will precess about the magnetic field instantaneously at a frequency called the Larmor Frequency. We can derive this geometrically by first noting that the differential of μ must lie in a plane parallel to the $x-y$ plane;

furthermore, as $\frac{\mathrm{d}\mu}{\mathrm{d}t}$ is always perpendicular to μ, this implies that the tip of μ will trace a circle.

$$\left\|\mathrm{d}\mu\right\| = \gamma \left\|\mu\right\| \|\mathbf{B}\| \sin\theta \mathrm{d}t \tag{3.7}$$

Now, let $\mathrm{d}\phi$ denote the differential angle subtended by the change in the magnetic moment on the circle on which the tip of the magnetic moment rotates. Then geometrically (using that the arc length subtended by an angle is equal to the radius multiplied by the angle in radians)

$$\left\|\mathrm{d}\mu\right\| = \left\|\mu\right\| \sin\theta \left\|\mathrm{d}\phi\right\|. \tag{3.8}$$

Using these two equations for $\left\|\mathrm{d}\mu\right\|$, we have that

$$\left\| \frac{\mathrm{d}\phi}{\mathrm{d}t} \right\| = \gamma \|\mathbf{B}\| = \omega_0. \tag{3.9}$$

Therefore μ precesses around $\mathbf{B}$ with frequency ω_0, which is called the Larmor Frequency.

3.1.3 Magnetic Moment in a Static Field with an RF Perturbation

We now consider a static background magnetic field perturbed by a time varying magnetic field orthogonal to the background magnetic field. Let the static field lie in the z direction, and let us use a reference frame rotating at the Larmor frequency clockwise around the z axis (See top row of Figure 3.4).

We note that that given a rotating reference frame with rotation vector Ω, the time rate of change in the fixed reference frame of a vector function $\mathbf{V}(t)$ is given by

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t}_{RotFrame} + \Omega \times \mathbf{V}. \tag{3.10}$$

Using this expression for the magnetic moment and comparing it to the already known differential equation for the magnetic moment in a static field, we have that

$$\frac{\mathrm{d}\mu}{\mathrm{d}t}_{RotFrame} = \gamma\mu \times \mathbf{B}_{eff}, \tag{3.11}$$

$$\mathbf{B}_{eff} = \mathbf{B} + \frac{\Omega}{\gamma}. \tag{3.12}$$

Hence, the motion of the magnetic moment with respect to the rotating frame is the same as in the lab frame except with a different, effective magnetic field.

Now, let ω be the frequency of an RF perturbation we will use to push the magnetic moment away from the orientation of the static field. Note that from the differential equation, this pulse must have components in the x or y direction. As our rotating reference frame, let

$$\Omega = -\omega\hat{z}. \tag{3.13}$$

Note that if we have a circularly polarized RF perturbation

$$\mathbf{B}_{circ} = B_1(\hat{x}\cos\omega t + -\hat{y}\sin\omega t), \tag{3.14}$$

then in the rotating frame this just becomes

$$\mathbf{B}_{circ} = B_1\hat{x}'. \tag{3.15}$$

Now we write the equations of motion in a rotating reference frame with the constant field and the circularly polarized field, where the rotation is at the RF frequency, ω

$$\frac{\mathrm{d}\mu}{\mathrm{d}t}_{RotFrame} = \mu \times (\hat{z}(\omega_0 - \omega) + \hat{x}'\omega_1), \tag{3.16}$$

where

$$\omega_1 = \gamma B_1 \tag{3.17}$$

is the precession frequency caused by the rf field. If the RF frequency matches the Larmor frequency, then the above equation simplifies to

$$\frac{\mathrm{d}\mu}{\mathrm{d}t}_{RotFrame} = \omega_1\mu \times \hat{x}'. \tag{3.18}$$

This implies that in the rotating reference frame, the magnetic moment will precess about the $\hat{x}'$ axis only (See bottom row of Figure 3.4). **This frequency matching is called the resonance condition, as matching this flips the spin to the new axis maximally. In essence, what has been demonstrated is that an RF perturbation applied at the resonant frequency (i.e. the Larmor Frequency) produces an effective magnetic field, in the rotating frame, solely in the $\hat{x}'$ direction, instead of the $\hat{z}$ direction. This makes the magnetic moment flip and precess around a different axis. Altogether this phenomenon is called nuclear magnetic resonance.** Nuclear magnetic resonance was first demonstrated by Isidor Isaac Rabi [56] in 1939, who used it to measure the magnetic moment/spin of atoms in molecular beams.

Suppose we turn on the RF perturbation for time τ then turn it off; this is called an RF pulse. In that case, the spin will rotate

$$\Delta\theta = \gamma B_1\tau. \tag{3.19}$$

Note that explicit solutions in the resonance case for a series of pulses can be written as the product of a set of rotation matrices applied to the initial magnetic moment. This is because of the form of the differential equation in the rotating frame.

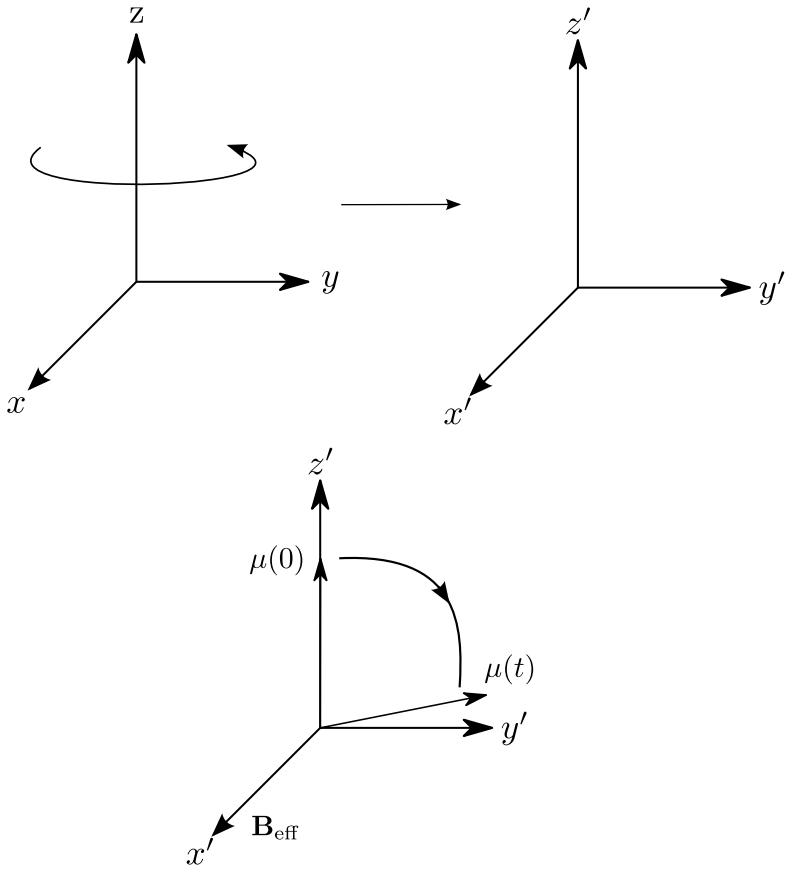


Figure 3.4: In the top row, we show the rotating reference frame (rotation about the z axis); note that while the z axis remains the same in both frames, the unit vectors of the transverse plane are different. In the bottom row, we show how the imposition of a RF perturbation at the Larmor Frequency can cause the magnetic moment to flip to a different axis in the rotating frame.

3.1.4 Quantum Note

In the preceding subsection, we derived the basics of Nuclear Magnetic Resonance in a semi-classical way, starting from the existence of an intrinsic spin/magnetic moment due to quantum mechanics and continuing with the classical equations for the time evolution of a magnetic moment in a constant magnetic field and a time varying magnetic field. A rigorous derivation of NMR should be grounded in a quantum mechanical calculation, starting from the Hamiltonian corresponding to the magnetic moment in a magnetic field/the Schrodinger Equation and calculating the expectation value of the spin operators i.e. the magnetic moment, up to the gyromagnetic ratio, in each spatial dimension. We note that this calculation results in the same answer as in the semi-classical case.

We can give a handwaving intuition for this by noting that as spin is a quantum phenomenon, it is **quantized**; it can only take a certain set of discrete values, depending on the particle. In MRI, we are primarily interested in the spins of the hydrogen protons of water; protons are spin

$\frac{1}{2}$ particles, meaning there are only 2 different spin states. The addition of an external magnetic field in a fixed direction causes splitting in the discrete energy levels of the proton, as the spin state parallel with the field has a lower energy than the anti-parallel spin state. One can show that the energy difference between these two states is $\hbar\omega_0$, where $\hbar$ is the reduced Planck's constant and ω_0 is precisely the Larmor frequency associated to the proton and the external magnetic field. Therefore transitions between spin states can only be induced by the addition or subtraction of this amount of energy by, for example, photons with a frequency equal to the Larmor frequency, corresponding to the the resonance condition on the RF perturbation previously derived semi-classically.

3.2 Nuclear Magnetic Resonance for an Ensemble of Spins

3.2.1 Net Magnetization

In the previous section, we considered only a single spin; however, we need to consider macroscopic bodies composed of many spins, as is the case in MRI. We deal with this by defining the net magnetization **M**:

$$\mathbf{M} = \frac{1}{V} \sum_{protons} \mu_i, \tag{3.20}$$

where V is a volume small enough such that external fields are constant over it, but large enough to contain many particles/spins. Nuclear magnetic resonance in bulk matter/solids, as will be examined here, was first demonstrated concurrently and separately by two different groups in 1946: the group of Felix Bloch [57] and the group of Edward Purcell [58].

Suppose that we have an ensemble of protons in a static magnetic field with strength B_0 at a temperature T (See Figure 3.5). As previously noted, the static field induces a splitting in the energy levels of the protons due to an energetically preferred spin state. This is because the potential energy of a magnetic moment in a magnetic field is

$$U = -\mu \cdot \mathbf{B}. \tag{3.21}$$

As there are two possible spin states, the ensemble of spins will tend toward the favored spin state; however, due to thermal interactions/interactions between spins, the spins **do not all** respond as in the previous sections. This can be made precise by calculating the average magnetization through the Boltzmann distribution of the ensemble of spins, as each spin state corresponds to a different energy; with no static field, the net magnetization should be zero, as the sum of the spins will average to 0. Letting M_0 denote the magnitude of the average magnetization, one can derive the approximation

$$M_0 \approx \frac{1}{4}\rho_0 \frac{\gamma^2\hbar^2}{kT} B_0, \tag{3.22}$$

where k is Boltzmann's constant and ρ is the volume density of spins. Therefore, a static magnetic field induces a non-trivial net magnetization.

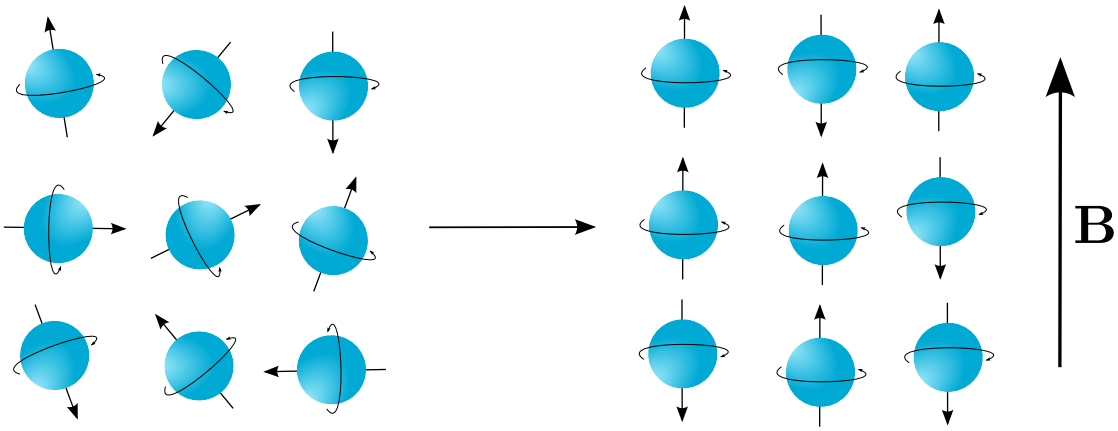


Figure 3.5: To the left is an ensemble of spins with no external magnetic field; in this case, no spin state is energetically favored, so there is no net magnetization as the spins will be randomly distributed. To the right is an ensemble of spins with an external magnetic field; this induces a favorable spin state aligned with the magnetic field. Note that there are spins which are anti-parallel, as thermal interactions/other factors can cause spin transitions from the energetically favored state. However, as there are more spins parallel than anti-parallel, this ensemble will produce a net magnetic moment.

If we now sum over each spin and take an average of the spin evolution equations for each spin, we have

$$\frac{d\mathbf{M}}{dt} = \gamma \mathbf{M} \times \mathbf{B}. \tag{3.23}$$

However, this equation also needs to be modified to take into account so-called **relaxation effects** as we now consider many spins, which can interact with each other as well as the surrounding environment.

3.2.2 Bloch Equation

In the following discussion, it will be helpful to separate **M** into its components parallel and perpendicular to the static, external field. Previously, while we derived the equations of motion for the magnetic moment, we did not discuss the initial conditions, i.e. the initial state of the magnetic moment. As previously stated, a static field induces an equilibrium, net magnetization aligned with the static field, due to the fact that magnetic moments aligned with the static field are energetically favored.

If an RF pulse pushes the magnetization away from equilibrium, then we see that the parallel component will change. Note that in the previously derived equation for the rate of change of

the magnetization, we have that

$$\frac{\mathrm{d}\mathbf{M}_{\parallel}}{\mathrm{d}t} = 0, \tag{3.24}$$

since the cross product has zero magnitude in the parallel direction. To model the return to equilibrium, which is mediated by thermal exchange with the lattice of surrounding spins, we can replace this equation with

$$\frac{\mathrm{d}\mathbf{M}_{\parallel}}{\mathrm{d}t} = \frac{M_0 - M_{\parallel}}{T_1}. \tag{3.25}$$

Note that this equation implies that $\mathbf{M}_{\parallel}$ exponentially approaches M_0, the equilibrium magnetization. T_1, the characteristic time of this asymptotic approach, is called the spin-lattice relaxation time.

Just as the spin-lattice interactions causes the longitudinal magnetization to tend towards the equilibrium magnetization, interactions between spins causes transverse magnetizations to decay. The transverse magnetization is the average over many transverse spins. Variations in local fields, interactions between the magnetic fields of other spins, etc, cause individual spins to have different precession frequencies, as they experience different local magnetic fields. This causes them to dephase/decohere/spread out relative to each other. Note that unlike the parallel case, there is no energy favorable condition restoring the transverse spins to any set state. Hence, the magnetization tends to zero since it becomes an average of many moments pointing in random directions. We modify the equations for the transverse magnetization by adding a term for this decay

$$\frac{\mathrm{d}\mathbf{M}_{\perp}}{\mathrm{d}t} = \gamma\mathbf{M}_{\perp} \times \mathbf{B} - \frac{\mathbf{M}_{\perp}}{T_2}. \tag{3.26}$$

Combining all these corrections into a single equation, we have the Bloch equation:

$$\frac{\mathrm{d}\mathbf{M}}{\mathrm{d}t} = \gamma\mathbf{M} \times \mathbf{B} + \frac{M_0 - M_{\parallel}}{T_1}\hat{z} - \frac{\mathbf{M}_{\perp}}{T_2}. \tag{3.27}$$

This equation gives the time evolution of the net magnetization, including the phenomenological relaxation effects.

3.2.3 Bloch-Torrey Equation

While the Bloch equation captures the time evolution of the net magnetization due to the effects of an external magnetic field and relaxation effects, it does not capture time evolution from the physical movement of the particles with spin. That is, if we consider a concentrated population of **fixed** spins which diffuse to spread out over a spatial domain, then there will still be a time evolution of the magnetization due to this diffusion, which has no relation to the

time evolution in the Bloch equation. A simple way to extend the Bloch equation to account for this is to add a diffusion term, resulting in the Bloch-Torrey Equation:

$$\frac{\partial \mathbf{M}}{\partial t} = \gamma \mathbf{M} \times \mathbf{B} + \frac{M_0 - M_\parallel}{T_1}\hat{z} - \frac{\mathbf{M}_\perp}{T_2} + \nabla \cdot (\mathbf{D}\nabla \mathbf{M}) \tag{3.28}$$

where **D** is the diffusion tensor (3x3 matrix), encoding the diffusivity in different directions.

3.3 Signal Detection

In the preceding sections, we showed how exposing a sample of magnetic nuclei to a constant magnetic field and an RF pulse can cause its net magnetization to change in a way predicted by the Bloch Equation. MRI is based on detecting these changes in net magnetization. Here we present a simplified model of signal detection in MRI. Signals from MRI come from an RF coil near the body of the object being probed and measuring the resulting Faraday induction in the coil from the changing magnetization of the body induced by the combination of a static magnetic field and an RF perturbation, as previously described. There is an effective current associated with the magnetization

$$\mathbf{J}_\mathbf{M}(\mathbf{r}, t) = \nabla \times \mathbf{M}(\mathbf{r}, t). \tag{3.29}$$

This effective current results in a magnetic field. From the formulas for magnetic fields, vector potentials, etc, we can write the flux through the recieve coil as a volume integral over the sample depending on the magnetization and the induced magnetic field per unit current in the recieve coil

$$\Phi_M(t) = \int_{sample} \mathrm{d}^3 r \mathbf{B}^{receive}(r) \cdot \mathbf{M}(r, t). \tag{3.30}$$

Hence,

$$s(t) = \frac{\mathrm{d}}{\mathrm{d}t} \int_{sample} \mathrm{d}^3 r \mathbf{B}^{receive}(r) \cdot \mathbf{M}(r, t) \tag{3.31}$$

$s(t)$ which is the electromotive force in the coil, is the measured signal for MRI images.

From the Bloch equation we can write the following general solution:

$$M_z(\mathbf{r}, t) = \exp(\frac{-t}{T_1(\mathbf{r})}) M_z(\mathbf{r}, 0) + (1 - \exp(\frac{-t}{T_1(\mathbf{r})})) M_0 \tag{3.32}$$

$$M_+(\mathbf{r}, t) = \exp(\frac{-t}{T_2(\mathbf{r})}) \exp(-i\omega_0 t + i\phi_0(\mathbf{r})) M_\perp(\mathbf{r}, 0), \tag{3.33}$$

where M_+ is the complex representation of the transverse magnetization; $M_x = Re(M_+), M_y = Im(M_+)$. ϕ_0 is the initial phase distribution of the transverse magnetization. Inserting these into Equation 3.31, decomposing $\mathbf{B}_x^{receive}(r) = \mathbf{B}_\perp \cos(\theta_B(r)), \mathbf{B}_y^{receive}(r) = \mathbf{B}_\perp \sin(\theta_B(r))$, and

assuming that the Larmor Frequency, ω_0, dominates the reciprocals of T_1, T_2, we have that

$$s(t) \approx \omega_0 \int \mathrm{d}^3 r \exp(\frac{-t}{T_2(r)}) \mathbf{M}_\perp(r,0) \mathbf{B}_\perp(r) \sin(\omega_0 t + \theta_B(r) - \phi_0(r)). \tag{3.34}$$

In this case, the signal is dominated by the oscillations in the transverse magnetization; **parallel magnetization contributes negligibly to the measured signal.** Note that this approximation does not take into account variations of the constant magnetic field on the z-axis nor time dependence of additional magnetic fields, although these are easily accomodated.

The resulting signal is usually demodulated (multiplied by sin and cosine with a certain reference frequency) then low pass filtered. This is done in order to get rid of the high frequency oscillations at the Larmor frequency ($\sin(\omega_0 t + \theta_B(r) - \phi_0(r))$) and replace it with low frequency oscillations at the offset. This demodulation results in two channels, called real and imaginary, where it is convenient to define a complex demodulated signal

$$s(t) = s_r(t) + i s_i(t). \tag{3.35}$$

The complex, demodulated signal turns Equation 3.34 into

$$s(t) \approx \omega_0 \int \mathrm{d}^3 r \exp(\frac{-t}{T_2(r)}) \mathbf{M}_\perp(r,0) \mathbf{B}_\perp(r) \exp(i((\Omega - \omega_0) t - \theta_B(r) + \phi_0(r))), \tag{3.36}$$

where Ω is the reference freq; note that if $\Omega = \omega_0$ we eliminate the high frequency oscillations of the signal.

3.4 Basic Imaging Model

In the previous section, we showed how we can measure the signal from the changing magnetization of a sample. However, as the name denotes, in MRI, the goal is to produce an image, the most basic of which is the spatial distribution of the spin density associated with the hydrogen proton. In fact, as we will show in this section, the key to producing an image is the application of a spatially varying magnetic field (also called a field gradient) on top of the static, main magnetic field. This idea of using magnetic field gradients, while introduced previously, was first used to produce an image by Paul Lauterbur [59] in 1973.

Here we consider a very simplified model of how to produce such an image. We assume that the initial phases, the phase of the receive field and the amplitude of the receive field are all independent of position, and thus can be ignored or absorbed into an overall constant. Furthermore, we neglect relaxation effects. Then the complex demodulated signal from the previous section can be written as

$$s(t) \approx \omega_0 \mathbf{B}_\perp \int \mathrm{d}^3 r M_\perp(r,0) \exp(i(\Omega t + \phi(r,t))), \tag{3.37}$$

where ϕ is the accumulated phase

$$\phi(r,t) = -\int_0^t \omega(r,t)\mathrm{d}t, \tag{3.38}$$

and $\omega(r,t)$ is the precession frequency. We note that this differs from Equation 3.36 in that we now assume that the precession frequency can change spatially/temporally; we assume that the spatial and time dependence of the precession frequency comes only from the spatial and temporal changes in the magnetic field; otherwise, it remains the precession frequency from the main magnetic field.

Let there be a $\frac{\pi}{2}$ pulse (i.e. an RF pulse which rotates the net magnetization from the z axis to the transverse plane). Then the initial perpendicular magnetization will be the equilibrium magnetization, M_0 which can be expressed in terms of the volume spin density as well as other factors. Then we can write

$$s(t) = \int \mathrm{d}^3\rho(r)\exp\left(i(\Omega t + \phi(r,t))\right), \tag{3.39}$$

where $\rho(r)$ is an effective spin density. We say effective because they are proportional to the volume spin density, but contain many multiplicative factors depending on temperature, freq, etc,

$$\rho(r) = \omega_0 B_\perp M_0(r) \tag{3.40}$$

Consider the one dimensional case where the phase depends only on z. Then

$$s(t) = \int \mathrm{d}z\rho(z)\exp(i(\Omega t + \phi(z,t))) \tag{3.41}$$

$$\rho(z) = \int\int \mathrm{d}x\mathrm{d}y\rho(r) \tag{3.42}$$

To obtain a simple image, we want to determine the spin density $\rho(z)$. Let us add a linearly varying magnetic field in z such that

$$B_z(z,t) = B_0 + zG(t). \tag{3.43}$$

Then the precession frequencies of the spins also become time/spatially dependent

$$\omega(z,t) = \omega_0 + \gamma z G(t) = \omega_0 + \omega_G(z,t). \tag{3.44}$$

Using a magnetic field gradient to establish a relationship between the position of the spins (z coordinate) and the precessional frequency is called frequency encoding. This is a cornerstone of MRI. Note that the gradient is added only after the pulse.

After setting the reference frequency of demodulation to the Larmor frequency due to the

static field, we have

$$s(t) = \int \mathrm{d}z \rho(z) \exp(i\phi_G(z,t)), \tag{3.45}$$

$$\phi_G(z,t) = -\gamma z \int_0^t \omega_G(z,t')\mathrm{t}', \tag{3.46}$$

where $\phi_G(z,t)$ is the associated accumulated phase with the additional magnetic field from G.

This is called the 1D imaging equation. Let

$$k(t) = \frac{\gamma}{2\pi} \int_0^t \mathrm{d}t' G(t'). \tag{3.47}$$

Then we can write the signal in the suggestive form

$$s(k) = \int \mathrm{d}z \rho(z) \exp(-2i\pi kz) \tag{3.48}$$

This shows that the measured signal is the Fourier transform of the spin density. We say that the spin density is Fourier encoded along z. In principle, we can reconstruct the spin density by applying the inverse Fourier transform to the signal. In practice, we need to sample many different values of k (through application of the magnetic field gradient), measure the corresponding signal, and apply a discrete Fourier transform. Furthermore, sampling many k values can be difficult since relaxation effects, among other things, destroy the transverse signal over time. In MR, the space of Fourier measurements is called the **k-space**.

To generalize to 3 dimensions one can define

$$\mathbf{G}(t) = G_x(t)\hat{x} + G_y(t)\hat{y} + G_z(t)\hat{z}. \tag{3.49}$$

where $\mathbf{G}(t)$ is now a vector. Then the z component of the magnetic field is

$$B_z(\mathbf{r},t) = B_0 + G(t)\cdot\mathbf{r}. \tag{3.50}$$

We then have

$$s(t) = \int \mathrm{d}^3 r \rho(r) \exp(-i2\pi\mathbf{k}\cdot\mathbf{r}) \tag{3.51}$$

$$\mathbf{k}(t) = \gamma \int_0^t \mathbf{G}(t')\mathrm{d}t' \tag{3.52}$$

We emphasize again that the model presented here has neglected many important effects, relaxation of the signal being chief among them. While the spatial distribution of the spin density is one image of interest, spatial maps of the relaxation times (T_1, T_2) are also of interest. Furthermore, we have not mentioned many practical concerns, such as how the k space is sampled, the limits on G that can be practically realized, etc. Indeed, the development of MR

sequences (particular arrangements of magnetic field gradients and RF pulses used), which is out of the scope of this thesis, is concerned with how to accomplish imaging taking these effects and practical concerns into account. One tailors the kind of image (i.e. the contrast) one wants by manipulating the field gradients/RF pulses such that the resulting signal expresses the contrast in which one is interested. However, at this point, we can assume that for the basic MR image model (applicable, with modifications, to Chapter ???), we can use the finite dimensional model

$$\mathbf{y} = F\mathbf{x} + \mathbf{n}, \tag{3.53}$$

where $\mathbf{y}$ is the measured signal, F is the Discrete Fourier Transform, $\mathbf{x}$ is the image of interest, i.e. the effective spin density weighted by different factors (e.g. by T_1, T_2), and $\mathbf{n}$ is Gaussian noise. Note that both the signal and the reconstructed image are complex-valued in general. The signal is complex-valued by construction. However, while the spin density is real-valued in principle, the data acquisition process, signal processing, etc usually add phases such that the effective spin density is complex-valued. The images shown for typical MR images are produced from taking the magnitude of the reconstructed image. For an example of acquired k-space and reconstructed magnitude image, see Figure 3.6

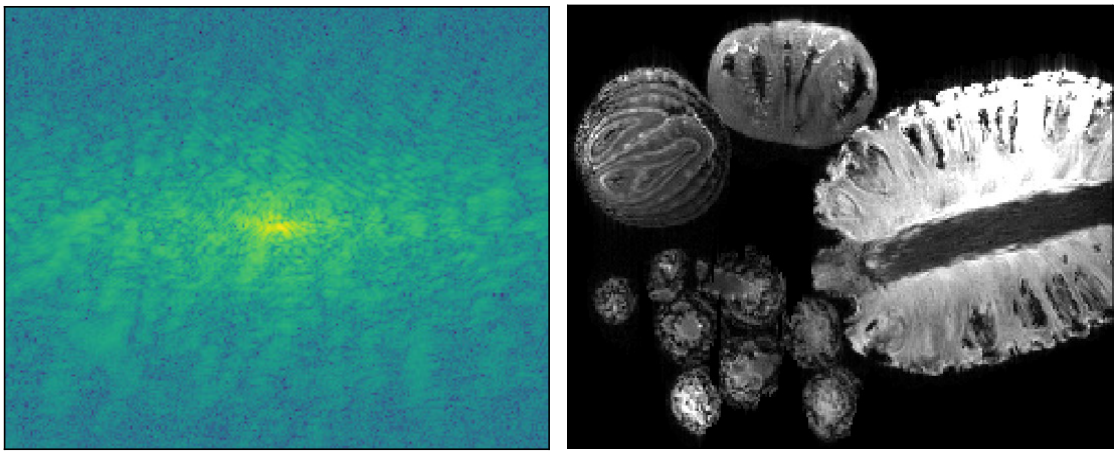


Figure 3.6: To the right, we show a magnitude MR image obtained from scanning several fruits/vegetables. To the left, we show the logarithm of the magnitude of the k-space measurements from which the image to the right was reconstructed.

3.5 Image Contrast

In the previous section, we derived an inverse problem which allows for the reconstruction of a spatial image of the spin density from MR measurements. However, we emphasize that in fact, what we recover is an effective spin density, since it is really the spin density multiplied or weighted by a series of complex factors, such as relaxation factors, peculiarities due to the specific MR machine used, additional factors due to the sequence used; detailed examination

of these factors are out of the scope of this thesis. Therefore, the main utility of the image is to see the contrast/differences in image magnitude between different parts of the scanned sample. The actual image magnitudes are not quantitatively meaningful in themselves because of the complex multiplicative factors. Such images are then called **contrast weighted** images. For example, the diagnostic value of scaling a contrast image globally by an arbitrary constant is unchanged since there is no difference in contrast.

However, through different sequences (different arrangements of RF pulses and magnetic field gradients), one can ensure that the resulting image is the spin density **multiplied predominately by a factor of interest**, such as the relaxation times T_1 and T_2, or the diffusion coefficient. In this way, the images produced are said to be T_1 or T_2 or diffusion weighted, meaning that the image intensity in a voxel depends largely on the value of the T_1 or T_2 or diffusion of the spins in that voxel. In other words, one can design sequences to choose what kind of contrast an MR image will have.

For example, if we express the Bloch equation for the transverse magnetization in the rotating frame, we find that

$$\frac{\mathrm{d}\mathbf{M}_\perp}{\mathrm{d}t} = -\frac{\mathbf{M}_\perp}{T_2}, \tag{3.54}$$

which has the well known exponential solution

$$\mathbf{M}_\perp(t) = \mathbf{M}_\perp(0)\exp(\frac{t}{T_2}). \tag{3.55}$$

Therefore, roughly, measuring the signal from the transverse magnetization at time T after tilting the magnetic moment to the transverse plane with a $\frac{\pi}{2}$ RF pulse will add an exponential weighting to the effective spin density of the form $\exp(\frac{T}{T_2})$. This implies, for example, that the actual spin density being equal, regions of the sample with lower T_2 relative to others will also have lower image magnitudes, allowing us to see contrast in the image due to T_2 differences.

Similarly, contrast images can also be produced for other parameters of interest, such as the T_1, the proton density, and diffusion coefficients; for examples, see Figure 3.7

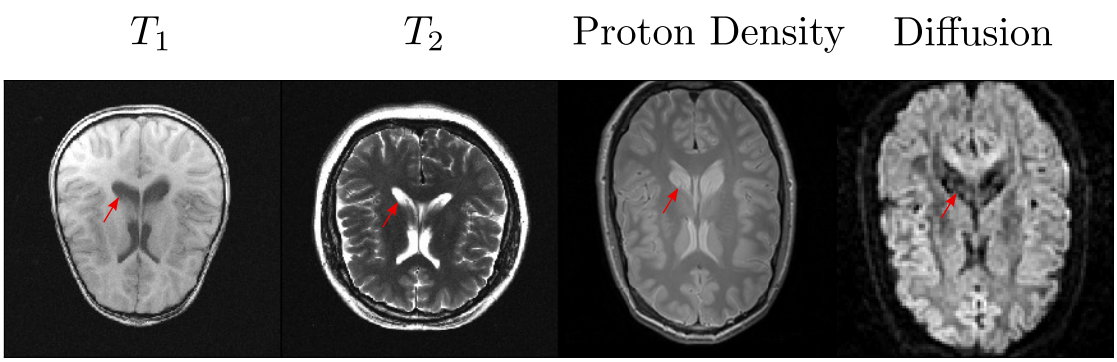


Figure 3.7: Here we show axial MR images of the brain with different contrasts from the fastMRI [60] and IXI [61] datasets. We can see that the relative contrast of the image intensities for the same parts of the brain (ventricles, white matter, gray matter) are different for each image.

4 Validation of Self-Supervised, Undersampled MRI Reconstruction

The content of the following chapter is based on the preprint version of the article: "Validation and Generalizability of Self-Supervised Image Reconstruction Methods for Undersampled MRI" submitted to the Journal of Machine Learning for Biomedical Imaging [62].

4.1 Introduction

For the first non-background chapter of our thesis, we begin, in some sense, by focusing on the end of the inverse problem pipeline: validation of methods. As mentioned in closing in Chapter 2, validation of inverse problems is difficult to discuss generically as acceptable validation depends entirely on the use of the solutions; for medical imaging, where solutions of inverse problems can be used by radiologists to diagnose pathologies, the validation in a realistic setting is of utmost importance in order to reflect the expected performance in the clinic. However, validation becomes complicated in situations where large, realistic datasets of ground truth data are not available, as is the case for the inverse problems in this thesis. In this chapter, we engage in a rigorous validation of self-supervised methods for reconstructing MR images from undersampled MR data; in contrast to the previous literature, we try to go beyond the typical validation pipeline by using/evaluating reconstructions of prospectively accelerated data, i.e. data as it would arise in the clinic, as much as possible. We begin with an explanation of parallel and undersampled MRI.

4.1.1 Parallel, Undersampled MRI Reconstruction

In Chapter 3, we showed that the measurements in MRI are connected to the underlying density of spins through the Fourier transform (Equation 3.53). However, as we mentioned, this model neglects many factors, some of which we will address here. First, our construction assumed the use of only one coil/sensor for taking measurements. However, one can use

multiple coils/sensors and reconstruct an image jointly from the data from each coil; this is called parallel MRI.

Furthermore, we did not discuss the sampling of the k-space in the previous section. Neglecting the physics of how the k-space is sampled in practice through MR sequences, we note that the sampling must be discretized and finite for practical reasons, even though in theory both the image and the k-space are continuous. Then the resulting image will also be discretized and finite. Consider a 2 dimensional image for simplicity, and suppose that the sampling is symmetric in both dimensions/positive and negative axes. Suppose we want an image which encompasses a field of view (FOV) or spatial extent of LxL millimeters (mm), with a spatial resolution of δx mm; i.e. the discretization of the image is δx mm. We can consider the corresponding continous k-space measurements as a band-limited function since its Fourier transform, which is the image, is limited to the FOV of LxL millimeters, i.e. the spatial frequency f satisfies $\|f\| \leq \frac{L}{2}$. Then the Nyquist-Shannon theorem dictates that a discretized sampling of the k-space can recover the continuous k-space without aliasing as long as the sampling interval or discretization δk satisfies

$$\delta k \leq \frac{1}{L} \tag{4.1}$$

Furthermore, in order to achieve an image resolution of δx, this implies a constraint on the number of samples taken in k-space. Roughly speaking, if the sampling of each k-space axis takes place in the interval $[-\frac{n}{2}\delta k, \frac{n-1}{2}\delta k]$, then

$$\delta x = \frac{1}{\delta k \times n} \tag{4.2}$$

Therefore in order reconstruct an image from sampling the k-space discretely and finitely, the sampling interval and number of samples must be chosen carefully in order to faithfully represent the Fourier transform of the underlying image. In the above example, this implies discretely sampling the 2D box $[-\frac{n}{2}\delta k, \frac{n-1}{2}\delta k] \times [-\frac{n}{2}\delta k, \frac{n-1}{2}\delta k]$. However, depending on the FOV, the image resolution desired, and the dimensionality (e.g. going to 3D), the time to sample the k-space according to these constraints can be prohibitive.

Therefore, it would be desirable to reconstruct MR images from **undersampled** measurements, where only a fraction of the samples theoretically necessary are acquired, as this will accelerate the acquisition. For example, a 5x acceleration corresponds to undersampling by a factor of 5/taking only 20 percent of the theoretically required measurements. However, undersampled measurements require special reconstruction techniques to compensate for having less information.

In this chapter, we consider the inverse problem of parallel, undersampled MRI reconstruction, where we want to reconstruct an image from the joint measurements from multiple sensors, with the measurement being undersampled in the aforementioned sense.

4.1.2 Motivation

Since the introduction of MRI, methods for image reconstruction have evolved with acquisition acceleration and have seen great advances with parallel imaging techniques such as sensitivity encoding (SENSE) [63] and generalized auto-calibrating partially parallel acquisition (GRAPPA) [64]. While parallel imaging reliably accelerates clinical contrasts by factors of two to three, more recent methods such as compressed sensing (CS) have achieved even higher acceleration factors [65]. Now, supervised deep learning methods reign as the state of the art in the reconstruction of accelerated acquisitions [48], [66], [67]. However, these supervised methods require a non-trivial amount of fully sampled data to use as ground truth/target, which can be difficult or infeasible to obtain depending on the type of acquisition. Consequently, there has been interest in unsupervised or self-supervised, deep learning approaches which train solely on accelerated acquisitions, with no need for ground truth, fully sampled data [68]–[71]. However, the validation of these methods is generally done by quantitative evaluation through pixel-wise metrics on retrospectively undersampled acquisitions (i.e., artificial undersampling of a fully sampled dataset), accompanied by qualitative evaluation on datasets where no ground truth is available. This limitation may stem from commonly used datasets [60], [72] being fully sampled, as well as difficulties in acquiring datasets which contain both fully sampled and prospectively accelerated scans without motion corruption. However, this neglects quantitative evaluation of reconstructions from prospectively undersampled data, the clinically relevant scenario, as well as potential differences between prospective and retrospective reconstructions; furthermore, the pixel-wise metrics generally used may not correlate well with the perceptual quality of the images. This point is crucial for clinical deployment as even if different methods can be robustly ranked using retrospective data, the image quality from prospective data from the different methods may be unsuitable for clinical use. Furthermore, if these techniques will be used in future clinical routines, they likely will be subject to variations of data quality and content. For example, different surface coils, parameter differences between centers or even the use of the same sequence on different organs. Therefore, the generalizability, i.e., inference data different from the training/tuning data (e.g. in terms of field strength, sequence parameters, motion, anatomy, etc.), using prospective data should be explored.

4.1.3 Contributions

In this work, we conducted an extensive, realistic validation of state of the art self-supervised reconstruction methods through two overarching experiments. First, by using data with both full and prospective sampling, we quantitatively and qualitatively evaluate both prospective and retrospective reconstructions using both pixel-wise and perceptual metrics for fidelity to ground truth, allowing us to study them individually as well as to see any relevant differences. Second, using an extensive, prospectively accelerated dataset with changes in anatomy, contrast, hardware, field strength, etc., we study the generalizability of the methods quantitatively, using no-reference image quality metrics and qualitatively, using rating by MR scientists and a

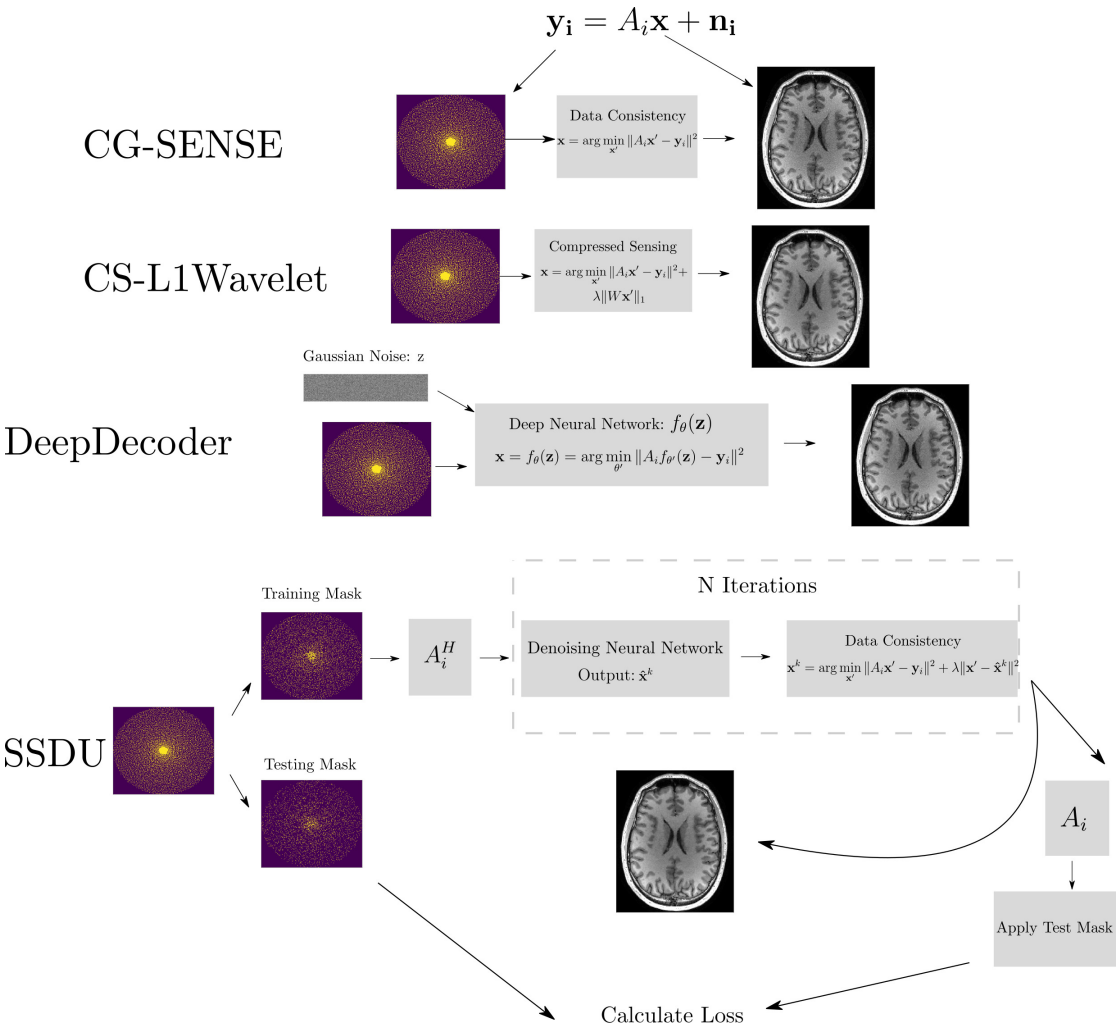


Figure 4.1: An overview of the basic formulation of the MR reconstruction inverse problem, as well as how each method in the paper solves the inverse problem.

radiologist.

4.2 Theory

The self-supervised, machine-learning based methods we examine in this paper rely on two powerful ideas drawn from machine learning: self-supervised denoising and restriction to the range of convolutional neural networks (CNN) as an effective prior for image reconstruction. These concepts have been shown to be both empirically effective and theoretically well founded, making them attractive for clinical use. In Figure 4.1, we show an overview of the different methods used in this paper. We begin with the basic inverse problem formulation of parallel MR image reconstruction. This formulation differs from the formulation in Chapter 3 in that one needs to take into account the undersampling (through a mask) and the spatial

sensivity of each coil (through a spatial sensitivity map). Let $\mathbf{y}_i, \mathbf{n}_i$ denote the undersampled MR measurements and Gaussian noise respectively, from the ith coil element and $\mathbf{x}$ denote the underlying image. These quantities are related by:

$$\mathbf{y}_i = A_i\mathbf{x} + \mathbf{n}_i, \tag{4.3}$$

$$A_i = M \circ F \circ S_i \tag{4.4}$$

where M is the element-wise multiplication by a mask (corresponding to the location of the undersampled measurements), F denotes the Fourier transform, and S_i denotes element-wise multiplication by the ith sensitivity map (note that this sensitivity map encodes the spatial sensitivity of the coil). The classical regularized reconstruction of $\mathbf{x}$ is the solution of an optimization problem

$$\mathbf{x} = \arg\min_{\mathbf{x}'} D(\mathbf{x}', \mathbf{y}) + \lambda R(\mathbf{x}'), \tag{4.5}$$

where $D(\mathbf{x}, \mathbf{y})$ measures the consistency of the solution to the data (e.g. $\| A_i\mathbf{x} - \mathbf{y}_i \|^2$), $R(\mathbf{x})$ is a regularization function, which, for example, prevents overfitting to the noise, and λ is the regularization parameter. In combination with incoherently undersampled measurements, compressed sensing reconstructions have been shown to effectively reconstruct the underlying images by setting $R(\mathbf{x})$ to encourage sparsity of $\mathbf{x}$ in a set domain [65]. Many state of the art deep learning methods, both supervised and unsupervised, implicitly or explicitly parametrize $R(\mathbf{x})$ with a neural network.

4.2.1 DeepDecoder

The first self-supervised method we examine is called DeepDecoder. DeepDecoder is based on a seminal work in the machine learning literature called Deep Image Prior (DIP) [73] which showed that untrained CNNs could be used to effectively solve inverse problems without ground truth. Concretely, let f_θ denote a randomly initialized CNN with parameters θ. Let $\mathbf{z}$ be a sample of a random, Gaussian vector. Then DIP solves Equation 4.5 by

$$\mathbf{x} = f_\theta(\mathbf{z}) = \arg\min_{\theta'} \| A_i f_{\theta'}(\mathbf{z}) - \mathbf{y}_i \|^2 \tag{4.6}$$

This formulation is equivalent to setting $R(\mathbf{x})$ to the indicator function with support over the range of the neural network; this assumes that the convolutional network f_θ itself provides a strong prior on the space of image solutions, such that only the data consistency term needs to be minimized. However, since only the noisy signal $\mathbf{y}$ is used during training, minimization can overfit the noise in the signal, depending on the inverse problem being solved (e.g. denoising, super-resolution), thus requiring early stopping [73]. DeepDecoder [70] is a CNN with a simplified architecture (only upsampling units, pixel-wise linear combination of channels, ReLU activation, and channel-wise normalization) which is amenable to theoretical analysis and was shown to be competitive with other architectures for solving inverse problems in a

DIP framework.

In [74], the authors theoretically showed that for the case of image recovery from compressed sensing measurements, CNNs (in particular, CNNs with the structure of DeepDecoder) are self-regularizing with respect to noise and can simply be trained to convergence with gradient descent without early stopping or additional regularization, provided that the true, underlying image has sufficient smoothness/structure. In a knee MR example, they showed that early stopping would have only provided a marginally better solution than running to convergence. Hence, from a theoretical and practical standpoint, DeepDecoder is attractive for self-supervised reconstruction from undersampled measurements. We emphasize that DeepDecoder entails training a separate network for each separate acquisition/slice, rather than training a single network over a dataset of undersampled acquisitions.

4.2.2 Self-supervised learning via data under-sampling

The second self-supervised method we examine is called Self-supervised learning via data under-sampling (SSDU). SSDU uses an unrolled, iterative architecture, with alternating neural network and data consistency modules, to reconstruct MR images using only undersampled measurements, with the adjoint image corresponding to the input k-space measurements as an initial guess. It solves Eqn 4.5 using an iterative, variable splitting approach where the kth iteration consists of

$$\hat{\mathbf{x}}^k = \text{CNN}(\mathbf{x}^{k-1}) \tag{4.7}$$

$$\mathbf{x}^k = \arg\min_{\mathbf{x}'} \| A_i \mathbf{x}' - \mathbf{y}_i \|^2 + \lambda \| \mathbf{x}' - \hat{\mathbf{x}}^k \|^2. \tag{4.8}$$

where the superscript denotes the iteration, CNN denotes a generic CNN, and $\hat{\mathbf{x}}^k$ denotes an auxiliary variable. The regularization parameter λ is learned during training. Let f_{SSDU} denote the function defined by the unrolled network. In each training step of SSDU, the k-space of the data is split into two disjoint sets, denoted by $\mathbf{y}_\Theta$ and $\mathbf{y}_\Lambda$. $\mathbf{y}_\Theta$ is passed to the unrolled network as input. The loss function for SSDU compares the simulated k-space measurements of the corresponding image output $f_{SSDU}(\mathbf{y}_\Theta)$ to $\mathbf{y}_\Lambda$:

$$L(\mathbf{y}_\Lambda, A_\Lambda f_{SSDU}(\mathbf{y}_\Theta)) \tag{4.9}$$

where A_Λ is the measurement operator corresponding to sampling the locations of Λ, and L is an equally weighted combination of the L_1 and L_2 loss. Hence, during each training step, f_{SSDU} only sees information from $\mathbf{y}_\Theta$, and the loss is only computed over a disjoint set $\mathbf{y}_\Lambda$. We note that at inference time, the entire, acquired k-space measurements are given as input. While the authors of SSDU give an intuitive explanation of this approach as similar to cross validation in order to prevent overfitting to noise or learning the identity, results from the machine learning literature on blind, signal denoising can help give a theoretical explanation. In the Noise2Self framework [75], the authors prove that a neural network can be

trained to denoise a noisy signal, using solely the noisy signal for training. In the following, we describe a special case of the general theory proven in [75]: the signal, $\mathbf{y}$ is partitioned into disjoint sets, $\mathbf{y}_\Theta$ and $\mathbf{y}_\Lambda$. The neural network takes as input $\mathbf{y}_\Theta$ and predicts a denoised signal; the loss function used for training is the mean squared error between the denoised signal restricted to Λ and $\mathbf{y}_\Lambda$. The authors showed that this loss function/strategy approximates **the mean squared error between the signal predicted by the network and the ground-truth signal without noise, plus a constant independent of the network. Hence the Noise2Self strategy allows to minimize the error between the predicted signal and the ground truth signal with only access to the noisy signal.** We can see that the training of SSDU conforms to the Noise2Self framework with the k-space measurements acting as the noisy signal, albeit with SSDU using an L_1 loss in addition to the L_2 loss. Thus, SSDU takes as input the noisy, acquired k-space measurements, and is optimized to output an image whose simulated k-space measurements are the acquired k-space measurements **without noise**. In this way, SSDU avoids overfitting to noise. This, combined with the powerful image prior from using a CNN as the neural network as well as the interleaving of the data consistency term, explains SSDU's demonstrated ability to provide denoised images which retain image sharpness, as compared to traditional methods. We can interpret SSDU as an iterative method which interleaves the application of a denoising network and a data consistency step. We note in contrast to DeepDecoder, that we can train different networks for separate acquisitions or train a single, reusable network on a dataset of undersampled acquisitions. In this paper, we do the latter.

In conclusion, both unsupervised approaches accomplish noise robust MR reconstruction using only noisy, undersampled MR measurements.

4.3 Methods

In the following experiments, we compare four image reconstruction methods:

1. **CG-SENSE**, which solves Equation 4.5 with no regularization using the conjugate gradient algorithm; this is a least squares fit to the acquired data similar to the description in [76].
2. **CS-L1Wavelet**, where we solve Equation 4.5 with a compressed sensing reconstruction, with $R(\mathbf{x}) = \|W\mathbf{x}\|_1$, where W is a wavelet transform operator.
3. **DeepDecoder** with a depth/width of 300/10 and Gaussian input of size (10,10).
4. **SSDU**, where we use a U-Net [77] with 12 channels and 4 downsampling/upsampling layers.

We used Sigpy[78] for the computation of CS-L1Wavelet and ESPiRiT[79] sensitivity maps. We implemented CG-SENSE and SSDU in Pytorch [80], and we used Github implementations of

DeepDecoder [I] and U-Net [II]. We used Adam [81] to optimize both SSDU and DeepDecoder. SSDU was trained until convergence (10 epochs) with a learning rate of 0.5e-4. For each subject, DeepDecoder was optimized using the acceleration strategy in [82]; a single slice for each subject is optimized to convergence (over 10,000 iterations) from a random initialization. All other slices are optimized for 1,000 iterations, initialized with the network model from this single slice. All training and inference was done on a NVIDIA Quadro RTX 8000 with 45GB of RAM.

4.3.1 Training Data and Hyperparameter Tuning

To mimic a realistic scenario with a sequence for which fully sampled, ground truth data is difficult/infeasible to acquire, and where the training dataset is limited in size and variability, we acquired for ten healthy subjects a 5x accelerated 3D MPRAGE prototype sequence [83] of the brain at 3T (MAGNETOM Prisma^{Fit}, Siemens Healthcare, Erlangen, Germany) using a 64ch Rx Head/Neck coil. These incoherently undersampled data were used for training/tuning the hyperparameters of all reconstruction methods. In what follows, all training/inference is done on 2D slices of both phase-encoding directions formed from performing the inverse Fourier transform along the readout direction. In the absence of prior knowledge/heuristics, the hyperparameters of the methods should also be tuned in a self-supervised way, as the traditional method for hyperparameter tuning, using a hold-out set of data for which the ground truth is known, is not available in the realistic scenario. We use the Noise2Self framework, which also underlies SSDU, for selecting hyperparameters, as it optimizes for preventing overfitting to the noise in the measurements. For example, to set the regularization parameter of CS-L1Wavelet, we treat it as a function with a single parameter (λ). We can then optimize this parameter using the Noise2Self training framework to estimate the λ which minimizes the noise-free error between simulated measurements and the acquired measurements. Concretely, we fix 20 logarithmically spaced values from 0.00001 to 0.1. We set each value as λ and run 50 image reconstructions corresponding to different, random masks and average the corresponding errors with respect to the complementary mask in order to approximate the true measurement error associated with using each value. We then select the value with the lowest measurement error as the optimal regularization parameter. This is done for each slice in each subject; the final regularization value which is used throughout this paper is the average over all subjects. The hyperparameters of DeepDecoder and SSDU are set similarly with a grid search over the network hyperparameters, albeit over a much smaller set of data due to the high computational demand.

[I] https://github.com/MLI-lab/cs_deep_decoder
[II] https://github.com/facebookresearch/fastMRI

4.3.2 Validation using Prospectively Accelerated and Fully Sampled Data

In our first experiment, using the aforementioned 3D MPRAGE prototype sequence used for acquiring the training/tuning data, we acquired both fully sampled and 5x prospectively accelerated scans of the following:

1. Siemens multi-purpose phantom E-38-19-195-K2130 filled with $MnCl_2 \cdot 4H_2O$ doped water
2. Assortment of fruits/vegetables (Pineapple, tomatoes, onions, brussel sprouts)

This allowed us to reconstruct prospective, retrospective (applying the same mask as in prospective sampling on the fully sampled data), and fully-sampled images.

No in-vivo data was used in this experiment since subject motion could bias the results. Furthermore, we used fruits/vegetables as a second phantom since they have more complex structures than a water filled container.

Quantitative Assessment

First, we qualitatively compared the results through visual inspection. Second, we quantitatively compare reconstructions to the ground truth using Peak Signal to Noise Ratio (PSNR) [84], the Structural Similarity Index Measure (SSIM) [85], and a metric we will call the Perceptual Distance (PercDis) score. While the first two are commonly used metrics in MR image reconstruction/image reconstruction in general, the PercDis score comes from computer vision (super-resolution, style transfer, etc), where it is called the perceptual loss [86]; the distance between two images is defined as the L_1 distance between the respective induced features from an intermediate layer of a pretrained network. The scores of center cropped slices, along the read-out direction, are averaged for the final score.

4.3.3 Generalizability of Self-Supervised Reconstruction Methods

In our second experiment, we examined the generalizability of the reconstruction methods. To that end, we scanned three, healthy subjects with the following prospectively accelerated sequences(anatomy):

1. 1.5T MPRAGE (Brain)
2. 3T MPRAGE (Brain)
3. 7T MPRAGE (Brain)
4. 3T MPRAGE with 1Tx/20Rx Coil (Brain)

5. 3T MPRAGE with Subject Motion (Brain)
6. 3T MPRAGE with Different Parameters (Brain)
7. 3T, T_1 SPACE (Brain)
8. 3T, T_2 FLAIR SPACE (Brain)
9. 3T, PD SPACE (Knee)
10. 3T, T_2 SPACE (Knee)

The brain scans at 1.5T, 3T and 7T (MAGNETOM Sola, Vida, and Terra, Siemens Healthcare, Erlangen, Germany) were done using a 1Tx/20Rx, 1Tx/64Rx (unless otherwise stated), and 8pTx/32Rx (Nova Medical, Wilmington, MA, USA) head coil, respectively. The knee scans at 3T were done with a 1Tx/18Rx coil. All detailed sequence parameters can be found in Table 4.2.

As ground truth data is not available since motion would render quantitative comparison difficult due to blurring from image co-registration, we evaluated the reconstructions from the above data quantitatively through no-reference image quality metrics and qualitatively through rating by four MR scientists and a radiologist.

No-Reference Image Metrics

No-reference image quality metrics quantify the quality of a given image (i.e. blurriness, noise) using only its statistical features in a way that correlates with the perceptual quality of a human observer. They have been shown to potentially be useful for MR/medical image evaluation without ground truth [87], [88]; we use the following three metrics: a metric used originally for assessing the quality of JPEG-compressed images which we call NRJPEG [89], the Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [90], and Perception based Image Quality Evaluator (PIQE) [91]. BRISQUE and PIQE have also been used in other image reconstruction challenges where the ground truth is not available, such as super-resolution [92]. The metrics were calculated for the central 100 slices (along the read-out direction) of each reconstruction.

Human Quality Rating

The human quality rating was done according to [47] by four experienced MR scientists and a radiologist. Using a 4-point ordinal scale, reconstructed images were evaluated for sharpness (1: no blurring, 2: mild blurring, 3: moderate blurring, 4: severe blurring), SNR (1: excellent, 2: good, 3: fair, 4: poor), presence of aliasing artifacts (1: none, 2: mild, 3: moderate, 4: severe) and overall image quality (1: excellent, 2: good, 3: fair, 4: poor). Raters were blinded to the reconstruction method.

4.3.4 Statistical Significance

For all quantitative metrics/ratings, we use the Wilcoxon rank sum test with significance level $\frac{0.05}{6}$ (Bonferroni correction with 6 pair-wise comparisons among the 4 methods) to determine statistical significance.

4.4 Results

In general, perceptually, CG-SENSE produces noisy but sharp images since it is not regularized. DeepDecoder produces smoother reconstructions with spatially varying noise behavior and sharpness, e.g Figure 4.2 (yellow arrows). CS-L1Wavelet and SSDU produce similar images, smoother than those of CG-SENSE with comparable sharpness; however, CS-L1Wavelet exhibits more artifacts, e.g Figure 4.2 (red arrows).

4.4.1 Validation Using Prospectively Accelerated and Fully Sampled Data

In Fig. 4.2 and Fig. 4.3, we can see spatial distortions of hyper/hypo-intense features in the prospective reconstructions and changes in contrast in comparison to the ground truth reconstruction; this distortion is not present in the retrospective reconstructions; however, they are similar across all reconstruction methods.

Retrospective reconstructions have significantly higher mean scores for all metrics in comparison to the prospective reconstructions in both acquisitions (see Table 4.1).

Comparing the methods, in the phantom, the prospective/retrospective reconstructions of DeepDecoder have the highest pixel-wise fidelity to the ground truth with a mean PSNR of (18.67/23.44) and SSIM of (0.49/0.52); however, qualitatively, it has more spatially varying oversmoothing than those of CS-L1Wavelet and SSDU. SSDU and CS-L1Wavelet perform similarly, with the highest qualitative similarity to the ground truth, with SSDU having a higher mean PSNR overall (17.79/21.95). In contrast to the PSNR/SSIM results, with the PercDis score, SSDU has the highest fidelity to the ground truth (0.63/0.61).

Qualitatively and quantitatively (with PSNR and SSIM), the differences between the methods are much less in the fruits/vegetables. The main qualitative difference is the greater denoising capabilities of SSDU and CS-L1Wavelet in comparison to CG-SENSE and DeepDecoder. Quantitatively, there are only minor differences between the methods with respect to PSNR and SSIM. In contrast, the PercDis scores clearly indicate that CS-L1Wavelet and SSDU (with similar scores) are perceptually more similar to the ground truth than CG-SENSE and DeepDecoder (with similar scores).

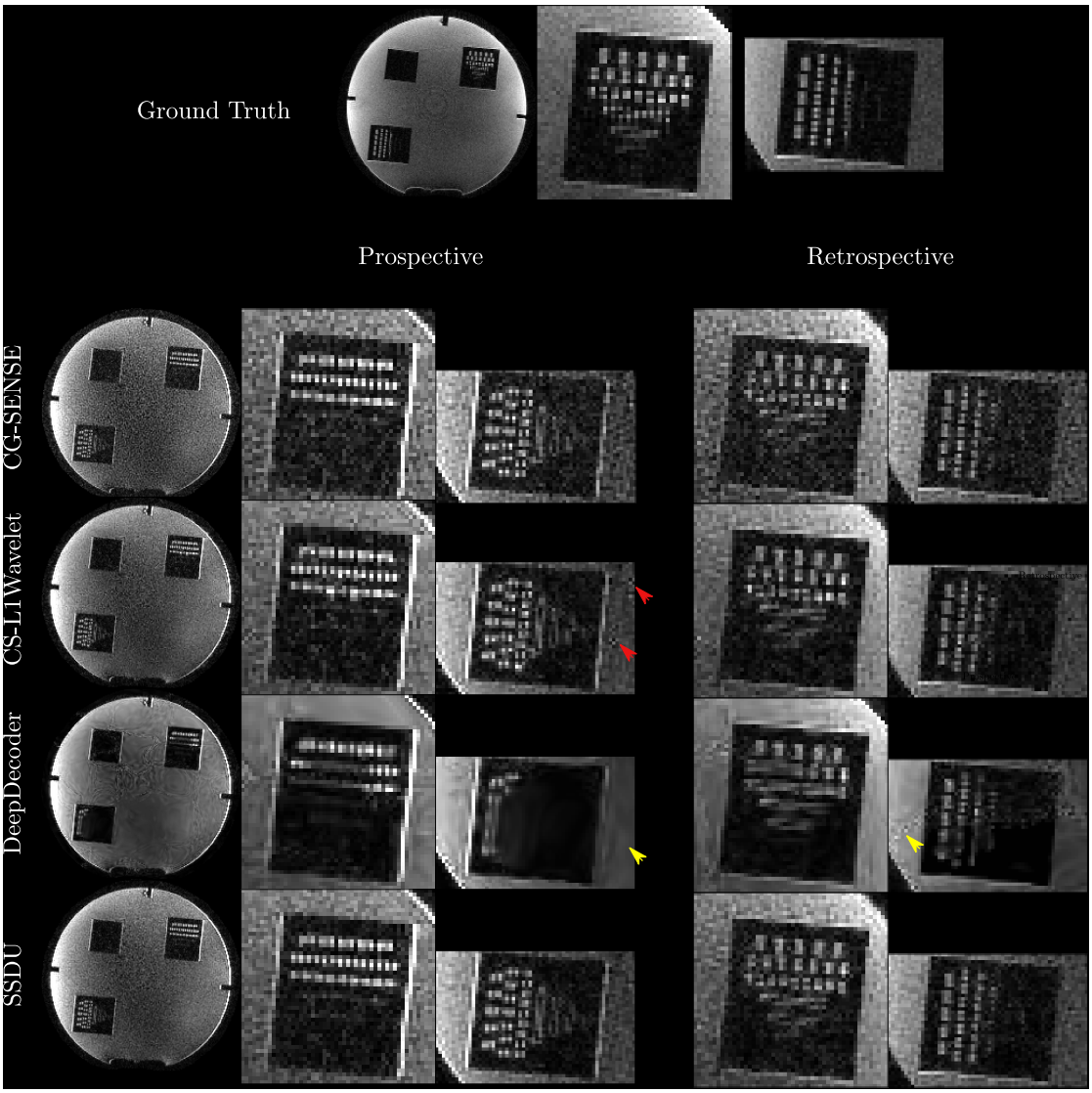


Figure 4.2: Ground truth images and reconstructed images using prospectively and retrospectively accelerated data from the multi-purpose phantom, scanned with a MPRAGE sequence at 3T. Reconstructions from prospectively accelerated data are distorted (see closeups) relative to the ground truth/retrospective reconstructions. DeepDecoder exhibits spatially varying smoothness/distortion (see yellow arrows) relative to CS-L1Wavelet and SSDU which have similar scores/appearance, although CS-L1Wavelet has more artifacts (see red arrows). CG-SENSE produces noisy but sharp reconstructions, while CS-L1Wavelet and SSDU reduce noise but preserve sharpness relative to CG-SENSE.

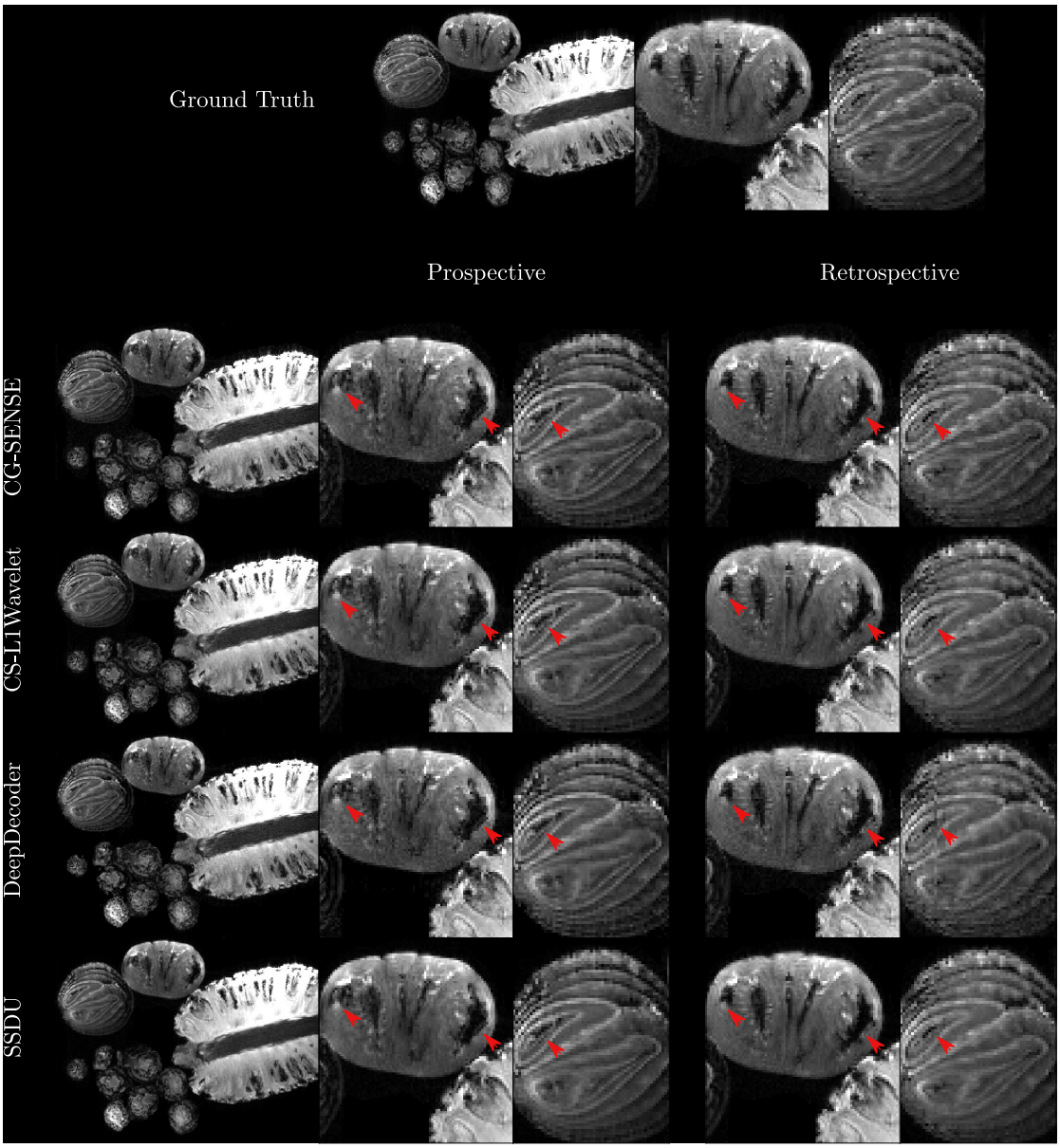


Figure 4.3: Ground truth images as well as reconstructed images using prospectively and retrospectively accelerated data from the fruits/vegetables, scanned with a MPRAGE sequence at 3T. Reconstructions from prospectively accelerated data are distorted in hypointense regions (see closeup/red arrows) relative to the ground truth/retrospective reconstructions. Qualitatively, the main difference between the methods are between CS-L1Wavelet/SSDU and CG-SENSE/DeepDecoder, where the former group is smoother than the latter.

PSNR ↑	**Phantom**		**Fruits/Vegetables**	
(μ,σ)	Prospective	Retrospective	Prospective	Retrospective
CG-SENSE	(13.54,9.69)	(16.82,12.17)	(33.4,4.86)	(39.3,5.73)
CS-L1Wavelet	(16.0,8.36)	(20.62,11.65)	(33.59,3.83)	(38.88,4.44)
DeepDecoder	(18.67,6.65)	(23.44,10.66)	(33.65,2.86)	(38.25,4.42)
SSDU	(17.79,6.9)	(21.95,10.09)	(33.88,3.75)	(39.49,4.27)

SSIM ↑				
(μ,σ)	Prospective	Retrospective	Prospective	Retrospective
CG-SENSE	(0.35,0.18)	(0.42,0.23)	(0.92,0.09)	(0.95,0.09)
CS-L1Wavelet	(0.4,0.21)	(0.47,0.27)	(0.93,0.08)	(0.96,0.08)
DeepDecoder	(0.49,0.22)	(0.52,0.28)	(0.93,0.04)	(0.95,0.08)
SSDU	(0.41,0.22)	(0.47,0.27)	(0.93,0.08)	(0.96,0.08)

PD ↓				
(μ,σ)	Prospective	Retrospective	Prospective	Retrospective
CG-SENSE	(1.05,0.08)	(1.02,0.06)	(0.45,0.16)	(0.29,0.09)
CS-L1Wavelet	(0.84,0.08)	(0.79,0.05)	(0.41,0.17)	(0.25,0.09)
DeepDecoder	(0.68,0.15)	(0.64,0.09)	(0.44,0.19)	(0.3,0.11)
SSDU	(0.63,0.13)	(0.61,0.1)	(0.42,0.16)	(0.26,0.09)

Table 4.1: Mean and standard deviation of PSNR/SSIM/PD scores of the reconstructions with respect to the ground truth for the phantom and the fruit/vegetables; arrows beside each metric denote whether higher or lower values are better. PSNR/SSIM/PD were calculated over all the slices in the read-out direction with center cropping. Using the Wilcoxon rank sum test with significance level $\frac{0.05}{6}$ (Bonferroni correction), we found statistically significant differences between each method for each metric **other than** (CS-L1Wavelet vs SDDU, Retrospective SSIM, Phantom) and (CG-SENSE vs SSDU, Retrospective PSNR, Fruits/Vegetables) Note that while with respect to PSNR/SSIM, DeepDecoder performs the best in the phantom, and all methods perform similarly in Fruits/Vegetables. In contrast, with respect to the PD score, SSDU performs the best in both cases by larger relative margins than with PSNR/SSIM.

4.4.2 Generalizability

Figures 4.4, 4.5 show axial MPRAGE brain slices at the different field strengths and corresponding closeups of the cerebellum and the left frontal lobe. Figure 4.6 shows a sagittal PD knee slice (3T) with closeups of articular cartilage interfaces in sagittal (femur) and axial (patella) views. These show the generalizability of the methods to different magnetic field strengths as well as changes in anatomy and contrast. Example reconstructions for the other sequences can be found in Figures 4.7, 4.8.

Perceptual Evaluation

Qualitatively, we can see from Figures 4.4, 4.5, 4.6 that all methods are able to generalize well (in the sense of approximately preserving performance/appearance on dataset used for training/tuning) to changing field strengths, anatomy, and contrast, although changing anatomy clearly worsened absolute image quality as compared to changing field strength. DeepDecoder preserves its spatially varying smoothing/artifacts, and SSDU/CS-L1Wavelet are able to produce images with less noise and comparable sharpness to CG-SENSE, although CS-L1Wavelet exhibits more artifacts. As expected, the perceptual quality of all methods increase with increasing field strength due to higher spatial resolution. Differences between the methods are less pronounced in the knee scan although overall image quality is worse.

No-reference Image Quality Metrics

In the first row of Figure 4.9, we show a bar plot of the scores for the no-reference image quality metrics averaged over all sequences and subjects. In general, CS-L1Wavelet and SSDU have the highest (by a small margin) mean NRJPEG score (10.54/10.39) and lowest, mean BRISQUE (29.35/28.06) and PIQE (25.56/22.87) scores, indicating better image quality in comparison to CG-SENSE and DeepDecoder.

Human Ratings

In the second row of Figure 4.9, we show bar plots of the scores from the MR scientists and the radiologist; we pooled the scores of the MR scientists. We see that MR scientists and the radiologist generally agree for evaluating SNR, aliasing, and overall quality, rating CS-L1Wavelet/SSDU as being better than or the same as CG-SENSE/DeepDecoder. We recall that lower ratings correspond to better quality. MR scientists rated CS-L1Wavelet/SSDU with a mean overall quality of (2.09/1.97) as compared to CG-SENSE/DeepDecoder with (2.96/3.57). The radiologist rated CS-L1Wavelet/SSDU with a mean overall quality of (2.73/2.23) as compared to CG-SENSE/DeepDecoder with (3.63/3.87). We note that for both sets of raters, the difference between CS-L1Wavelet and SSDU in overall image quality was found to not be statistically significant. Furthermore, when we restrict our analysis to the average score change between the subgroup of changes in field strength vs. the subgroup of PD Knee/T_2 Knee

scans, the overall image quality rating of CG-SENSE/CS-L1Wavelet/DeepDecoder/SSDU all worsen in the knee scans for the MR scientists, with increases of 0.26,0.40,0.11, and 0.79 respectively. In contrast, for the radiologist, this shift results in changes of -0.33,0.33,-0.16, and 0.83 respectively, indicating that only CS-L1Wavelet and SSDU worsened.

4.5 Discussion

In contrast to the previous literature, this work critically examines the validation and generalizability of self-supervised algorithms for undersampled MRI reconstruction through novel experiments with a focus on prospective reconstructions, the clinically relevant scenario. To this end, we analyze results from acquiring both fully-sampled and prospectively accelerated data on two phantoms and prospectively accelerated, in-vivo data over a wide variety of different sequences.

4.5.1 Validation using Prospectively Accelerated and Fully Sampled Data

Concerns about the differences between prospective and retrospective reconstructions were also raised in [93], in the context of end-to-end, supervised methods for parallel MR image reconstruction. In particular, they noted that retrospective undersampling neglects potential differences in signal relaxation across echo trains, and verification should be performed before clinical use. From our results using both fully sampled and prospectively accelerated data, it is clear that for the 3D MPRAGE sequence, prospective vs. retrospective reconstructions can differ meaningfully, with retrospective reconstructions having greater fidelity to the fully sampled reconstruction; prospective reconstructions exhibit spatial distortions and local changes in contrast, with respect to the ground truth. This is despite the methods being tuned/trained on prospectively accelerated data; hence, this can be attributed to the differences in the prospectively vs. retrospectively sampled k-space data, potentially due to the different gradient patterns used in the sequences. This difference is relevant both for self-supervised and supervised machine learning methods; indeed, end-to-end, supervised methods which are trained on retrospective data may yield even greater distortion than self-supervised methods when prospective data is used for inference. However, the performance ranking of the different methods was the same in both prospective and retrospective reconstructions. Therefore, retrospective image quality cannot necessarily be taken as a reliable proxy for prospective image quality; however, they can be used to show differences between methods.

The quantitative results in the phantom show how ranking by PSNR and SSIM can be misleading, as images that are perceptually/qualitatively more similar to the ground truth (SSDU,CS-L1Wavelet) can have significantly worse or almost identical mean PSNR/SSIM scores than images which are less qualitatively similar (CG-SENSE,DeepDecoder). In contrast, ranking with the PercDis score, which measures distances between the feature activations within a pretrained classification network of the images rather than the images themselves, better matches with the perceptual quality of the images, showing that SSDU or SSDU/CS-L1Wavelet

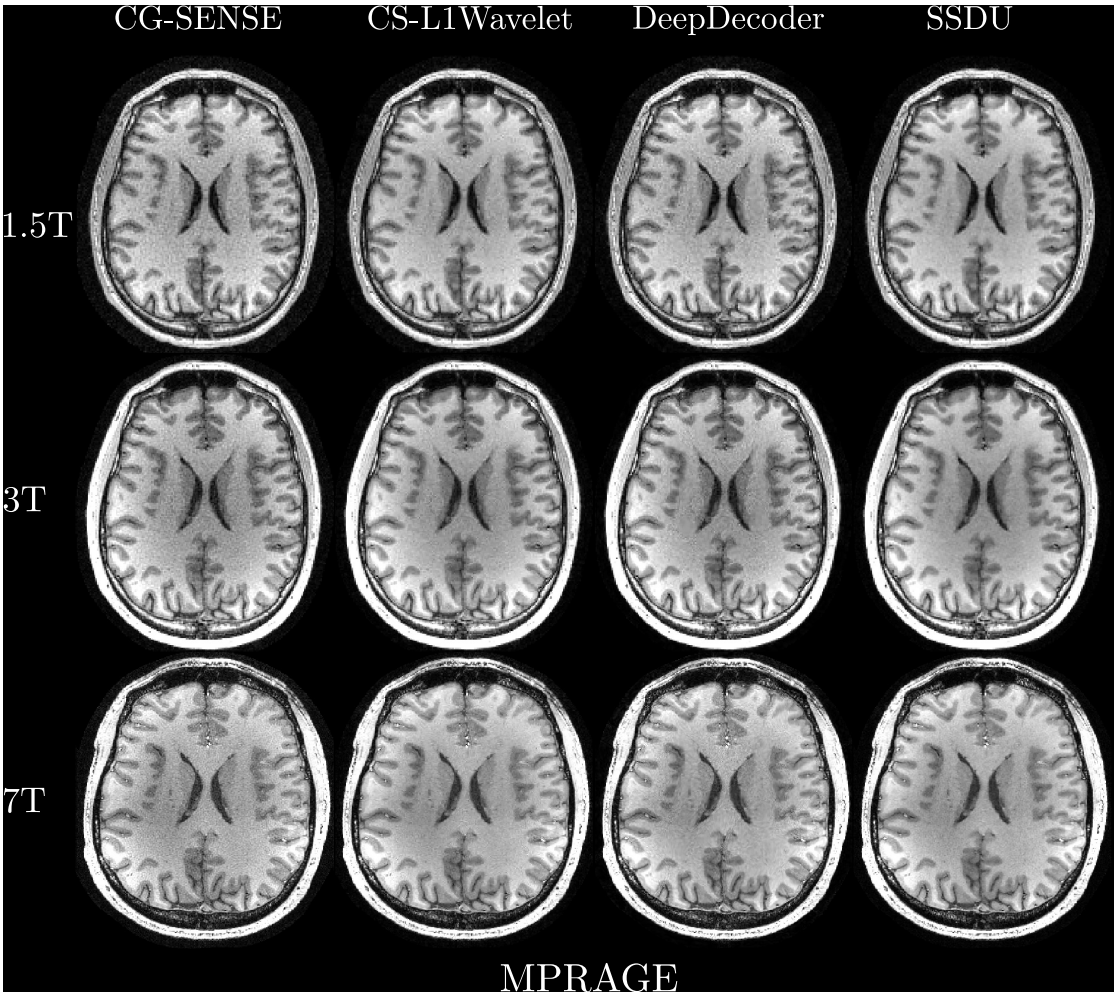


Figure 4.4: Axial slices from prospective reconstructions of MPRAGE scans of the brain at different field strengths. **Images are not co-registered**; The interpolation of image co-registration introduces blurring and thus was omitted. We chose slices at similar locations for visualization. CG-SENSE produces noisy but sharp reconstructions, and DeepDecoder produces smoother reconstructions with spatially varying noise and oversmoothing. CS-L1Wavelet and SSDU produce similarly smooth/sharp reconstructions.

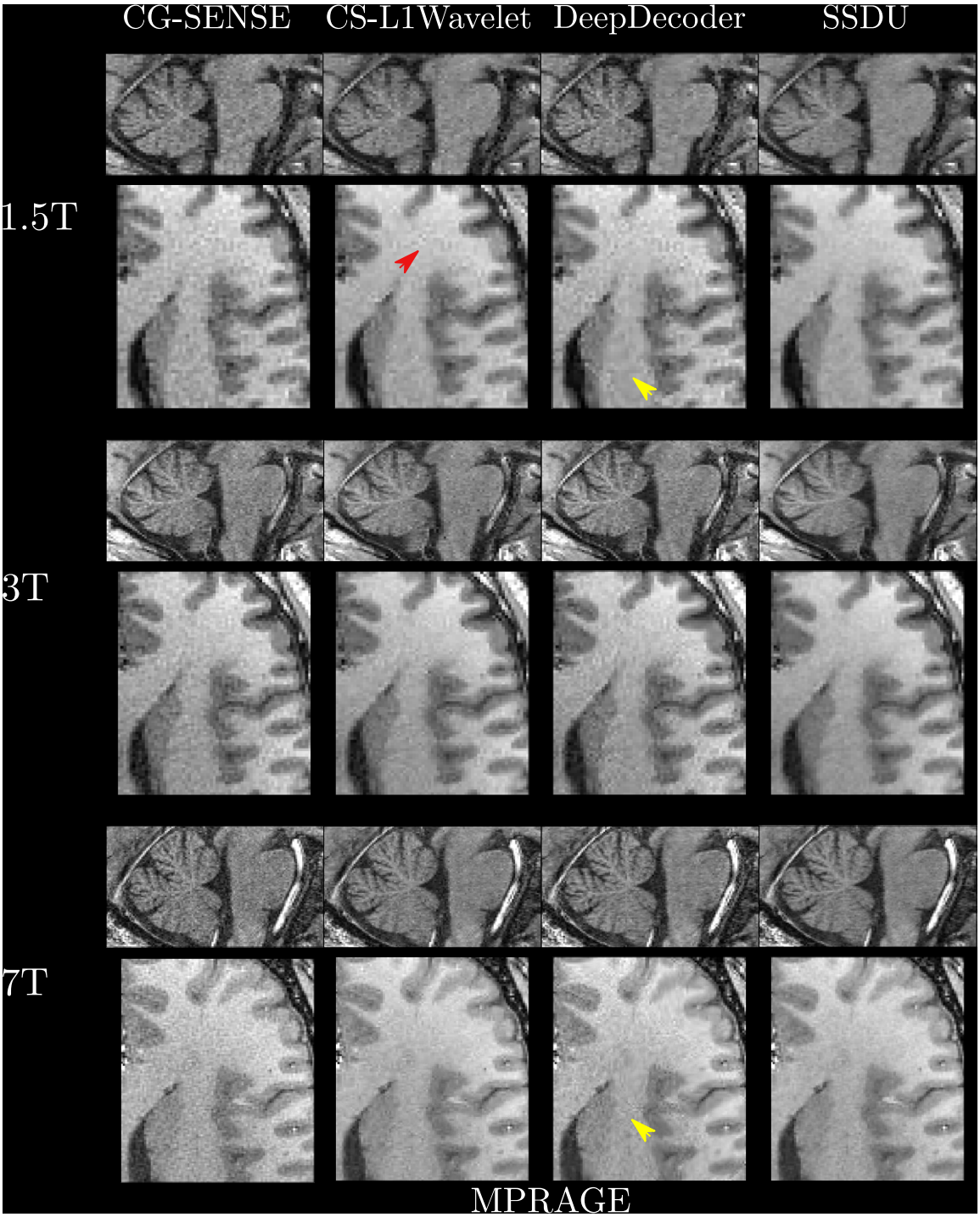


Figure 4.5: Closeups of the prospective reconstructions of MPRAGE scans of the brain at different field strengths; we show closeups of the cerebellum in a sagittal view as well as the left frontal lobe in an axial view. In the axial closeups, the spatially varying smoothness of DeepDecoder is apparent (yellow arrows); furthermore, wavelet artifacts of CS-L1Wavelet can be seen in, for example, the axial closeup at 1.5T (red arrow). In general, we can see that all methods improve in sharpness (as can be seen from the closeups of the corpus callosum) with increasing field strength.

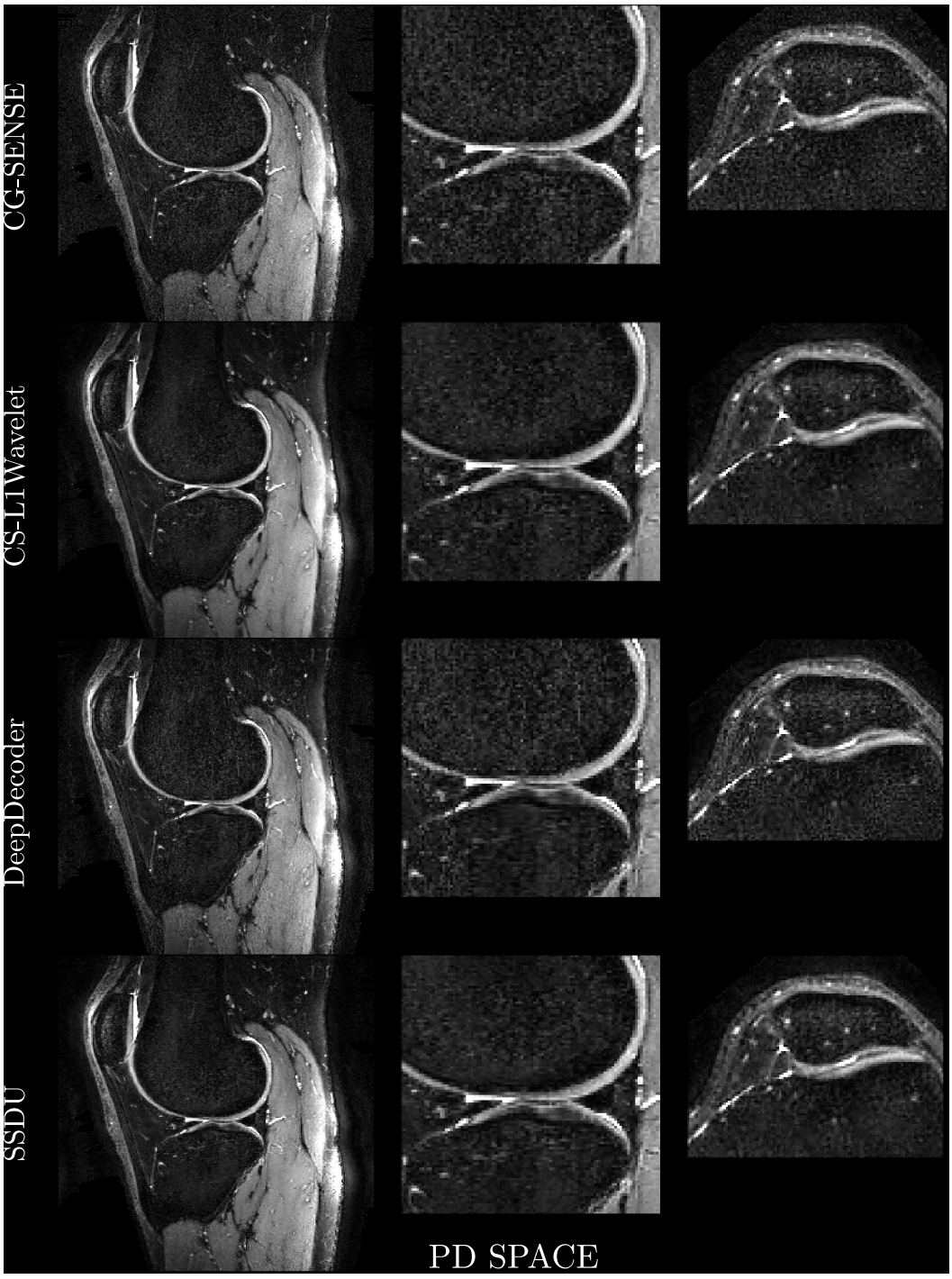


Figure 4.6: Prospective reconstructions from PD SPACE scans of the knee, where we show a sagittal slice as well as closeups on the articular cartilage interface in sagittal (femur) and axial (patella) views. Qualitatively, the main differences are between CS-L1Wavelet/SSDU and CG-SENSE/DeepDecoder, where the former group removes noise better than the latter.

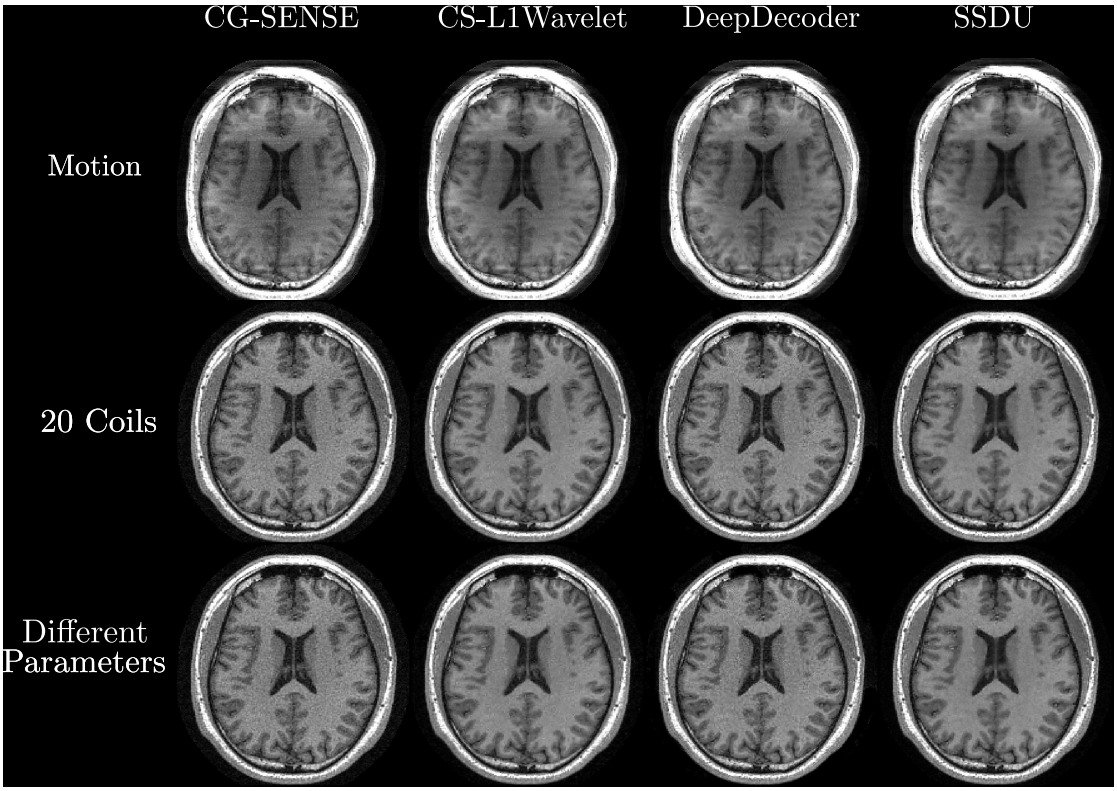


Figure 4.7: Here we show axial brain slice reconstructions from three different perturbations of the MPRAGE sequence: the addition of motion, using 20 coils instead 64 coils, and changing the parameters of the MPRAGE sequence. The images are **not** registered due to interpolation effects from co-registration.

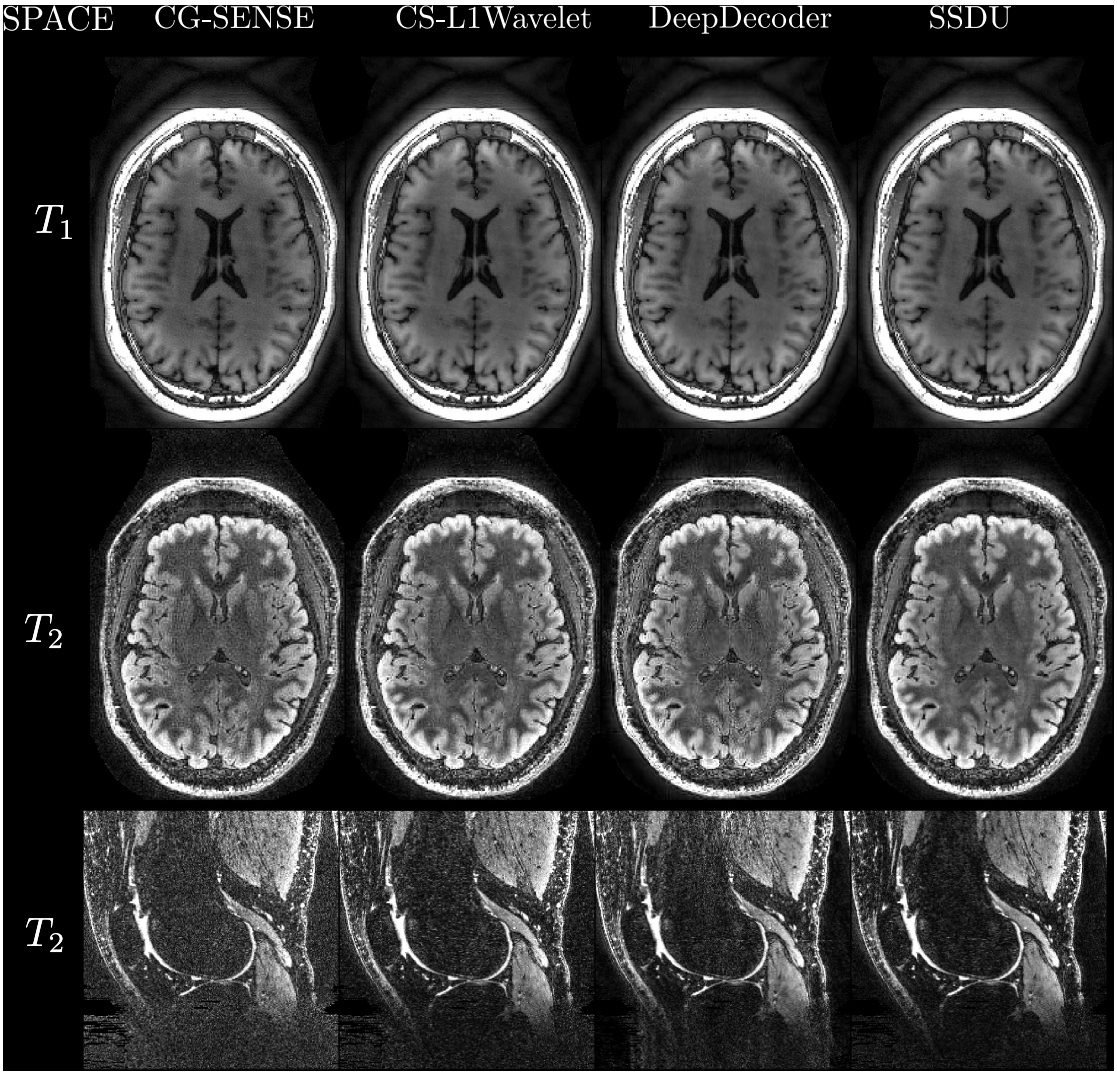


Figure 4.8: Here we show axial brain slices and a sagittal knee slice from the reconstructions from the SPACE acquisitions. The images are **not** registered due to interpolation effects from co-registration.

Figure 4.9: Barplot of the no-reference image metrics averaged over all the subjects/different sequences in the generalizability study (top row). The arrow next to each metric indicates whether higher/lower scores are better. Barplots of the qualitative rating done by the MR Physicists (pooled together) and the radiologist respectively (bottom row). Using the Wilcoxon rank sum test with significance level $\frac{0.05}{6}$ (Bonferroni correction), we found statistically significant differences between all methods with respect to the no-reference image metrics. With respect to the MR physicists the following differences **were not** statistically significant: (DeepDecoder,CS-L1Wavelet,Sharpness),(DeepDecoder,SSDU,Sharpness), all of the aliasing comparisons, and (CS-L1Wavelet, SSDU, Overall Quality). All other comparisons were found to be statistically significant. With respect to the radiologist, all sharpness and aliasing comparisons were found to **not be** statistically significant. In the SNR comparisons, only (CG-SENSE/DeepDecoder vs. SSDU) were found to **be** statistically significant. In the overall quality comparisons, only (CG-SENSE vs DeepDecoder) and (CS-L1Wavelet vs. SSDU) were found to **not be** statistically significant. Overall, the no reference image metrics and human rating agree that CS-L1Wavelet/SSDU exhibit better overall image quality than DeepDecoder/CG-SENSE.

are better, by a significant margin (relatively with respect to the same differences in PSNR/SSIM), than the other methods. The PercDis score or perceptual loss [86] was created precisely because they found this metric better suited for measuring perceptual similarity than PSNR/SSIM. This apparent tradeoff between PSNR/SSIM and perceptual similarity is well-known in the computer vision community, where it is called the perception-distortion tradeoff [94]. This concept has also recently been explored in MR; in [95], the authors train an in-painting network on the Fastmri dataset, and use the features of intermediate layers for quantitative evaluation, producing a perceptual distance tailored for MR images. In [96], the authors propose a new reconstruction method which uses distances in feature space (trained from ground truth MR reconstructions) to better recover textures/perceptual appearance than using just pixel-wise metrics.

4.5.2 Generalizability

We note that as our generalizability study is conducted on prospective reconstructions, which we showed can exhibit distortions relative to fully-sampled reconstructions, it cannot be considered as clinical validation; however, as all methods are affected the same way, this study still can give a good idea of how well each method generalizes. While one might conjecture that generalizability is less of a problem for self-supervised methods, if the parameters/hyperparameters of the methods are tuned for a specific sequence/anatomy as in our case, this could potentially impact the robustness of the methods, as these parameters/hyperparameters are obtained from training/tuning on 3D, brain MPRAGE scans acquired at 3T. This is despite the data consistency inherently embedded in CS-L1Wavelet, DeepDecoder, and SSDU.

Generalizability and robustness of reconstruction methods have been studied in the context of end-to-end, supervised methods for MR reconstruction in [97]–[99]. We briefly summarize some relevant conclusions from these articles. [97] found that that different domain shifts reduced performance more than others (e.g. changing SNR vs. image contrast), and that transfer learning is a viable strategy for handling distribution shifts. [98] found that data consistency is important for robustness, and that at acceleration factor 4, distribution shifts are less of an issue. [99] found that supervised methods are vulnerable to adversarial perturbations, i.e. perturbations constructed such that minimal changes in the input data result in significant changes in the output. In [100], the authors examine the robustness of end-to-end methods, compressed sensing, and variations of Deep Image Prior/DeepDecoder to distribution shifts, adversarial perturbations, and recovery of small features. They found that for both supervised and self-supervised methods, distribution shifts resulted in decreased PSNR/SSIM scores; in addition, the decrease was roughly the same for each method, preserving the ranking of the methods. Furthermore they found that all methods, including self-supervised methods, were vulnerable to adversarial attacks. We note that these works are based on retrospective reconstructions/retrospective sampling from fully-sampled datasets for their validation.

In line with [97], we found that different distribution shifts affected generalization differently;

changing anatomy/contrast worsened the overall image quality rating in comparison to changing the field strength for all methods according to the MR scientists; in contrast, the radiologist found that only SSDU/CS-L1Wavelet worsened. However, as the mean scores in the knee scans for CG-SENSE/DeepDecoder were already 4 (the worst score), the decrease may not reflect any substantial difference in quality. As in [98], data consistency is crucial for the robustness of self-supervised methods as network parameters are trained solely through the modelling/the acquired undersampled data; in particular, we do not see any hallucination that can occur with end-to-end networks without data consistency. Furthermore, we see that as CG-SENSE produces a plausible image with acceleration factor 5, this can explain why distribution shifts were not so troublesome, as the self-supervised methods mainly needed to denoise, rather than recover anatomy/missing high frequency details.

In contrast to [100], our PSNR/SSIM results on the phantoms do not preserve the ranking between methods, although the PercDis results do, approximately. However, the qualitative metrics between distribution shifts over the different brain/knee scans seem to preserve ranking according to the no-reference image metrics/human ratings; this is consistent with PercDis being a better measure for perceptual image quality/similarity than PSNR/SSIM. In addition, the distribution shift in [100] was between two, similar datasets of knee MRI, as compared to our distribution shifts, where we change anatomy, contrast, etc.

For a clinical scenario, it was of interest to see if self-supervised methods could potentially work, without retraining, on other sequences, as retraining after deployment could be impractical. Furthermore, while adversarial perturbations are valuable for studying the input stability of reconstruction methods, they need to be manually constructed for each method and added to the input data. As clinical MR reconstruction is a closed loop, this kind of manual perturbation would require hacking the internal MR computer. Therefore, transfer learning and adversarial perturbations were outside the scope of this work, although from [97], [98], [100], we would expect an increase in image quality from transfer learning and vulnerability to adversarial perturbations for the methods considered in this paper. For example, [82] found, in a retrospective study, that DeepDecoder had different optimal (judged by PSNR/SSIM) hyperparameters for brain vs. knee scans. However, SSDU and CS-L1Wavelet, tuned only on 3T MPRAGE brain data, are able to achieve an overall image quality of fair to good on a diverse dataset.

4.5.3 Ranking Methods through Quantitative Metrics

From a perceptual viewpoint (PercDis score, no-reference image metrics, human rating), SSDU and CS-L1Wavelet performed the best, with an edge to SSDU in the PercDis score/no-reference image metrics. From a pixel-wise metric viewpoint (PSNR,SSIM), DeepDecoder was better than or similar to all methods, as was also found in [100]. CG-SENSE consistently performed the worst or similarly to all methods over all metrics. With respect to validation, both approaches have their advantages and disadvantage; while pixel-wise metrics are the

natural way to compare against a ground-truth, they may not correlate well with the perception of a radiologist. While perceptual metrics may be intuitive, the absence of ground truth can make it less objective. To our knowledge, current state of the art MR image reconstructions are generally not evaluated with perceptual metrics such as PercDis or [95], which require ground truth, or the no-reference image quality metrics. However, given the close correspondence of the image quality metrics/PercDis to the human ratings/perceptual evaluation, as well as other evidence from the literature [87], [88], perceptual metrics could be used as a complement to pixel-wise metrics/human ratings.

4.5.4 Future of Validation

To assist validating future methods, we will make available all the raw data acquired/used in this paper at Zenodo. However, whatever metrics or datasets are used for validating methods, the ultimate test for reconstruction methods is the usefulness to radiologists for reliably diagnosing pathology in comparison to currently used methods [101], [102]. This can imply many things, including fine grained analysis of small textures/details/pathologies as well as tissue specific analysis, requiring novel datasets with extensive annotations by radiologists. [103], [104] are two recent works in this direction, providing datasets with bounding box annotations/pathology annotations to further validate reconstructions. Furthermore, in this chapter we only considered self-supervised methods, as we focused on prospectively undersampled data with little to no fully-sampled counterpart. While previous studies have used retrospectively undersampled data to compare both fully supervised and self-supervised methods, finding them competitive under certain conditions, for the future, ideally validation would include both self-supervised and supervised methods on prospectively undersampled data.

4.6 Conclusion

Rigorous validation is required to introduce new reconstruction algorithms into clinical routines. In this study, validation of prospective reconstructions, generalizability, and different image quality metrics were investigated. The results show that self-supervised image reconstruction methods have potential, but that further development is required to not only improve image quality but also to define a reliable, standardized way of validating new methods. Reliable validation can facilitate quicker translation to the clinical routine, with the ultimate goal of improving patient care.

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Sequence Type	MPRAGE	MPRAGE	MPRAGE	MPRAGE	MPRAGE	MPRAGE	SPACE	SPACE	SPACE	SPACE
Field Strength (T)	1.5	3	7	3	3	3	3	3	3	3
Body Part	Brain	Brain	Brain	Brain	Brain	Brain	Brain	Brain	Knee	Knee
Coils	1Tx/20Rx	1Tx/64Rx	8pTx/32Rx	1Tx/20Rx	1Tx/64Rx	1Tx/64Rx	1Tx/64Rx	1Tx/64Rx	1Tx/18Rx	1Tx/18Rx
Resolution (mm^3)	1.3x1.3x1.2	1x1x1	0.7x0.7x0.7	1x1x1	1x1x1	1x1x1	1x1x1	1x1x1	0.3x0.3x0.6	0.3x0.3x0.6
Field of View (mm^3)	240x240x160	256x240x208	250x219x179	256x240x208	256x240x208	256x240x208	250x250x176	250x250x176	160x160x134	160x160x115
Inversion Time (s)	1	0.9	1.1	0.9	0.9	0.972	-	2.05	-	-
Repetition Time (s)	2.4	2.3	2.5	2.3	2.3	1.93	0.7	7	0.9	1
Echo Time (ms)	3.47	2.9	2.87	2.9	2.9	2.61	11	392	29	108
Echo Spacing (ms)	7.86	6.88	7.8	6.88	6.88	6.28	3.72	3.66	4.84	5.12
Bandwidth (Hz/Px)	180	240	250	240	240	280	630	651	488	416
Turbo Factor	192	198	250	198	198	198	42	220	35	44
Acceleration Factor	4.2	5	5	5	5	5	4	6	7	7
Acquisition Time	1:28 min	1:34 min	2:42 min	1:34 min	1:34 min	1:20 min	3:27 min	3:46 min	4:41 min	3:52 min

Table 4.2: Detailed sequence parameters of all used datasets.

5 Neural Network Enhanced MCMC

The content of the following chapter is based on the postprint version of the article: "Robust biophysical parameter estimation with a neural network enhanced hamiltonian markov chain monte carlo sampler" published in the Proceedings of the International Conference on Information Processing in Medical Imaging [105]. DOI: 10.1007/978-3-030-20351-1_64.

5.1 Introduction

In Chapter 4, we examined methods where self-supervised machine learning methods were embedded in the variational framework introduced in Chapter 2; in this chapter, we propose a method for how self-supervised machine learning can be embedded into a probabilistic framework for solving inverse problems.

5.1.1 Probabilistic Framework and MCMC for Inverse Problems

Recall the setting for inverse problems introduced in Chapter 2: let $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}$, and a function $M : \mathbb{R}^n \to \mathbb{R}^m$. Such that

$$M(\mathbf{x}) = \mathbf{y} + \mathbf{n} \tag{5.1}$$

Let $\mathbf{n} \sim \mathcal{N}(0, \sigma)$ i.e. Gaussian noise with standard deviation σ. Then we can construct a probabilistic framework where we view the problem as recovering the posterior probability distribution of $\mathbf{x}$ given the measurements $\mathbf{y}$ [106].

Using Bayes' theorem, the posterior probability distribution of $\mathbf{x}$ given the measurements $\mathbf{y}$ is proportional to the product of the likelihood function and the prior on $\mathbf{x}$:

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x}), \tag{5.2}$$

where,

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(M(\mathbf{x}), \sigma^2) \tag{5.3}$$

$$= \frac{1}{\sqrt{(2\pi)^m \sigma^{2m}}} \exp(-\frac{\|\mathbf{y} - M(\mathbf{x})\|_2^2}{2\sigma^2}), \tag{5.4}$$

where we assume that the elements of $\mathbf{y}$ are statistically independent. The prior distribution $p(\mathbf{x})$ can encode constraints such as sum constraints or upper/lower bounds through, for example, Dirichlet and uniform distributions [107]. The proportionality constant is a normalizing factor. An immediate candidate to recover $\mathbf{x}$ is the maximum a posteriori (MAP) point estimate

$$\mathbf{x}^{MAP} = \arg\max_{\mathbf{x}'} p(\mathbf{x}'|\mathbf{y}) \tag{5.5}$$

In fact, the MAP estimate links the probabilistic framework outlined here and the variational framework in Chapter 2. We can see this by first noting that maximizing the posterior probability is equivalent to maximizing the logarithm of the posterior probability. Furthermore, all factors independent of $\mathbf{x}$, such as the the normalizing factor, can be omitted after taking the logarithm. Then

$$\mathbf{x}^{MAP} = \arg\max_{\mathbf{x}'} \log(p(\mathbf{x}'|\mathbf{y})) \tag{5.6}$$

$$= \arg\max_{\mathbf{x}'} -\frac{\|\mathbf{y} - M(\mathbf{x}')\|_2^2}{2\sigma^2} + \log(p(\mathbf{x}')) \quad = \arg\min_{\mathbf{x}'} \|\mathbf{y} - M(\mathbf{x}')\|_2^2 - \lambda \log(p(\mathbf{x}')) \tag{5.7}$$

At this point, one can identify $-\log(p(\mathbf{x}'))$ as exactly the regularization function from the variational framework, since their purpose is the same; minimizing the former encourages solutions $\mathbf{x}$ which are more probable according to the assumptions of prior knowledge on $\mathbf{x}$, which is exactly the point of the regularization function. Hence, we have

$$\mathbf{x}^{MAP} = \arg\min_{\mathbf{x}'} \|\mathbf{y} - M(\mathbf{x}')\|_2^2 + \lambda R(\mathbf{x}') \tag{5.8}$$

Therefore, the MAP estimate (in the case of Gaussian noise) reduces to the variational solution. However, there are two potential problems with this point estimate. First, the general problems of uniqueness and feasibility of optimization, i.e. finding the MAP estimate. Second, the underlying assumption of this point estimate is that the mode is a good representation of the underlying probability distribution. Intuitively, we can see the truth of this assumption for many commonly used distributions in three or less dimensions e.g. normal or exponential. However, this assumption can fail as the dimensionality and complexity of the distribution increases. That is, define (loosely) the typical set to be the set of points in parameter space containing most of the probability mass. Due to the geometry of high dimensional spaces [108] and the concentration of measure phenomenon [108], [109], the typical set in high dimensions tends to lie in narrow bands of parameter space further and further away from the mode of

the distribution. Hence, a mode point estimate can lead to spurious results. One approach to handle these issues is to first characterize the posterior distribution with Markov Chain Monte Carlo (MCMC) techniques [110] by sampling from the posterior distribution. One can then, as an example, use the mode, mean, median, etc. of the marginal posterior distributions of the elements of $\mathbf{x}$ for the parameter estimate. In this chapter, we use the expectation of each parameter over its marginal posterior, approximated by

$$\mathbf{x}^* \approx \frac{1}{N}\sum_{i}^{N}\mathbf{x_i}, \tag{5.9}$$

where the subscript i denotes one of the N samples.

Two examples of highly nonlinear inverse problems, whose difficulties are well known, are multi-compartment T_2 relaxometry and multi-compartment diffusometry; in this chapter, we test our proposed method on an inverse problem which combines the two.

5.1.2 Multi-Compartment T_2 Relaxometry/Diffusometry

In contrast (no pun intended), to contrast images as introduced in Chapter 3, MRI can be used to produce images which are quantitatively meaningful; the magnitude of each image voxel can correspond to measuring something quantitatively. An example of a quantitative measurement is the spin-spin relaxation time T_2, which is the physical decay time of the transverse magnetization; a quantitative T_2 map would be an image where the magnitude of each voxel is the average T_2 of the underlying spins in that voxel.

For example, recall the exponential solution for the transverse magnetization from the Bloch Equation:

$$\mathbf{M}_\perp(t) = \mathbf{M}_\perp(0)\exp(\frac{t}{T_2}) \tag{5.10}$$

If we were able to acquire a sequence of measurements ($\mathbf{M}_{perp}(T_E^n)$, we could obtain the T_2 using a log-linear regression for example. We use the variable T_E to denote the sampled times as most methods for acquiring such measurements use so-called **spin or gradient echo** sequences, with the acquired sample times being called echo times (T_E) This can approximately, for example, be done by acquiring a sequence of N contrast images, $\mathbf{x}^i$ such that

$$\mathbf{x}^i \approx \mathbf{M}_\perp(x, y, t=0)\exp(\frac{T_E^i}{T_2(x,y)}) \tag{5.11}$$

where T_E^i denotes when in the time evolution of the spins when the signal was acquired. For example see Figure 5.1, where we show a sequence of scans with varying TEs.

Given a voxel indexed by (i, j), let $\mathbf{y}_{i,j}^{T_2} = (\mathbf{x}_{i,j}^1, \mathbf{x}_{i,j}^2, \ldots, \mathbf{x}_{i,j}^N)$. Then for each voxel, we have the

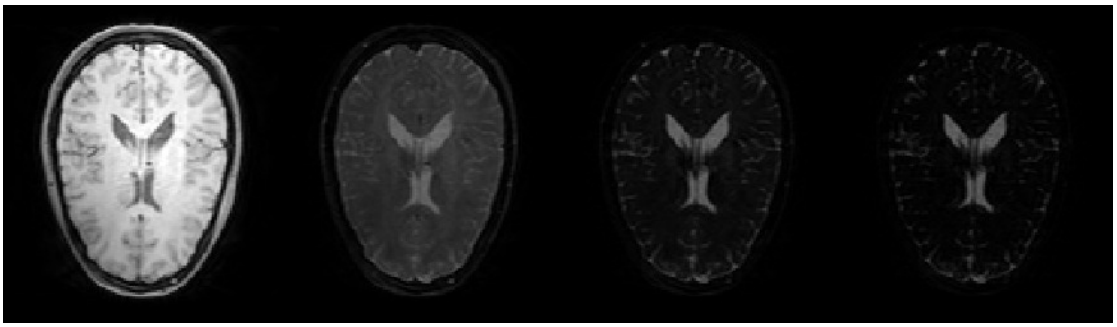


Figure 5.1: Here we show a sequence of T_2 weighted axial MR images of the brain, where the TE increases from left to right. Note that larger TE implies greater signal decay, which is clear in the images.

inverse problem $(M, \mathbf{y}_{i,j}^{T_2}, (A, T_2^{i,j}))$ where

$$M(A, T_2^{i,j}) = (A\exp(\frac{T_E^1}{T_2^{i,j}}), A\exp(\frac{T_E^2}{T_2^{i,j}}), \ldots, A\exp(\frac{T_E^N}{T_2^{i,j}})) \tag{5.12}$$

where A is an overall constant which includes the net magnetization, and $T_2^{i,j}$ denotes the T_2 at the voxel indexed by (i, j). This problem can be made into a linear inverse problem by taking the logarithm of the measurements/the model.

Hence, in practice, one acquires N different acquisitions of the same subject (corresponding to the different T_E^n), then reconstructs N corresponding images using the MR image model or modifications thereof. Then for each voxel there are N image intensities which can be fit to a model to find the T_2 in the voxel. Therefore, while contrast images are reconstructed as a whole from the k-space, quantitative maps can be reconstructed independently and voxel-wise using a sequence of contrast images.

Compartment Models

As previously stated, the T_2 can be found voxel-wise. However, voxels can contain multiple different spin populations, each with a different T_2 due to, for example, different local environments. Hence, if one calculates the T_2 through a log-linear regression, the resulting T_2 of the voxel will be some average of the underlying T_2s in the voxel.

As an example, suppose that there are k different spin populations with different spin-spin relaxation times in a voxel. Further assume that these spin populations do not interact/mix/exchange spins with each other. We refer to each spin population as a **compartment** as with these assumptions, the transverse magnetization of the voxel can be written as a simple linear

combination over the compartments:

$$\mathbf{M}_{\perp}(t) = \mathbf{M}_{perp}(0) \sum_{i=1}^{k} w_i \exp(\frac{t}{T_2^i}) \tag{5.13}$$

$$\sum_i w_i = 1 \tag{5.14}$$

where w_i, T_2^i are the fraction of spins and the T_2 associated to the ith compartment. The associated inverse problem of recovering A, w_i, T_2^i (constructed similarly as in the single spin case) is significantly more ill-posed and complex than in the single population cases as now the problem can no longer be converted to a linear form, as the model is a sum of exponentials.

Diffusometry

In Chapter 3, we also introduced the Bloch-Torrey equation, which takes into account the diffusion of spins; analogously to the T_2, one can estimate the diffusion properties of a voxel from a sequence of contrast images, and the different spin populations can have different diffusive properties due to differences in the local environment. While we will use the analogous inverse problem to recover the diffusion properties of different compartments, we omit the derivation.

Multi Echo Spherical Mean Technique Model (MESMT)

MRI Diffusometry and T_2 relaxometry can be combined into a multi-modal analysis which jointly estimates diffusivities, T_2's, and water volume fractions of different compartments. The extended spherical mean technique (SMT) framework introduced by [111] is one example of this, generalizing the diffusion MRI model SMT [112] by including the effects of changing the echo time T_E in the acquisition on the MRI signal and using the additional information to simultaneously estimate the T_2's and diffusivities of the compartments; in particular we focus on an application targeting brain white matter, where we define three compartments/populations of spins: intra-axonal, extra-axonal, and cerebrospinal fluid (CSF). The acquired signal in a voxel is a function of the diffusion weighting b (analogous to T_E in T_2 acquisitions) and T_E. For given b, T_E, the model for the signal is

$$\mathscr{M}(T_E, b, \mathbf{x}) = v_I \exp(\frac{-T_E}{T_2^I}) \frac{\sqrt{\pi}\,\mathrm{erf}(\sqrt{b\lambda_{\parallel}})}{2\sqrt{b\lambda_{\parallel}}} \tag{5.15}$$

$$+ v_E \exp(\frac{-T_E}{T_2^E}) \exp(-b\lambda_{\perp}) \frac{\sqrt{\pi}\,\mathrm{erf}(\sqrt{b(\lambda_{\parallel}-\lambda_{\perp})})}{2\sqrt{b(\lambda_{\parallel}-\lambda_{\perp})}} \tag{5.16}$$

$$+ v_{CSF} \exp(\frac{-T_E}{T_2^{csf}}) \exp(-bD_{csf}), \tag{5.17}$$

where v_I, v_E, v_{CSF} are the volume fractions of the intra-axonal, extra-axonal, and cerebrospinal fluid (CSF) compartments respectively, T_2^I, T_2^E, T_2^{csf} are the respective T_2s, and $\lambda_\parallel, \lambda_\perp$ are the parallel and perpendicular diffusivities. We note that the model function M is then concatenation of $\mathscr{M}$ over the different T_E, b considered. The likelihood of this model is constructed as in the introduction. In the fitting, we fix the values of $T_2^{csf} = 2s$ and $D_{csf} = 0.003\frac{mm^2}{s}$ at those of free water [112], [113], but we allow v_{CSF} to be free. Therefore $\mathbf{x}$ in this inverse problem is made up of the free volume fractions, T_2s and diffusivities.

5.1.3 Contributions

The main contributions of this chapter are, first, to extend a Hamiltonian MCMC sampler parametrized with neural networks proposed in [114]. This sampler, called L2HMC, proposes to use a neural network modification to Hamiltonian dynamics to propose new samples; in particular, the network is trained, in a **self-supervised way**, to maximize the expected distance between sucessive samples. This approach is an example of how a model-driven method (Hamiltonian MCMC sampling) can be augmented with a data-driven method (neural network parametrization), without requiring ground truth. We modify the loss function in [114] to balance acceptance probability and mixing such that both fast mixing and stable exploration of problematic regions of state space are possible; in particular, we enforce a weak conformity to standard Hamiltonian dynamics, which does not exist in [114], through an acceptance probability term in the loss function, which we show leads to more stable sampling and faster mixing than [114]; evidence of the latter is shown on a toy example. In this way, our MCMC sampler shows how incorporating more model-driven information (Hamiltonian dynamics) can help with learning to sample. We note that as in Chapter 4, our proposed approach is not specific to any model/inverse problem.

As our intent was to create methods for MR imaging, our second main contribution is to apply our extended sampler to solve the relatively complex and ill-posed inverse problem described in the previous section defined by the MESMT model, for which there is no ground truth available. We provide a proof of concept using synthetic data generated from realistic prior knowledge, comparing to a least squares fitting and application of two state of the art Hamiltonian samplers (L2HMC and NUTS).

5.2 Related Work

5.2.1 Hamiltonian Markov Chain Monte Carlo

In the following, we denote the posterior distribution from which we want to sample as $p(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^n$ being the state variables. MCMC methods sample from the posterior by generating a sequence of samples where each new sample $\mathbf{x}_t$ is generated from the previous sample $\mathbf{x}_{t-1}$ according to a transition distribution $T(\mathbf{x}_t|\mathbf{x}_{t-1})$ [115]. In order for the posterior to be the unique distribution to which this sequence converges, the transition distribution must satisfy

ergodicity, which can usually be safely assumed, and an invariance property which is usually shown by proving a property called detailed balance $p(\mathbf{x}_t)T(\mathbf{x}_{t-1}|\mathbf{x}_t) = p(\mathbf{x}_{t-1})T(\mathbf{x}_t|\mathbf{x}_{t-1})$.

One well known way to construct a transition satisfying detailed balance called the Metropolis-Hastings algorithm [110] is as follows: given a proposal distribution $q(\mathbf{x}'|\mathbf{x}_{t-1})$, sample a candidate $\mathbf{x}'$; then, accept $\mathbf{x}'$ with probability $A(\mathbf{x}'|\mathbf{x}_{t-1}) = min(1, \frac{p(\mathbf{x}')q(\mathbf{x}_{t-1}|\mathbf{x}')}{p(\mathbf{x}_{t-1})q(\mathbf{x}'|\mathbf{x}_{t-1})})$. If accepted, $\mathbf{x}_t = \mathbf{x}'$. If rejected, $\mathbf{x}_t = \mathbf{x}_{t-1}$. However, even if a sampler satisfies these properties, the convergence is only proven asymptotically [115]. The typical procedure is to first have a burn-in stage where the sampler is run for some amount of steps in order for it to converge. Then, the actual sampling begins, with the burn-in samples being discarded [115].

For efficient exploration, the samples should ideally be uncorrelated, which can be accomplished by large distances between samples in the sample space, i.e. mixing. Autocorrelation analysis using multiple chains of samples can be used as a rough measure of how many samples are necessary. We emphasize that a balance must be found between the acceptance probability and the mixing; acceptance probabilities which are very high can mean the samples are very close/correlated and large distances between samples can lead to only a small number of samples being accepted. One powerful MCMC method which scales with the dimensionality and complexity of the posterior is Hamiltonian MCMC (HMCMC) [116]. In HMCMC, one generates proposal samples by integrating along trajectories of a Hamiltonian dynamical system constructed from combining the posterior distribution of interest with a momentum distribution. This is then followed by the Metropolis acceptance step to yield a new sample. Formally a joint distribution is constructed with state variables $(\mathbf{x}, \mathbf{p})$:

$$p^H(\mathbf{x}, \mathbf{p}) \propto \exp(-U(\mathbf{x}) - K(\mathbf{p})), \tag{5.18}$$

$$p(\mathbf{x}) \propto \exp(-U(\mathbf{x})), \tag{5.19}$$

$$K(\mathbf{p}) = \frac{1}{2}\mathbf{p}^T\mathbf{p}, \tag{5.20}$$

where we omit a normalizing constant and $\mathbf{p}$ are the momentum variables which are added. This form is motivated from statistical physics by the canonical distribution of energy states of a system, where U and K denote the potential and kinetic energy respectively, and the Hamiltonian (total energy) is $H = U + K$ [116]. HMCMC samples from $p^H(\mathbf{x}, \mathbf{p})$, and we can obtain the marginal distribution of $\mathbf{x}$ from the samples. H defines a dynamical system, which is a set of differential equations used to evolve $\mathbf{x}, \mathbf{p}$ forward in time from an initial sample.

In practice, these equations are integrated numerically, characterized by a step size ϵ and a number of steps L such that $L\epsilon$ is the time period over which a sample trajectory is evolved. The most common numerical scheme is the leapfrog scheme, which we write below for one time step with initial condition $(\mathbf{x}, \mathbf{p})$ and result $(\mathbf{x}', \mathbf{p}')$.

$$\mathbf{p}^{\frac{1}{2}} = \mathbf{p} - \frac{\epsilon}{2}\partial_x U(\mathbf{x}), \mathbf{x}' = \mathbf{x} + \epsilon\mathbf{p}^{\frac{1}{2}}, \mathbf{p}' = \mathbf{p} - \frac{\epsilon}{2}\partial_x U(\mathbf{x}'). \tag{5.21}$$

Given an initial $\mathbf{x_0}$, an initial momentum $\mathbf{p_0}$ is sampled from a distribution, usually a standard

Gaussian [115]. The proposed sample from running the dynamics, $(\mathbf{x}', \mathbf{p}')$, is then accepted in a Metropolis-Hastings step with probability $\alpha = min(1, \frac{\exp(-U(\mathbf{x})-\frac{1}{2}\mathbf{p}^T\mathbf{p})}{\exp(-U(\mathbf{x_0})-\frac{1}{2}\mathbf{p_0}^T\mathbf{p_0})})$. Ideally, this procedure is then repeated until the convergence of the samples to the distribution, with the output of each proposal becoming the new initial sample. The main advantage of HMCMC is that it generally proposes samples which are far away from the initial sample, thus efficiently exploring the posterior, while maintaining reasonable acceptance probabilities [115]. A state of the art HMCMC sampler called the No U Turn Sampler (NUTS) [117] improves on standard HMCMC by adaptively tuning L, ϵ to manage the distance between samples and acceptance probability. HMCMC can perform poorly in certain circumstances, in particular, in highly curved sample spaces such as those that might arise in the posteriors derived from parameter estimation of complex models [118].

5.2.2 L2HMC

Levy et. al. [114] recently proposed a framework called L2HMC which parametrizes the standard HMCMC sampler with a neural network and maximizes the expected distance between samples through minimization of a loss function which rewards large expected squared distances between samples. Furthermore, the parametrization is carefully tailored to preserve detailed balance and have a tractable Jacobian for the correction of the acceptance probability due to the potential non-volume preserving dynamics. The algorithm of L2HMC is structurally similar to standard HMCMC, but modifications are made to the proposal stage. First, for each step t, $1 \leq t \leq L$, a random binary mask $m_t \in \{0,1\}^n$ is constructed such that approximately half of the entries of the mask are 1. The conjugate mask is denoted as m_t^c. Instead of updating $\mathbf{x}$ in one step according to the classical algorithm, the update is split into two steps each updating only the variables of $\mathbf{x}$ corresponding to m_t, m_t^c separately. These are denoted as $\mathbf{x}_{m_t} = \mathbf{x} \odot m_t$ and $\mathbf{x}_{m_t^c} = \mathbf{x} \odot m_t^c$ respectively, where $\odot$ is the component-wise multiplication operator. Each update equation is modified with scaling factors for each term depending on only variables which are not being updated. Concretely, let $\zeta_1 = (\mathbf{x}, \partial_x U(\mathbf{x}'), t)$. Then $\mathbf{p}$ is first updated according to

$$\mathbf{p}^{\frac{1}{2}} = \mathbf{p} \odot \exp(\frac{\epsilon}{2} S_p(\zeta_1)) - \frac{\epsilon}{2} \partial_x U(\mathbf{x}) \odot \exp(\epsilon Q_p(\zeta_1)) + T_p(\zeta_1), \tag{5.22}$$

where $S_p, Q_p, T_p : \mathbb{R}^{2n+1} \to \mathbb{R}^n$ are scaling functions parameterized by a neural network. Let $\zeta_2 = (\mathbf{x}_{\mathbf{m_t^c}}, \mathbf{p}, t)$ and $\zeta_3 = (\mathbf{x}^{\frac{1}{2}}_{\mathbf{m_t}}, \mathbf{p}, t)$. Then $\mathbf{x}$ is updated according to

$$\mathbf{x}^{\frac{1}{2}} = \mathbf{x}_{\mathbf{m_t^c}} + m_t \odot \left[\mathbf{x} \odot \exp(\epsilon S_x(\zeta_2)) + \epsilon(\mathbf{p}^{\frac{1}{2}} \odot \exp(\epsilon Q_x(\zeta_2)) + T_x(\zeta_2)) \right], \tag{5.23}$$

$$\mathbf{x}' = \mathbf{x}_{\mathbf{m_t}} + m_t^c \odot \left[\mathbf{x} \odot \exp(\epsilon S_x(\zeta_3)) + \epsilon(\mathbf{p}^{\frac{1}{2}} \odot \exp(\epsilon Q_x(\zeta_3)) + T_x(\zeta_3)) \right], \tag{5.24}$$

where $S_x, Q_x, T_x : \mathbb{R}^{2n+1} \to \mathbb{R}^n$ are also scaling functions parameterized by a neural network. Finally, let $\zeta_4 = (\mathbf{x}', \partial_x U(\mathbf{x}'), t)$:

$$\mathbf{p}' = \mathbf{p}^{\frac{1}{2}} \odot \exp(\frac{\epsilon}{2} S_p(\zeta_4)) - \frac{\epsilon}{2} \partial_x U(\mathbf{x}') \odot \exp(\epsilon Q_p(\zeta_4)) + T_p(\zeta_4). \tag{5.25}$$

These learned scaling functions, structured as a two layer neural network, can allow the sampler to learn, for example, how to carefully navigate regions of high curvature in the parameter space rather than having to manipulate ϵ and L to accomplish this. As in NUTS [117], the time reversed version of the above dynamics can also be used to propose samples, and L2HMC takes a random combination of the forward and backward dynamics proposal as the final proposal [114]. Let θ be the vector of parameters of the above functions. After each complete cycle of proposal and acceptance, the loss function is optimized using Adam [81]. Concretely, let $\xi = (x, p)$ be the initial sample, and $\xi' = (x', p')$ be the sample after the acceptance step. Let $\delta(\xi, \xi') = \|x - x'\|_2^2$ and $A(\xi', \xi)$ denote the acceptance probability. Then the loss function $\mathscr{L}(\theta)$ used is

$$\mathscr{L}(\theta) = \mathbb{E}_{p(\xi)} \left[-\frac{\delta(\xi, \xi') A(\xi', \xi)}{\lambda^2} + \frac{\lambda^2}{\delta(\xi, \xi') A(\xi', \xi)} \right], \tag{5.26}$$

where the expectation is taken over the batch of samples over which the training is taking place. λ is the typical length scale of the distribution, which Levy et. al. set in the case of a multivariate normal distribution, as the smallest standard deviation in the covariance matrix. For simplicity, in eq. 13, we omit an additional term with identical form as above [114] designed to enhance burn-in by using an arbitary proposal distribution. Fig 5.2 shows a flowchart of the algorithm of sampling with the neural network parametrization.

5.3 Methods

5.3.1 Neural Network Enhanced Hamiltonian MC (NNEHMC)

The first contribution of this chapter is to extend L2HMC by augmenting the loss function to balance acceptance probability and the distance between samples. Let $A_{HMC}(\xi', \xi)$ denote the acceptance probability used in standard HMCMC. We introduce the loss function

$$\mathscr{L}^{NNEHMC}(\theta) = \mathbb{E}_{p(\xi)} \left[-\delta(\xi, \xi') A(\xi', \xi) - \beta A_{HMC}(\xi', \xi) \right]. \tag{5.27}$$

We removed the reciprocal distance term as it did not meaningfully change the dynamics of the sampling in the distributions we considered. Further, we do not integrate the time reversed dynamics in our sampling. We argue that this form of loss function more faithfully and naturally enhances the desirable properties of Hamiltonian dynamics. In theory, Hamiltonian dynamics preserve energy along trajectories; hence, since the probability of a sample is proportional to $\exp(-H)$, the acceptance probability is always 1 [115]. However, the introduction of numerical integration causes violation of this property; nonetheless HMCMC still,

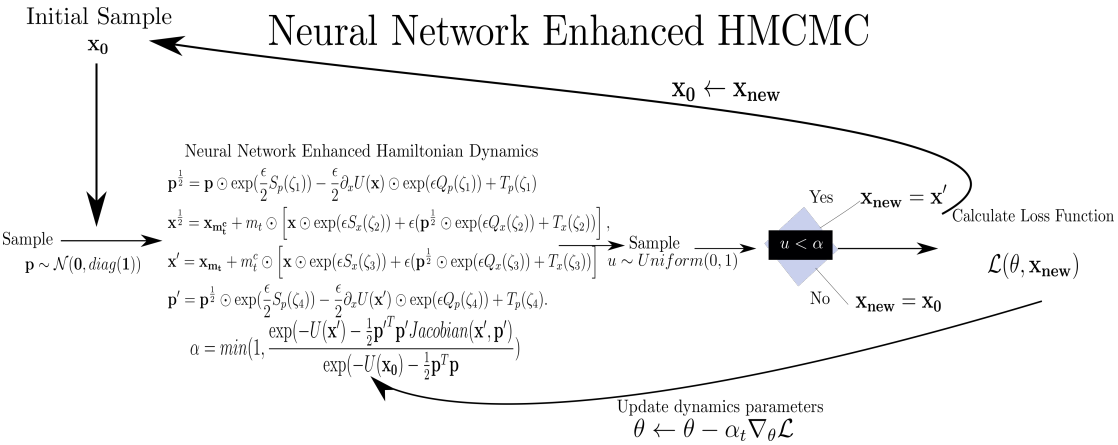


Figure 5.2: Flowchart of the training algorithm for neural network parametrization of HMCMC. The components of the algorithm are very similar to standard HMCMC; however, the differences lie in the altered dynamics/proposal stage and the update of the neural network parameters after each step. When sampling, the network parameters are fixed at the last training values.

generally, provides high acceptance rates, with additional tuning possible through changing ϵ or L. One can view this tuning as reducing the error of the leapfrog scheme such that the numerical integration gets closer and closer to the theoretical Hamiltonian dynamics with its property of preserving energy. However, the dynamics of L2HMC is no longer a numerical approximation of Hamiltonian dynamics due to the scaling terms. Hence, while it is valid as an MCMC sampler, there is no theoretical basis for the sampler to produce samples with high acceptance probabilities, which are largely independent of the squared distance as in standard HMCMC.

As a way of both inducing the sampler to remain close to Hamiltonian dynamics and balancing the acceptance probability and mixing, we add the negative standard HMCMC acceptance probability in the loss function, with the parameter β enforcing the tradeoff between it and the negative expected squared distance. We argue that this loss function can lead to two desirable properties. First, it could lead to faster mixing and faster convergence than in L2HMC since, from the beginning, it can balance learning the standard, approximately energy preserving Hamiltonian dynamics with opportunities to move great distances. One can interpret the additional term as enforcing approximate conformity, in some sense, to Hamiltonian dynamics, mediated by β. Second, crucial for parameter estimation, we argue that this term helps to keep the sampler stable when exploring high curvature regions. In these regions, the acceptance probability can drop to zero easily due to large distance steps and numerical issues can develop [118]. The acceptance probability of the neural network parametrized sampler differs from the classical acceptance by the Jacobian of the new, scaled dynamics, which is identically 1 in the standard case. Hence, if the standard acceptance probability is the dominant term, it can still enforce high acceptance probabilities for the sampler. We thus treat the neural network as an enhancement that allows the sampler to learn "approximate"

Hamiltonian dynamics which can balance and enhance the desirable properties of HMCMC while learning to minimize its weaknesses. We henceforth refer to our sampler as Neural Network Enhanced Hamiltonian MC (NNEHMC).

In the results, we compare the performance of NNEHMC and L2HMC on a toy distribution also tested in [114]. The distribution is a strongly correlated 2-D Normal distribution with mean zero, and a covariance matrix obtained from $diag(100, 0.1)$ rotated by 45 degrees. For both samplers, we use the same $\epsilon = 0.1, L = 10$, initialize with the same 200 samples, train in batches of 200 samples for 5000 steps, then fix the neural network parameters and sample 200 chains for 2000 steps using the trained sampler[114]. We tune β in NNEHMC by looking at the autocorrelation analysis and the acceptance probabilities. We set $\lambda = 0.1$ as is done in [114]. We compare the two samplers by the autocorrelation of the samples as well as the effective sample size derived from the autocorrelation, which can be seen as a measure of how many of the samples are "useful" for inference [115].

5.3.2 Biophysical Parameter Estimation

The second contribution of this chapter to apply NNEHMC to biophysical parameter estimation in a recently proposed MRI model, the Multi Echo Spherical Mean Technique (MESMT) [111], which was described in the introduction.

Multi Echo Spherical Mean Technique (MESMT): Experimental Setup

We simulate three datasets from the MESMT model using three different $T_E's = 50, 75, 100ms$, with three $b = 300, 2150, 4000 s/mm^2$ values per dataset, and fit them simultaneously. The ground truth parameters are as follows: $v_I = 0.5, v_E = 0.3, v_{CSF} = 0.2, \lambda_{\parallel} = 0.0015 \frac{mm^2}{s}, \lambda_{\perp} = 0.0002 \frac{mm^2}{s}, T_2^I = 140ms, T_2^E = 70ms$. Since the volume fractions must sum to one, we use a 3D, symmetric Dirichlet prior for the volumes: $(v_I, v_E, v_{CSF}) \sim \textbf{Dir}(1.0, 1.0, 1.0)$. We can bound the T_2's and diffusivities based on prior physical knowledge [112], [113], using uniform priors as follows:
$T_2^I \sim \mathcal{U}(5ms, 200ms)$, $T_2^E \sim \mathcal{U}(5ms, 100ms)$, $\lambda_{\parallel} \sim \mathcal{U}(0.0005 \frac{mm^2}{s}, 0.003 \frac{mm^2}{s})$,
$\lambda_{\perp} \sim \mathcal{U}(0.0001 \frac{mm^2}{s}, 0.0005 \frac{mm^2}{s})$. We generated one hundred signals from the ground truth parameters by adding one hundred realizations of Gaussian noise with a standard deviation of $\sigma = \frac{1}{120}$. We simulated many instances of a typical diffusion acquisition using Dmipy [119] with a mean SNR of 20 on the b_0 data, then performed spherical averaging on each instance. The standard deviation of the resulting signals over the instances was estimated to be around $\frac{1}{120}$, which motivates our setting of σ. We then estimated the parameters over each signal using NUTS, L2HMC, and NNEHMC within the Bayesian framework described above. We also show a fitting using constrained least squares (LSQ). We imposed the same constraints in both the probabilistic and deterministic fittings. We initialize NUTS with a variational inference estimate [120], and use 1000 samples for burn-in and 1000 samples for inference. We initialize L2HMC and NNEHMC with the first 50 samples of the NUTS burn-in, train on batches of 50

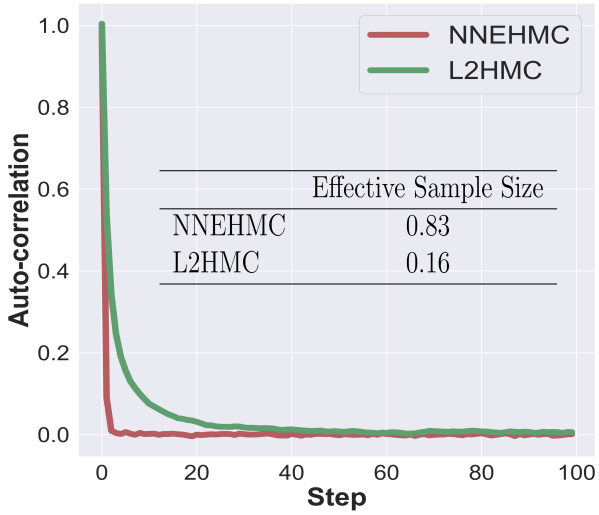


Figure 5.3: Plot of the average autocorrelation of 200 chains of length 2000 for L2HMC and NNEHMC with the corresponding effective sample size. The autocorrelation and effective sample size are calculated as in [114]. We see that NNEHMC mixes faster in sampling steps and has a larger effective sample size.

samples for 1000 steps, then fix the parameters of the network and use the trained sampler to generate 1000 samples for inference. We set $\lambda = \sigma$, since it is roughly the length scale of the distribution. In the results, we report the relative absolute error as follows: letting g denote the ground truth parameter and e as the estimate, the relative absolute error is computed as $|g-e|/g$. We note that we scale b by 10e−2 and the diffusivities by 10e2 in the sampling and results.

5.4 Results and Discussion

5.4.1 Strongly Correlated Gaussian

In Fig 5.3, we show the average autocorrelation of the samples over 50 chains from sampling the strongly correlated Gaussian as a function of steps in the chain as well as a table with the effective sample sizes derived from the autocorrelation. We note that NNEHMC mixes faster and has an effective sample size almost eight times larger than that of L2HMC. Further, on the same computer, NNEHMC requires 179s of computation time while L2HMC requires 1561s. This is mostly because NNEHMC does not use the time reversed dynamics. In cases with tractable distributions and derivatives one can also speed up the sampling by using GPU computation [121].

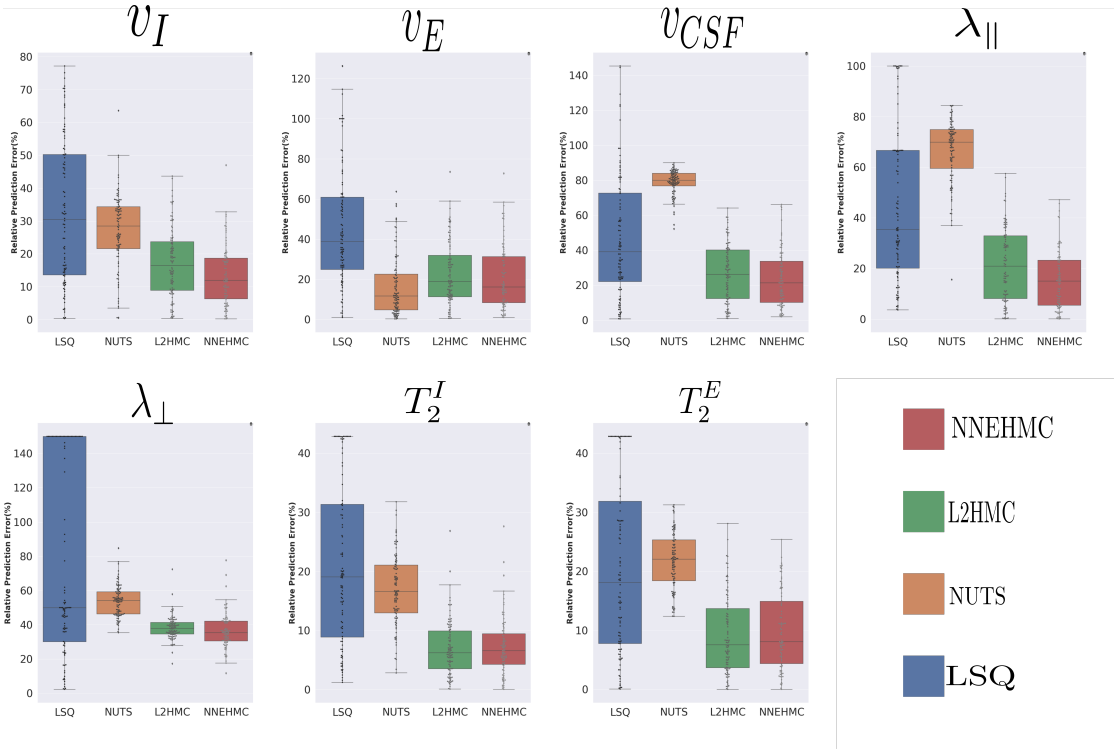


Figure 5.4: Box plots of relative absolute errors from ground truth using least squares (blue), NUTS (orange), L2HMC (green), and NNEHMC (red). We note that in general, NNEHMC has the lowest mean error and variance. Further, NUTS has significant issues in the estimation of $\lambda_{\|}$ and the volume fractions, which is not observed in NNEHMC or L2HMC.

5.4.2 Multi Echo Spherical Mean Technique (MESMT)

In Fig. 5.4, 5.5, we show the relative absolute error and an example of the marginal posterior probability distributions produced by the MCMC samplers.

We can see that, in general, the MCMC samplers are more accurate and precise than the least squares fitting. However, we see that NNEHMC and L2HMC significantly outperform NUTS in estimating volume fractions and $\lambda_{\|}$, even though they all start from the same initialization. Inspection of the probability distributions reveals that NUTS gives distributions biased away from the ground truth for these parameters. Furthermore, we note that NNEHMC generally outperforms L2HMC regarding the accuracy and variance of the estimates. Unlike in the toy example, where we knew the precise mean and variance of the distribution, we can only compute an approximate autocorrelation analysis in this case. We obtained an effective sample size of 1.5e−3 for NNEHMC and 1.9e−3 for L2HMC. However, the mean computation times for a single signal are $280s$ for NNEHMC and $443s$ for L2HMC.

Furthermore we emphasize that using L2HMC on this model is numerically unstable. By changing the random seed in our implementation, 18 out of the 100 trials with L2HMC either decline to and remain at zero acceptance probability for all chains by the end of training or

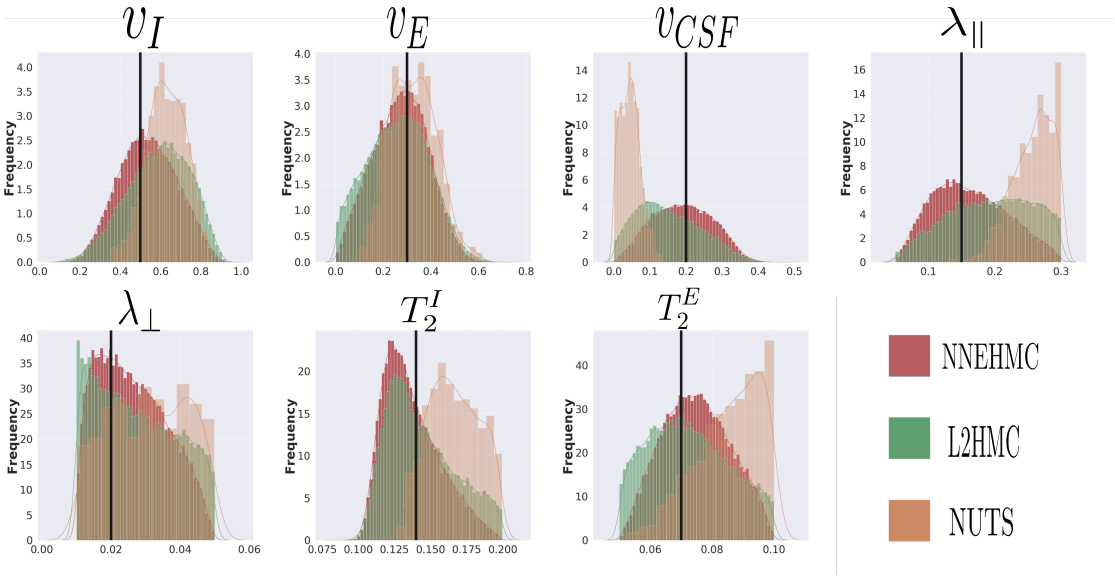


Figure 5.5: Representative plots of the marginal probability distributions for each parameter, where the black vertical line denotes the ground truth value. We can see that NNEHMC provides informative posterior distributions from which inference seems justified, while NUTS provides quasi-uniform distributions and distributions biased towards the parameter bounds.

encounter numerical errors (NaN, infinities). This can happen, for instance, if the proposal samples move too far away. NNEHMC was robust to such changes. We do not consider these results in the analysis since they are invalid for parameter estimation and would artificially bias the results for L2HMC negatively. In order for NUTS not to develop similar numerical issues, we had to set a desired acceptance probability of 99%. It is probable that the poor results of NUTS stem, in part, from inefficient sampling due to a highly curved parameter space which the adaptive tuning could not overcome. Thus, we can see that the parametrization with a neural network can enable efficient sampling of problematic regions in parameter space; however, regularization with an acceptance probability term is needed for stability.

5.5 Discussion and Conclusion

The motivation to use MCMC for inverse problems stems from the desire to explicitly consider the probablistic nature of inverse problems. The motivation to use machine learning in MCMC was to learn from the data, in a self-supervised way, the best way to propose new samples; this is made possible since one criterion for judging the samples is the autocorrelation of the samples, which can be optimized easily using a tractable loss function. As shown, MCMC and self-supervised machine learning can be combined in a natural way, as NNEHMC and L2HMC are natural extensions of HMCMC samplers; in NUTS, for example, the innovation was largely the automatic way in which the parameters of the leapfrog algorithm for running the Hamiltonian dynamics are tuned during the burn-in period. This is analogous to the

self-supervised way in which the proposal networks are trained; in both cases, parameters which determine the proposed samples are tuned/optimized. That the training period is clearly analogous to the burn-in period further illustrates how seamlessly self-supervised methods can be introduced to MCMC, a traditional approach. Depending on the speed of the MCMC sampler, the addition of the neural network component could come at little to no addition in computational time since all methods use burn-in periods.

However, this seamless combination also inherits the main drawback of MCMC methods: ths computation method. While our method is parallelizable since it fits the signal from one voxel, the computation time of 280s per signal is prohibitive, considering that a single brain image could require the solution for millions of voxels. However, this computation time could potentially be reduced with tuning of the number of burn-in samples or inference samples. Furthermore, the current implementation of our methods and L2HMC were written in Tensorflow 1.0; it is possible that the additional optimizations in subsequent versions/switching to Pytorch could provide a substantial speedup, particularly as our method currently makes extensive use of while/for loops due to the iterative nature of running the Hamiltonian dynamics.

While our proposed method of embedding a self supervised network with a weak constraint to Hamiltonian dynamics into a MCMC scheme can be applied broadly to many inverse problems, we validated on a synthetic case of a joint diffusometry/relaxometry inverse problem, where we simulated MR measurements in a white matter voxel of the brain. While our method was dramatically better in terms of robustness and accuracy than competing methods, we note that it still exhibited significant relative errors with respect to the ground truth. In some sense, this was expected as the model was extremely nonlinear (sum of products of error functions and exponential functions) and degenerate due to the multi-compartment nature (i.e. multiple different mixtures of compartments could explain the same signal). Furthermore, we simulated a realistic SNR. However, in line with the theme of this thesis, subsequent work [122] has shown that the solution of this inverse problem can be dramatically simplified with realistic modelling of the constituent compartments in white matter, in combination with using targeted values for the experimental parameters TE and b. Realistic modelling shows that as $b \to \infty$, the signal contributions of two out of three of the compartments in our model go to zero. Then, one can simply fit the T_2 and diffusion coefficient of the remaining compartment by varying TE and b, with the constraint that b is "large enough". One can then plug in these values into the full problem, thereby reducing the complexity/degeneracy since some previously estimated parameters are now fixed. For future work, it would be interesting to combine this work with our method to solve the reduced problem.

Furthermore, our application only studied a specific joint diffusometry/relaxometry inverse problem, i.e. the marriage of a spherically averaged diffusion model and an exponential decay model for the T_2; using different measurement models/assumptions, there are many inverse problems in this area [123]–[126], with correspondingly different solution methods.

In this chapter, we have proposed and tested a parametrization of Hamiltonian MCMC with a neural network (NNEHMC) which jointly optimizes sample acceptance probability and distances between successive samples in order to efficiently and stably sample probability distributions, particularly in regions of parameter space with high curvature; in particular, our method is an extension of an existing work (L2HMC), where we applied a weak constraint to Hamiltonian dynamics through an addition to the loss function. High curvature regions frequently occur in the probabilistic estimation of parameters in bio-physical models since the posterior distributions are parametrized, in part, by highly nonlinear models. We show on a recently proposed MRI model that the neural network enhancement provides parameter estimates which are more accurate and precise than those given by a least squares fitting and the state of the art NUTS and L2HMC samplers; in addition NNEHMC provides more numerically stable sampling than NUTS or L2HMC. Furthermore, we show that the neural network parametrization provides qualitatively different and more informative posterior distributions than those produced from NUTS; NNEHMC can produce posterior distributions which are Gaussian-like centered near the correct parameter values. This highlights the potential of augmenting MCMC methods with neural networks to improve probabilistic solutions of inverse problems.

6 Model Informed Machine Learning

The content of the following chapter is based on the postprint version of the article: "Model-informed machine learning for multi-component T_2 relaxometry" published in Medical Image Analysis [127]. DOI: 10.1016/j.media.2020.101940.

6.1 Introduction

In this chapter, we transition from self-supervised methods to supervised methods for solving inverse problems; in particular, in this chapter we show how the combination of realistic models and priors allows for the generation of large synthetic datasets so that supervised methods can be used in a problem where there is usually no ground truth data available.

As in Chapter 5, we consider an inverse problem related to T_2 relaxometry; however, instead of focusing on recovering a single T_2 for each compartment, we consider a different viewpoint by viewing the T_2 in each voxel as a spectrum or distribution. Furthermore, we consider a more realistic model for the signal that takes into account practical difficulties in data acquisition as well as realistic priors for the structure of the solution.

In the previous discussions, we presented the T_2 as single number per voxel or per compartment; however, tissue heterogeneity and partial volume effects in a voxel can render it more appropriate to consider distributions of T_2s per voxel rather than a single T_2 values [128], as the mixture of spins can be more continuous than discrete. We distinguish single-component T_2 relaxometry, where each voxel is characterized with a single T_2 or the T_2s of a small number of compartments, from multicomponent T_2 relaxometry, where each voxel is characterized with a T_2 distribution. In general, T_2 distributions are reconstructed from multi-echo T_2 MRI signals, which can be acquired, for example, through multi-echo spin echo sequences, where a 90° excitation pulse is followed by a train of 180° refocusing pulses. Given a sequence of n pulses, the signal $\mathbf{s}$ is a vector of n measurements at the corresponding echo times (TE_i).

Let $p(T_2)$ and α denote the distribution of T_2s in a voxel and the effective flip angle of the refocusing pulses, respectively. If $\alpha = 180°$ and the voxel is assumed to have a single T_2, then the decay of the signal is exponential, as is implied by the Bloch equations [129]. In practice, inhomogeneities in the transmit field (B_1+) result in an effective refocusing pulse that can vary significantly from 180° and can be spatially heterogeneous [130]. This leads the resulting signal to deviate from the ideal exponential behavior, which can be modelled using the extended phase graph (EPG) formalism [131]. The EPG formalism considers as parameters α, TE, T_1 and a single T_2. The code used for the EPG simulations in this chapter is based on work in [132]. We use the common simplification of fixing $T_1 = 1000ms$, as the T1 relaxation time cannot be estimated using the acquisition sequences we examine in this chapter [133]; hence, it is commonly fixed to its mean value in brain tissue. Then the normalized signal follows

$$\mathbf{s}(TE_i) = \int EPG(TE_i, T_1, T_2, \alpha)\, p(T_2) \mathrm{d}T_2. \tag{6.1}$$

Identifying $\mathbf{y}$ as series of signal measurements at different T_E and $\mathbf{x} = p(T_2)$, we can see that this model defines an inverse problem where $M(\mathbf{x})$ is the expectation of the EPG signal, weighted by $\mathbf{x}$.

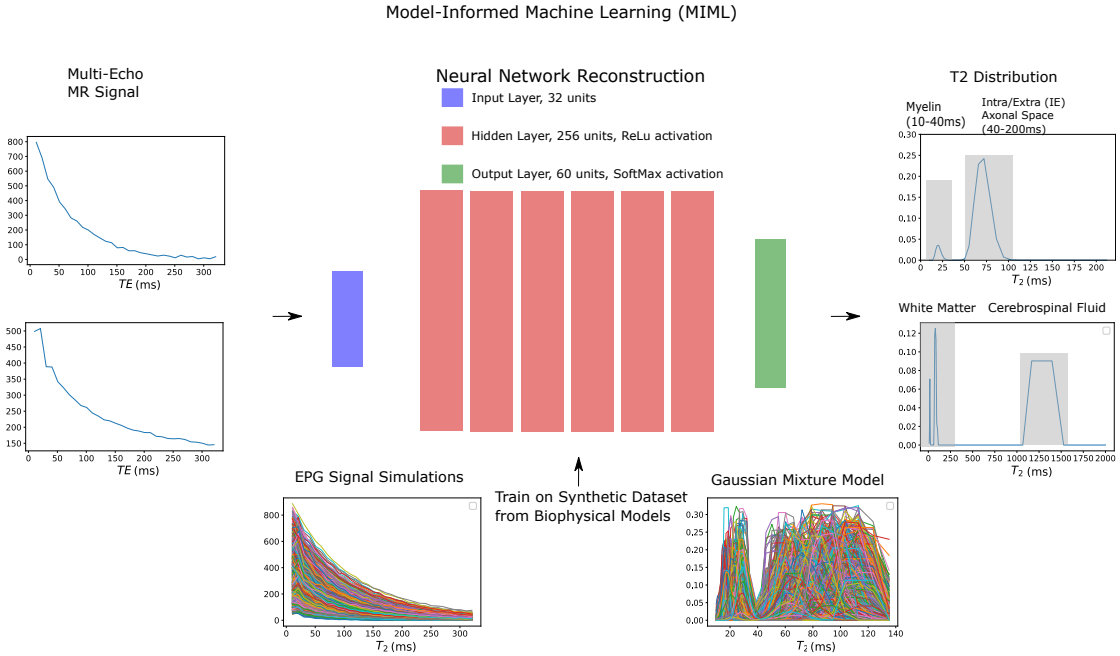
One key application of multi-component T_2 relaxometry is in neuroimaging, where the different parts of the T_2 distribution are assumed to arise from the different anatomical compartments in brain tissue, particularly in white matter, as is considered in Chapter 5. This can be used, for instance, to generate a map of the myelin water fraction (MWF) [134] such that areas of demyelination corresponding to the effects of neurodegenerative disorders can be identified [135]. In particular, it is commonly assumed/modelled that the T_2 distribution in white matter contains multiple lobes having well-separated peaks, and that the eventual overlap between the T_2 lobes of myelin and the intra/extra axonal space water pools is minimal [128], [136]–[139]. However, we note that there is generally no in-vivo ground truth data, making supervised methods infeasible.

6.1.1 Related Work

In order to estimate $p(T_2)$ from Equation 6.1 two main approaches are generally used: parametric and non-parametric approaches. Parametric approaches rely on *a priori* information on the T_2 distribution in brain tissue, particularly white matter, in order to fit the parameters of biophysical models to the MRI signal [140]–[145]. In these approaches, the MRI signal is modelled as a linear combination of signals from a fixed number of water pools (around 2-3) such as myelin water, the water in the intra-/extra-axonal space, and cerebrospinal fluid:

$$p(T_2) = \sum_{i=1}^{n} v_i F_i(\mathbf{m}_i, T_2) \tag{6.2}$$

Here n is the number of water pools assumed, and $F_i, \mathbf{m_i}, v_i$ are the probability distribution,



18

Figure 6.1: An overview of our method (MIML) for multicomponent T_2 relaxometry where we learn a mapping from the multi-echo MR signal to the corresponding T_2 distribution. On the left are example MR signals, on the right are the corresponding T_2 distributions: the first distribution is in white matter (WM), where there are assumed to be two lobes: one at a T_2 of around 10-40ms corresponding to myelin water and one at a T_2 of around 50-120ms corresponding to the intra and extra axonal spaces. The second distribution includes WM and cerebrospinal fluid (CSF), whose T_2 is commonly assumed to be around 1-2s. Our method consists of training a neural network on a synthetic dataset derived from biophysical models to learn the mapping from signal to distribution. At the bottom, we show a small subset of 1000 simulated signals and corresponding T_2 distributions from our synthetic training dataset.

parameters of the probability distribution, and volume fraction of the ith water pool. A wide variety of parametric distributions (Delta, Gaussian, Truncated Gaussian, Wald, Gamma, Log-Gaussian, Laplacian) are used to model the T_2 distributions in these pools; however, [140] shows that using these different distributions have negligible differences on the corresponding signal when using the same means and variances; they conclude that due to the ill-posedness of the inverse problem, extracting more than general lobular shapes (characterized by the mean and variance) is extremely difficult if not impossible, even at extremely high signal to noise ratios (SNR). We note that the compartment, single-component model used in Chapter 5 can be integrated in this framework by using a Dirac delta function as the distributions. The parameters estimated are the water volume fractions and the parameters of the distributions which are done through optimization [143], [145] or Monte Carlo methods [142], [146]. To stabilize the fitting and to use prior information on the compartments, constraints are enforced on the parameters. **For instance, the mean T_2 of myelin water is typically bounded between 10 and 40ms, and the mean T_2 of CSF is typically assumed to be greater than 1s.** Some works, such as [143], go even further and fix the mean or standard deviations of the probability distributions of some compartments to predetermined values. While parametric estimations are generally stable and histologically validated, they are usually computationally expensive and restricted by the biophysical model used; the number of compartments needs to be fixed for each voxel before fitting. Further, we note that the *a priori* information used in the parametric approaches i.e. the assumption of lobular structure, bounds on the parameters of the distribution, etc. comes from historical evidence, where studies used **non-parametric** methods to estimate the T_2 distributions and assigned lobes in their reconstructions to different water pools [147].

In contrast, non-parametric approaches do not make *a priori* assumptions on the data, such as the number of compartments. This is relevant for studying abnormal brain tissue, where compartments not considered in standard biophysical models might be present [135]. In addition, they generally require orders of magnitude less computation time than parametric methods. Non-parametric methods discretize equation (1) as a product of a dictionary matrix and a discretized T_2 distribution and solve directly for the discretized T_2 distribution [130], [137] using non-negative least squares (NNLS) algorithms [148]. The T_2 distribution, $p(T_2)$, is recovered by solving an inverse problem [130], [137]. First, given discretized ranges of flip angle (α) values and T_2 values, a dictionary D_α of T_2 decay signals is constructed for each α value through the EPG formalism. D_α **is a matrix where the columns are the simulated MRI signals (obtained from the experimental TEs) over a range of T_2 values.** Given a flip angle α, the corresponding dictionary D_α, and the MRI signal $\mathbf{s}$, the following optimization problem is solved

$$\underset{\mathbf{p}\geq 0}{\arg\min} \|D_\alpha \mathbf{p} - \mathbf{s}\|_2^2 + \lambda \Phi(\mathbf{p}) \tag{6.3}$$

where Φ is a regularization function with parameter λ, and $\mathbf{p}$ is the discretized, un-normalized T_2 distribution to be estimated. The flip angle corresponding to $\mathbf{s}$ is chosen by solving the

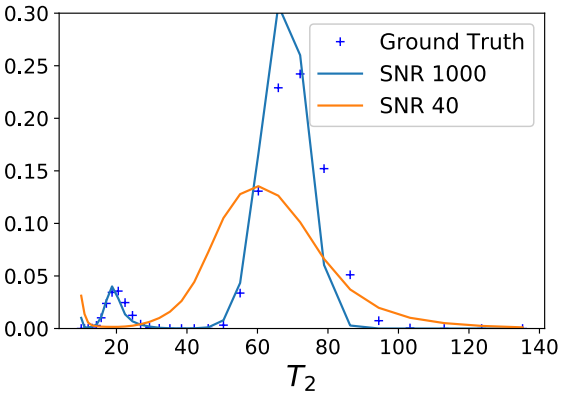


Figure 6.2: Here we show reconstructions from two different SNRs (40,1000) using NNLS with Laplacian regularization, with the corresponding ground truth for comparison. We can see that at the lower SNR, the reconstructed distribution is severely oversmoothed.

above problem (with $\lambda = 0$) for multiple values of α and taking the value which corresponds to the least fitting error [130]. **Two standard choices for $\Phi(\mathbf{p})$ [149] are**

- $\Phi(\mathbf{p}) = \|\mathbf{p}\|_2^2$**, which we refer to as Tikhonov regularization.**
- $\Phi(\mathbf{p}) = \|\mathbf{Lp}\|_2^2$**, where L is a finite difference approximation of the Laplacian operator. We refer to this as Laplacian regularization.**

These choices are used in order to promote increased conditioning of the problem and the smoothness of the resulting distribution [150]. Without regularization, solutions to Eq. (3) are vulnerable to noise and usually produce inaccurate solutions that overfit the signal with e.g. false positive peaks, etc. A common heuristic for selecting λ is to accept λ such that the signal fitting error is approximately 1.02-1.025 times greater than the error from NNLS with no regularization [151]. However, it is known that regularization can introduce undesirable bias to the reconstructed signals, e.g. over-smoothing. In particular, regularization can contradict the expectation of disparate lobes in the distribution corresponding to disparate tissues in the same voxel (e.g. myelin and intra/extra axonal space water), particularly at lower SNRs. For example, at low SNRs, the myelin water lobe can become completely over-smoothed, for an example see Fig. 6.2.

Once the T_2 distribution is recovered, generating parameters of interest such as volume fractions of the water pools in the voxel require either a distribution where distinct lobes can be assigned to distinct compartments (such as in the right side of Fig. 6.1) or *a priori* information. After examining distributions reconstructed from experimental scans, the different lobes of the distributions (if distinct lobes are present) are assigned to different water pools based on theoretical and experimental grounds [135]. From the mean and standard deviation of these lobes, bounds are derived for the T_2 values for each water pool. Then water volume fractions for each pool are calculated by integrating the probability distribution between the bounds of

the T_2 for each pool. For instance, at 3T the myelin water fraction (MWF) is usually computed as

$$MWF = \frac{\int_{T_2=10ms}^{T_2=40ms} \mathbf{p}(T_2)\mathrm{d}T_2}{\int_{T_2=10ms}^{T_2=2000ms} \mathbf{p}(T_2)\mathrm{d}T_2}, \tag{6.4}$$

where the bounds 10-40ms were obtained from the myelin water lobe in NNLS reconstructions in past papers [147].

We note previous studies found that both parametric and non-parametric methods require a high signal-to-noise ratio (SNR) to detect different components in the T_2 distribution [152]–[154]. For a clinically achievable SNR=100, more than 5% of the voxels were incorrectly estimated to have no myelin water component, and the percentage raised to 12% for SNR=50 [155]. Similar results were reported in [156], where the myelin water component was not found in human brain regions located in myelinated areas of the frontal and lateral projections fibers. In addition, [154] found that in synthetic studies, NNLS with Tikhonov Regularization tends to underestimate the true MWF value in the range of 0.3 to 4 percent at SNR 1000, with the problem worsening at lower SNRs; for reference, the MWF is assumed to be in the range of 0-30 percent in normal appearing white matter.

Recently, [157], [158] have both proposed to augment non-parametric approaches with machine learning in order to speed up the computation time. As training data, they acquired brain scans in several subjects *in vivo* using a 3D multiple echo gradient and spin echo sequence with 32 echoes [159]. They then ran regularized NNLS reconstructions on the data and obtained the probability distributions and MWF for each voxel. [158] trained a multi-layer perceptron (MLP) to take as input the raw data, and output the MWF, using the *in vivo* NNLS reconstructions as ground truth. [157] trained MLPs to reconstruct the MWF as well as the probability distributions from the raw echo data, using the *in vivo* NNLS reconstructions as ground truth. These approaches have the advantage of reconstructing regularized NNLS solutions for the whole brain in under a minute, a fraction of the time required using the standard NNLS algorithm. However, as their ground truth is the regularized NNLS solution, their method inherits all the problems of NNLS. Further, by training on data acquired from specific MRI machines using a specific sequence, there is the problem of generalizing to different machines and different sequences. Both would require new acquisitions as well as additional training time.

In summary, parametric methods implicitly regularize and stabilize the problem by using biophysical models and prior knowledge to constrain the space of T_2 distributions. However, the resulting optimization problems to be solved are significantly more costly than those of non-parametric methods, with an additional loss of flexibility due to imposition of the number of compartments and other details of the model. Non-parametric solutions are fast, but also ill-posed and highly susceptible to noise; hence, regularization is necessary, with the concomitant drawbacks of over-smoothing and sparsity of the reconstructed distributions,

particularly at clinically achievable SNRs for sequences with high spatial resolution. Further, the extraction of parameters of interest such as the MWF is theoretically based on assuming a lobular structure of the reconstructed distribution, which is often not the case in middling to high levels of noise.

6.1.2 Contributions

In this chapter, we propose a new method for multi-component T_2 relaxometry in brain tissue. In Fig. 6.1, we show the overview of our proposed method as well as a prototypical T_2 distribution in white matter, composed of the myelin water lobe and the lobe corresponding to the water in the intra/extra axonal space; in addition, we show the corresponding MRI signal. We propose to combine machine learning and aspects of parametric and non-parametric approaches to the reconstruction of T_2 distributions from multi-echo T_2 data. We do this by creating a synthetic dataset derived from biophysical models and training a multi-layer perceptron (MLP) [38] on this dataset to take as input the MRI signal and directly output the associated T_2 distribution. In this way, we fully use all the available model-driven information (the realistic EPG model and tractable prior information on the structure of solutions) while also leveraging data-driven methods (MLP). We call our method Model-Informed Machine Learning (MIML). Our main contributions are as follows:

- Construction of an extensive synthetic dataset that we construct purely from simulations guided by biophysical models, which we use for training the MLP.
- Introduction of a robust loss function for the network to recover the T_2 distribution consisting of a combination of the mean squared error and the Wasserstein-1 Distance [160]. We show that training with the Wasserstein distance significantly increases the accuracy of MWF estimates on a realistic, synthetic case, compared to training with solely a mean squared error (MSE) loss function.
- Rigorous and extensive evaluation of our method and previous work in non-parametric and parametric approaches, on synthetic and real datasets (*ex vivo*, *in vivo*, healthy, pathological). We show that our method outperforms other methods in terms of accuracy, plausibility, and robustness of the reconstructed distributions and MWF maps as well as lesion visualization.

6.2 Methods

Our method for reconstructing T_2 distributions from MRI data is based on a MLP which is trained to learn a map directly from MRI signals with a 32 echo acquisition scheme to the corresponding T_2 distribution, as is the result in non-parametric methods. To reduce the inherent ill-posedness of this problem, the training is conducted on a synthetic dataset of pairs of MRI signals and T_2 distributions which we constructed using EPG simulations and is

Range of Mean and Standard Deviation for Simulated Water Pools

Water Pool	Range of Mean T_2 (μ)	Range of Std. of T_2 (σ)
Myelin	15-30ms	0.1-5ms
Intra/Extra Axonal Space (IES)	50-120ms	0.1-12ms
Gray Matter (GM)	60-300ms	0.1-12ms
Pathology	300-1000ms	0.1-5ms
CSF	1000-2000ms	0.1-5ms

Table 6.1: The ranges for the possible mean (μ) and the standard deviations (σ) used for the Gaussian, T_2 distributions of the different water pools used in our dataset.

informed by biophysical models and realistic values for the parameters of interest, such as the range of T_2s for different water pools, taken from the literature. This implicitly constrains the space of possible T_2 distributions (as in parametric approaches). We show an overview of our method in Fig. 6.1.

6.2.1 Synthetic Dataset Generation

To generate the synthetic T_2 distributions, we start from standard biophysical models for the brain [137]. Brain tissue can be roughly subdivided into white matter, grey matter, cerebrospinal fluid, pathological tissue, and combinations of these tissues. The water of these tissues are made up of a combination of different pools of water. We model the T_2 distributions of brain tissue as a mixture of Gaussians, where each Gaussian component corresponds to a different water pool (e.g. myelin water, intra/extra axonal space water).

$$p(T_2) = \sum_i \frac{v_i}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(T_2-\mu_i)^2}{2\sigma_i^2}\right) \tag{6.5}$$

$$v_i \in [0,1], \sum_i v_i = 1 \tag{6.6}$$

Here v_i is the volume fraction of the ith water pool, and μ_i, σ_i are the mean and standard deviation of the T_2 distribution of the ith water pool. We justify our choice of modelling using Gaussians by noting that [140] found that modelling the T_2 distributions using a variety of different distributions including the Gaussian distribution had negligible differences in parametric methods; essentially, they found that the ill-posedness of the reconstruction made it extremely difficult to distinguish between different distributions when the mean and standard deviation were fixed. In Table 6.1, we show the water pools we consider as well as the range of the means and standard deviations for each water pool.

These water pools were chosen based on the commonly used biophysical models for the water pools in brain tissue. In parametric models, white matter is modelled as a combination of the myelin and intra/extra axonal water pools [140]–[144]. Further, they consider a water pool

which accounts for cerebrospinal fluid (CSF). The water in gray matter can be modelled as similar to the IES water pool, with an extended mean T_2. However, brain pathologies can result in T_2 distributions different from those of white matter, gray matter, and CSF; for example, [161] found that MS lesions can contain a water pool in the range between that of the IES pool and the CSF pool. We set the mean values in line with those reported in the literature [136], [147], [161], [162]. We included an extensive range for the standard deviations, ensuring that our dataset has both sparse, intermediate, and wide T_2 distributions in order not to bias our dataset towards any extreme. However, there can be partial volume effects, where different configurations of brain tissues are contained in a single voxel; for example, water from white matter and CSF could be present in a single voxel. Therefore, in our dataset, we divide T_2 distributions in the brain into seven cases, each with a characteristic mixture of water pools.

- **White matter (WM)**: 2 Water Pools (Myelin and IES).
- **Cerebrospinal fluid (CSF)**: 1 Water Pool (CSF)
- **Gray matter** : 2 Water Pools (Myelin and GM)
- **Mixture of WM and CSF**: 3 Water Pools (Myelin,IES, and CSF)
- **Mixture of WM and GM**: 3 Water Pools (Myelin,IES, and GM)
- **Mixture of CSF and GM**: 3 Water Pools (GM and CSF)
- **Pathology**: 1 Water Pool (Pathology)

We note that as there is a small quantity of myelin in gray matter, the gray matter pool is composed primarily of the GM component in Table 6.1 as well as the myelin water component which is constrained to have a random v_i between 0 and 5 percent. Concretely, suppose we want to generate a random T_2 distribution for the case of a mixture of WM and CSF. This distribution is characterized by the combination of three water pools (myelin, IES, and CSF).

$$p(T_2) = \sum_{i=1}^{3} \frac{v_i}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(T_2-\mu_i)^2}{2\sigma_i^2}\right) \tag{6.7}$$

Therefore, by randomly selecting v_i from a Dirichlet distribution and uniformly sampling (μ_i, σ_i) for the three pools within the bounds in Table 6.1, we can generate a random T_2 distribution. Given the T_2 distribution, we use the EPG formalism to simulate the corresponding signal from an acquisition based on acquiring 32 echos with around 10ms spacing between each echo. In the real data we use for our evaluation, three slightly different echo times are used; the *in vivo* scans of healthy subjects use an echo train of 10.68ms, 21.36ms, ... 341.76ms, the *in vivo* scan of the subject with pathology uses an echo train of 10.36 ms, 20.72 ms, ..., 331.52 ms, and the *ex vivo* scan uses an echo train of 10ms, 20ms, ... 320ms. In the following, we describe our procedure with a single, fixed echo train: for the evaluation, we generated

three training datasets, one for each echo train. We note that alternative sequences with different numbers of echoes/different spacings can be accommodated by generating a new dataset. We first construct a family of dictionaries of EPG signals, defined as in the previous section, (D_α), where we vary α from 90 to 180. We use a high resolution T_2 grid (1ms to 2000ms with a spacing of 0.1ms) for the dictionaries. We generate 200,000 T_2 distribution variations per case by sampling (v_i) and (μ_i, σ_i) randomly from flat Dirichlet and uniform distributions, for a total of 1.4 million distributions. We randomly vary the flip angle (α) of the acquisition for each signal between 90 and 180 ° so that our method learns to account for different flip angles automatically, rather than having to first estimate the flip angle as in non-parametric methods. Given $v_i, (\mu_i, \sigma_i)$, we numerically approximate the corresponding T_2 distribution on the same T_2 grid as used for the dictionaries (D_α). Let $\mathbf{p_{HR}}$ denote the discretized distribution. Given the pre-constructed dictionary of EPG signals D_α corresponding to the randomly chosen α, the EPG signal corresponding to this distribution, s_{EPG}, is

$$s_{EPG} = D_\alpha \mathbf{p_{HR}} \tag{6.8}$$

As noted in the previous section, non-parametric approaches commonly use a much coarser, logarithmically spaced grid of T_2s for the discretization of the distribution. This allows to significantly reduce the computation time. Therefore, to directly compare our approach with non-parametric approaches, we downsample the ground truth distributions from the high resolution T_2 grid to a grid of 60 T_2's logarithmically spaced from 10ms to 2000ms, as is commonly used [159]. Denoting this downsampled, ground truth distribution as $\mathbf{p_{DS}}$, the dataset consists of the pairs $(s_{EPG}, \mathbf{p_{DS}})$. We note that our method does not depend on this downsampling; we use it for a fair comparison with non-parametric approaches. As outlined in the related work, the SNR of the signals is a crucial aspect of the reconstruction and hence the dataset generation. We define SNR with respect to the first echo of the signal sequence. From previous studies [136], [154], it is known that NNLS methods, perform well in the high-SNR regime (on the order of 1000). However, clinical scans with high spatial resolutions will rarely meet this SNR requirement; in the real scans of healthy subjects we use in our evaluation, we estimate a mean SNR on the order of 100. In order to make our method robust to the realistically low SNR regime, in training we randomly vary the SNRs of the signals between 80 and 200 in order to cover the potential SNR range of the voxels. We use a Rician noise model to add noise to the signals. In our evaluation, we show that training on this SNR range results in robustness to a wide range of SNRs (40-1000) on synthetic data. The data generation for 1.4 million signal/distribution pairs took less than one hour on a cluster using parallelization on 46 threads.

Using the synthetic datasets described, we train a MLP to map the MRI signal to the corresponding T_2 distribution.

6.2.2 Mapping the MR Signal to the T_2 Distribution

Architecture

Our network is composed of 6 hidden layers with 256 neurons per layer and an output layer with 60 units, corresponding to the size of the discretization of the distributions we use. The hidden layers use a ReLu function as the activation function, while the output layer uses a SoftMax activation function since the output should be the T_2 distribution. The input to the network is a vector with 32 elements corresponding to the 32 echos of the standard acquisition sequence. We note that we normalize the input by the magnitude of the first echo before feeding it to the network. To select the structure of the network, we trained 12 networks where we varied the number of hidden layers (3-6) and the number of neurons per layer (64,128,256,512). We selected 6 hidden layers and 256 neurons as this configuration had the lowest validation loss at the end of training; however, we note that the validation loss was not significantly different between the configurations.

Loss Function

Let $(\mathbf{x}, \mathbf{p_x})$ denote the normalized MRI signal and the corresponding T_2 distribution. Let $\Phi(\cdot, \theta)$ denote the multi-layer perceptron function with parameters θ, with $\Phi(\mathbf{x}, \theta)$ the predicted distribution. Given a batch of training samples $(\mathbf{x_i})$ of size n, the cost function we use to train Φ is

$$L(\theta) = \frac{1}{n}\sum_{i}^{n} \lambda \|\mathbf{p}_{\mathbf{x_i}} - \Phi(\mathbf{x_i}, \theta)\|_2^2 + W_1(\mathbf{p}_{\mathbf{x_i}}, \Phi(\mathbf{x_i}, \theta)) \tag{6.9}$$

where the first term corresponds to the squared L_2 norm (MSE loss) and the second term corresponds to the Wasserstein-1 distance on probability distributions [160]. We set λ to give approximately equal numerical weight to both terms in the loss function. Let u, v denote 1-D probability distributions with cumulative distribution functions U, V. Then the Wasserstein-1 Distance is equivalent to the following formulation [163]

$$W_1(u, v) = \int_{-\infty}^{\infty} |U(p) - V(p)| \mathrm{d}p \tag{6.10}$$

In this formulation, the Wasserstein distance can be efficiently computed on GPU using the cumulative sum function. The Wasserstein-1 distance is an appropriate metric to judge reconstruction quality in our application of T_2 distribution recovery as it correctly penalizes deviations from the ground truth distribution in relation to the location of the lobes in contrast to other losses such as MSE or Kullback-Liebler (KL) divergence. In particular, given two non-overlapping lobes, if the lobes are moved toward each other (but still do not overlap), the Wasserstein Distance will decrease significantly while the MSE and the KL Divergence will not change. An example is presented in Fig 6.3.

Using the Wasserstein distance helps us to avoid, for example, cases where the location

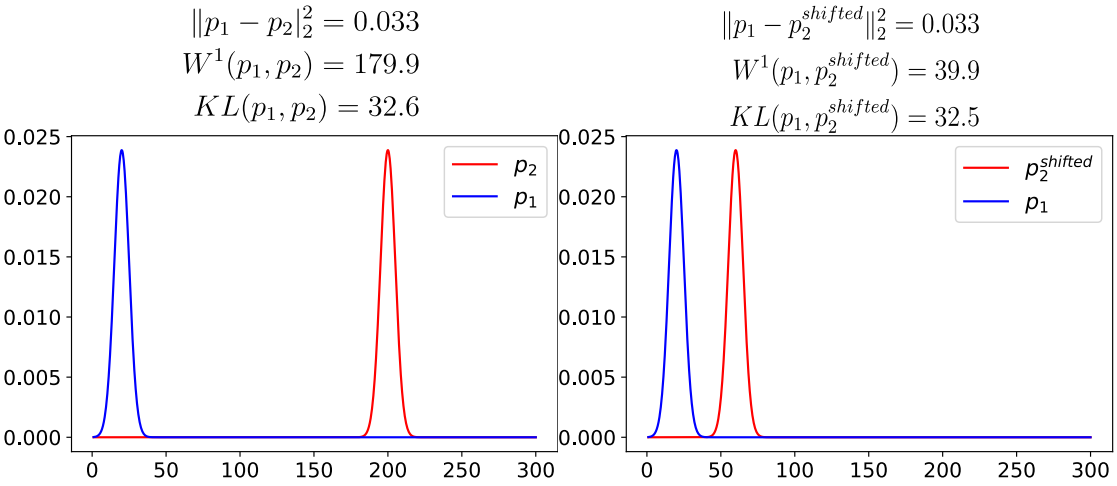


Figure 6.3: Here we show how the Wasserstein Distance, KL Divergence, and the MSE changes between two, non-intersecting lobes, as one is shifted closer to the other. We can see that while the MSE and the KL Divergence are almost the same for both cases, the Wasserstein distance decreases significantly when the means of the lobes are closer together. For judging the recovery of the T_2 distributions, it is then clear that the Wasserstein distance is an appropriate metric which coincides with our intuition as to judge how "close" two distributions are in our application

of lobes in the distribution could be arbitrarily placed with a similar loss if other metrics are used. We note that training with either MSE loss or Wasserstein-1 distance exclusively leads to suboptimal results, due to increased Wasserstein-1 distance in the first case and unstable reconstructions in the second case. We find that training with a combination of these results worked optimally; we further show in our evaluation that adding the Wasserstein-1 distance improves the accuracy of MWF estimation in realistic cases in comparison to training exclusively with MSE loss.

Implementation Details

We used TensorFlow 2.0 [164] on Python 3.6 [165] with an Nvidia GTX 2070 laptop GPU for constructing and training the network. For each case, we use 80 percent of the generated data for training, corresponding to a total of 1,120,000 signal/distribution pairs. We reserve 10 percent of the dataset as the validation set and the remaining 10 percent as the test set in our evaluation on synthetic data. We use the Adam optimizer [81] with a learning rate of 1e-3 and a batch size of 2000. We trained for 30 epochs, where we stopped the training based on the validation loss oscillating/no longer decreasing. We use the epoch with the lowest validation loss as the final model. This training took approximately 70 seconds to complete, showing the feasibility, given a large database of signals, to retrain models specific to given sequences, etc.

6.3 Evaluation

We perform reconstructions of the T_2 distributions from synthetic and real data using the following methods:

- Our proposed method, MIML,trained on signals with SNR 80-200 and the appropriate sequence of echoes.
- NNLS with Tikhonov regularization (NNLS-T)
- NNLS with Laplacian regularization (NNLS-L)
- Gaussian Mixture Fitting (GMF)

Both NNLS methods were implemented in-house in Python with full parallelization, and we use a standard method for selecting the regularization parameter [130], [151] by keeping the signal fitting error close to 1.025 times the signal fitting error obtained using NNLS without regularization. GMF is our implementation of a parametric approach similar to that of [140], where we fit a Gaussian mixture model with three compartments (Myelin water, IES water, CSF), extracting the volume fractions, the means/standard deviations of the T_2 of each compartment, and the overall normalization factor. We model as follows:

$$p(T_2) = \sum_{i=1}^{3} v_i \mathcal{N}(\mu_i, \sigma_i, T_2). \tag{6.11}$$

Then, the corresponding model signal is

$$\mathbf{s}^m(TE_i) = M_0 \int EPG(TE_i, T_1, T_2, \alpha) p(T_2) \mathrm{d}T_2. \tag{6.12}$$

where M_0 is the normalization constant. We calculate this integral numerically using a high resolution grid of T_2s as in the dataset construction. Given the experimental decay signal $\mathbf{s}$, the parameters $\mathbf{x} = ((v_i, \mu_i, \sigma_i), M_0)$ are calculated by solving the following optimization problem:

$$\mathbf{x} = \arg\min_{\mathbf{x}} \|\mathbf{s} - \mathbf{s^m}(\mathbf{x})\|^2 \tag{6.13}$$

where we constrain the μ_i, σ_i according to the bounds used for generating the dataset for MIML. Finally, v_i are constrained to the interval $(0,1)$, and are normalized before each calculation of the model signal during the optimization. As jointly estimating the flip angle adds significantly to the computation time and contributes to instability, we fix the flip angle in the Gaussian mixture fitting for each voxel to that calculated using a standard NNLS method [130]. We validated the accuracy of this flip angle estimation by comparing against B1 maps acquired on healthy subjects. We used the least squares optimization function in the Python library Scipy [166] to fit the signals to the Gaussian model.

6.3.1 Synthetic Data

Test Split of Synthetic Dataset

We show reconstructions on the test split of the synthetic dataset we generated using the acquisition sequence of 10.68ms, 21.36ms, ... 341.76ms. We show results over an SNR range from 40 to 1000 (40,80,150,200,400,1000). We compare the methods using the MSE and Wasserstein Distances of the reconstructed distributions with respect to the ground truth distributions.

Realistic Synthetic Case in WM

MWF mapping is a crucial application of T_2 relaxometry. In order to analyze the robustness and performance of our approach in a realistic case in WM, we show reconstructions on the following model of the distribution in a white matter voxel, with one lobe for myelin water and one lobe for IES water.

$$p(T_2) = v_m * InvGamma(\mu_m, \sigma_m) + v_{IE} * InvGamma(\mu_{IE}, \sigma_{IE}) \tag{6.14}$$

where we fix the values of the parameters to realistic values in line with those reported in the literature [135], [147]: v_m=0.15, v_{IE}=0.85, μ_m= 20ms, μ_{IE}= 70ms, σ_m=2.5ms, σ_{IE}=6ms. We use the inverse Gamma distribution to create the ground truth distribution to test the robustness of our method to changes in the assumed biophysical model. To study robustness to noise, we vary the SNR on the corresponding synthetic MRI signal from 40 to 1000, as in the test split. We generate 1000 realizations of noisy signals per SNR used. Further, we also show numerical results using our method **without** using the Wasserstein Distance in the loss function. We refer to this variant as MIML'. We compare the methods using the MSE, Wasserstein Distance, and estimated MWF of the reconstructed distributions with respect to the ground truth.

6.3.2 Real Data

As there is no ground truth for the T_2 distributions in real data, we evaluate the methods as in the literature by examining the MWF maps/comparing to anatomical scans or correlation to histology, the plausibility of the T_2 distributions, maps of the mean T_2 in the 50-200ms range, etc. We also report the mean SNR for each dataset, calculated in the same manner as in [154], where the first echo of the signals is divided by the standard deviations of the residuals from the NNLS-T reconstruction.

Ex Vivo Data

We show reconstructions from a Multi Echo Spin Echo (MESE) scan from the White Matter Microscopy Database [167] with 32 echoes (starting from 10ms with 10ms spacing), with a

TR of 3s and 8-fold averaging, of a single, cervical slice of a dog's spinal cord acquired *ex vivo* with an Agilent 7T animal scanner [168]. Five days before scanning, the spinal cord (perfused and post-fixed with paraformaldehyde 4) was extracted and washed in Phosphate-buffered saline (PBS) solution. After scanning, the spinal cord was osmified for two hours, embedded in EMbed 812 Resin, cut using a microtome, and polished. A scanning electron microscope (Low-angle backscattered electron mode) (JEOL 7600F) was used to image an entire slice of the spinal cord at a resolution of 0.26 micrometers per pixel. Using this histology image, we construct a histological map of the fraction of myelin in each voxel using a deep learning segmentation tool called Axon Deepseg [169]. We then register this histological map to the MRI space. **The resulting histological map is a map of the fraction of the voxel corresponding to the segmented myelin, not a map of the MWF.** However, assuming that the fraction of myelin in a voxel scales with the amount of myelin water, the two maps should be linearly correlated. We conduct a correlation analysis between the histological map and the MWF maps produced from the different methods. The estimated SNR on this slice is 784.

Healthy Subjects

We show reconstructions from high-resolution human brain scans acquired from 4 healthy controls using a 3T MRI scanner (MAGNETOM Prisma, Siemens Healthcare, Erlangen, Germany) located at CHUV Hospital (Lausanne, Switzerland), with a standard 64-channel head-/neck coil. The dataset was collected using a 3D multi-echo gradient and spin-echo (GRASE) sequence accelerated with CAIPIRINHA [170] with the following parameters: matrix-size=144x126; voxel-size = 1.6x1.6x1.6mm3; ΔTE/N-echoes/TR = 10.68ms/32/1s; prescribed FA =180°; number-of-slices = 84; CAIPIRINHA acceleration factor = 3x2; number of averages = 1; acquisition time=10:30min. Each subject was also scanned using an MPRAGE sequence for whole-brain T_1-weighted imaging [171]. To test the repeatability of the reconstructions, the healthy controls were scanned twice over two consecutive scanning sessions (scan-rescan scenario). We compare the MWF maps and the T_2 distributions produced from each method, show the coefficient of variability of the MWF in regions of interest (ROI) in WM, and conduct a study of the reproducibility of each method. The data for these subjects have an estimated mean SNR of 128.

MS Subject

We show reconstructions on a high-resolution human brain scan of a patient with relapsing-remitting multiple sclerosis, scanned using a 3T MRI scanner (MAGNETOM Prisma, Siemens Healthcare, Erlangen, Germany) located at the University Hospital of Basel (Basel, Switzerland) with a standard 32-channel head coil. In this case, MET2 data was collected using the previously described GRASE sequence for the healthy subjects, albeit with a starting echo time of 10.36ms and lower spatial resolution (voxel-size=1.8x1.8x1.8mm3) to accelerate the scan. In addition, a FLAIR [172] scan was acquired. A probabilistic lesion mask was generated by first using a convolutional neural network (CNN) trained to segment WM lesions [173] on

FLAIR images with subsequent manual correction by an expert. The FLAIR image/lesion mask were then registered to the multi-echo T_2 space. We use a threshold of 0.9 to denote a voxel as lesional. We analyze maps of the geometric mean T_2 in the range 50-200ms and MWF maps to study the MS lesions as in [174]. We also compare the correspondence of these maps to the lesion masks. In addition, we compare the T_2 distributions produced from each method in both normal-appearing tissue and the lesions. The estimated SNR of this scan is 112.

6.4 Results

6.4.1 Synthetic Data

Test Split of Synthetic Dataset

In order to visualize the average performance over the test split, in Fig. 6.4 we plot the mean distribution over all the ground truth distributions in the test split. In addition, we show the mean reconstructed distributions over the test split from the methods we compare. We also show plots zooming in on the different T_2 regions for better visualization. **MIML performs robustly and consistently across the whole SNR range, providing the best conformity to the ground truth distributions over the entire range of T_2s.** In contrast, NNLS with Tikhonov and Laplacian regularization both require SNR 1000 in order to generate a plausible distribution in the T_2 range $10-50$ms, with SNRs below this resulting in highly over-smoothed distributions. Further, even at high SNRs, both methods have over-smoothing in the T_2 range $50-2000$ms. For the Gaussian mixture fitting, we note that only the cases of WM and WM + CSF correspond to the model used, as it is necessary to fix the number of compartments beforehand. Therefore, the relevant T_2 ranges to examine are $10-120, 1000-2000ms$. **For GMF, high SNRs (200-1000) are required for plausible distributions with respect to the ground truth, with remaining distortions at low T_2 values.** In Fig. 6.5 we show boxplots of the MSE and the Wasserstein Distance between the ground truth distributions and the reconstructed distributions from the different methods over the SNR range. As the model used in GMF only applies to WM and WM+CSF, we show the results over the whole test set as well as over just the WM and WM+CSF cases in the test set.

For both MSE and Wasserstein Distance, MIML performs the best with the lowest median error and comparable or smaller interquartile ranges, across the whole SNR range. As expected, all methods improve with increasing SNR. **The limitations of the GMF model are clear, as it provides competitive results with the other methods only when restricted to the signals from the WM and WM+CSF cases, due to the need to fix the model/number of compartments beforehand. Overall, MIML, which is trained on signals with SNR 80 to SNR 200, generalizes well to the test set as well as to SNRs outside the range on which it was trained.** From the plots of the mean distributions and the boxplots of the error metrics, we can see that MIML performs better in distribution reconstruction than the other methods, parametric and non-parametric, across a wide range of SNRs. **In addition, the flexibility of using MIML in**

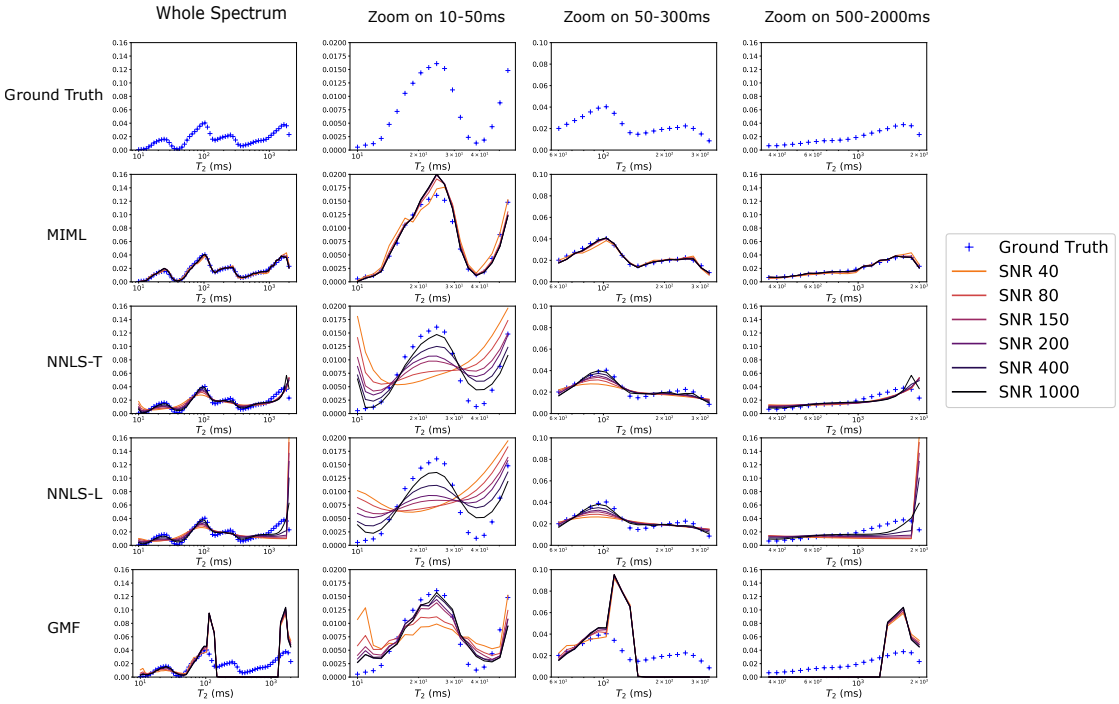


Figure 6.4: Plots of the mean distribution over all of the ground truth distributions in the test split of our synthetic dataset, as well as the mean reconstructed distribution over the test split from each method. We zoom in on the T_2 ranges 10-50ms, 50-400ms, 400-2000ms to show the average performance in the different cases (WM, CSF, etc.). Our method produces the most robust and accurate reconstructions with respect to changing SNR and the ground truth distributions respectively. All other methods require high SNRs (400-1000) for plausible distributions that, however, still retain distortions, particularly in the range of T_2s associated with myelin water (10-50ms). We note that the poor performance of GMF outside the T_2 range 10-120ms, and 1000-2000ms is due to model mismatch; GMF is valid only for the WM and WM+CSF cases. We use a logarithmic scale for the T_2 axis.

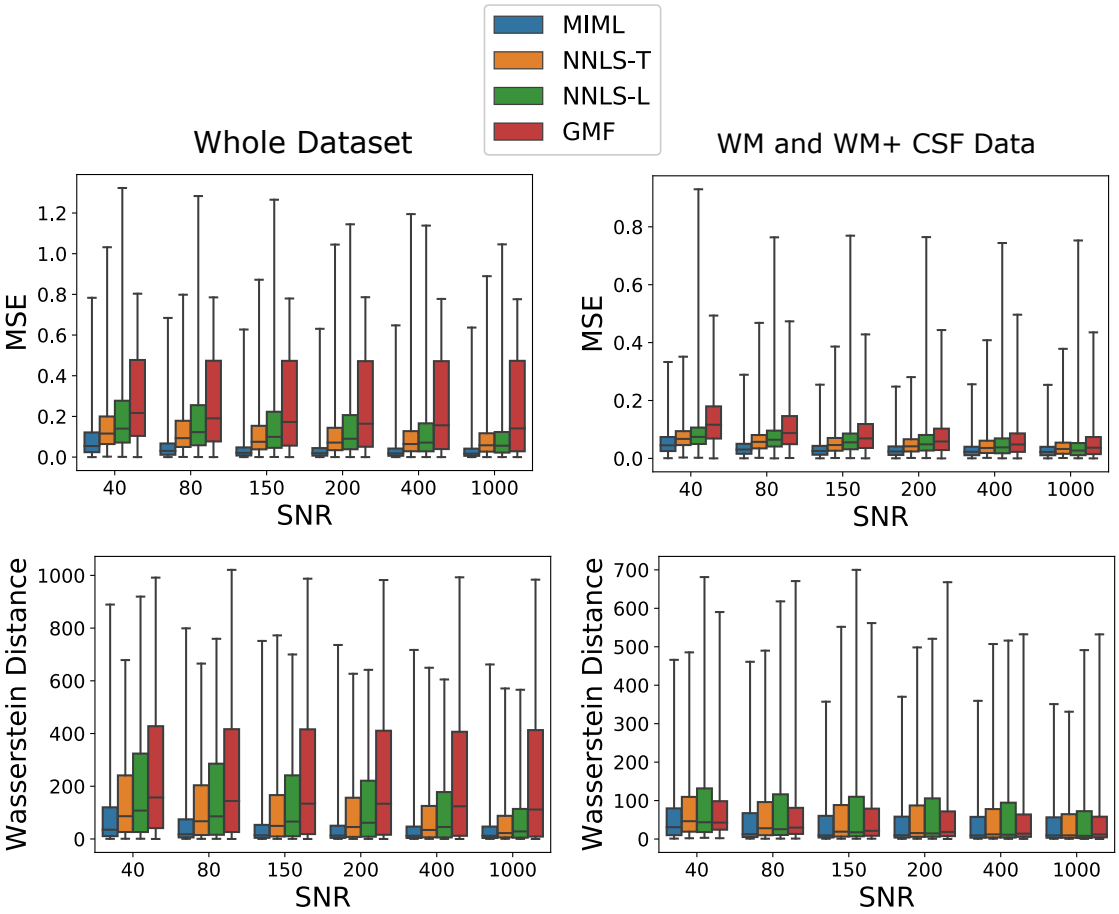


Figure 6.5: Boxplots of the MSE and Wasserstein Distance between the ground truth distributions in the test set of our synthetic dataset and the corresponding, reconstructed distributions from each method over a range of different SNRs. We show results over both the whole test set, as well as results restricted to the WM and WM+CSF cases, where the GMF model is valid. In general, MIML provides the most accurate and robust reconstructions, with the lowest median errors as well as lower or comparable interquartile ranges. As expected, the performance of GMF becomes comparable to other methods when we restrict to only the WM and WM+CSF cases, where the GMF model is valid.

comparison to GMF is clear, as MIML does not require fixing the number of compartments. However, the test set is generated according to the Gaussian mixture model; further, as we randomly generate the ground truth distributions, not all of the ground truth distributions are realistic, though we note that unrealistic distributions in the training can improve the generalizability of MIML.

Realistic Synthetic Case in WM

In Fig. 6.6, we plot the ground truth distribution and the mean reconstructed distributions from each method over the SNR range. In addition, we show boxplots of the MSE and Wasserstein Distance between the ground truth distribution and the reconstructed distributions from the different methods over the SNR range. Further, we show a boxplot of the error in MWF estimation. **MIML performs robustly and consistently, on average, across the whole SNR range.** However, the reconstructed distributions resolve a more spread out myelin water lobe than in the ground truth, even at SNR 1000; this could be due to training on significantly lower SNRs or the model mismatch. At SNRs below 400, NNLS-T and NNLS-L are unable to resolve a myelin water lobe due to over-smoothing as well as a displaced IES lobe; GMF resolves the myelin water lobe, but with a significantly displaced mean. At SNR 1000, NNLS-T and NNLS-L are able to resolve the myelin water lobe accurately, albeit still with a small distortion at $T_2 = 10$; GMF is able to accurately capture the myelin water lobe at SNR 1000, albeit with a displaced IE lobe.

With regard to MSE and Wasserstein Distance, MIML performs the best, with the lowest median error and comparable or smaller interquartile range across the whole SNR range. As expected, MIML' performs similarly to MIML with respect to MSE and significantly worse with respect to Wasserstein Distance, as it is only trained with the MSE loss. With regard to the estimated MWF (obtained by summing from T_2 bounds of 10-40ms), we see that MIML performs the best in the SNR range 80-400, with median errors closest to zero, and comparable or smaller interquartile ranges; we remind that the ground truth MWF was 0.15. At SNR 40, all methods either significantly over or underestimate the MWF, while at SNR 1000, MIML and NNLS-L provide comparable median errors. However, we note that NNLS-L has a significantly higher standard interquartile range than MIML at SNR 1000. Further, the results are consistent with results in [154] that the NNLS methods tend to underestimate the MWF. MIML' provides mediocre performance, generally underestimating the MWF value. Comparing the performance of MIML and MIML', we can see that using the Wasserstein Distance in the loss function during training significantly improves the performance of our method in terms of MWF estimation in a realistic case as well as the Wasserstein Distance of reconstructed distributions to the ground truth. Finally, in Fig. 6.7, we show the reconstructed distributions and the mean distribution for each method for SNRs of 200 and 1000. **Although the mean distribution from NNLS-T and NNLS-L corresponds well to the ground truth distribution at SNR 1000, the reconstructions of NNLS-T and NNLS-L are highly sensitive to added noise, with huge variability in the reconstructed distributions, particularly in the**

myelin lobe. In contrast, MIML, and to a lesser extent, GMF, are much more robust to the noise, showing little variability in the reconstructed distributions. Overall, MIML performs accurately and robustly across the SNR range with respect to the MSE, Wasserstein Distance, and the MWF value, showing the robustness to changing the assumed Gaussian model for the distribution as well as the applicability in a realistic case. Other methods perform comparably, on average, at high SNR values (SNR 1000), as expected.

From the results on the synthetic data, we conclude that MIML, even trained on a limited range of SNRs, is able to robustly and accurately reconstruct T_2 distributions over a wide range of SNR values. Overall, MIML outperforms all other methods in terms of MSE and Wasserstein Distance with respect to the ground truth. Furthermore, from the realistic case, MIML is the most accurate overall method for MWF estimation, showing the applicability to MWF estimation. In addition, we can see the robustness to changes in the assumed model for the T_2 distributions, and the importance of including the Wasserstein Distance in the loss function of MIML. **Finally, from examining all the reconstructed distributions and the mean reconstructed distribution, MIML is the most robust to noise, while the non-parametric methods show high variability and sensitivity to noise even at SNR 1000.** In the next section, we show results on real data from *in vivo* and *ex vivo* scans, considering both healthy and pathological cases.

6.4.2 Real Data

Ex vivo Data

We note that in Equation (4), the MWF is obtained by summing from $T_2 = 10$ms to $T_2 = 40$ms. This formula, commonly used for acquisitions at 3T, in theory should be adjusted for higher field strengths due to the shortening of T_2s [175], [176]. We note that these limits historically derive from assignment of the different lobes in T_2 distributions to different water pools e.g. myelin, IE space, etc.[147]. For instance, in [136], the authors use NNLS-T on their data (acquired at 1.5T) and found two large T_2 lobes, one in the range of 10-50ms and the other in the range of 70-100ms; they then assigned these to myelin water and the IES water respectively. In the following, we restrict our analysis to the white matter, and we will show two versions of MWF maps, with accompanying correlations to histology obtained as follows:

- Fixed Limits: Following [136], we fix the limits of summation for each method by taking the limits of the myelin water lobe in the mean T_2 distribution from using NNLS-T. This corresponds to bounds of 10-35ms.
- Tailored Limits: For each method, we set the limits of summation from the limits of the low T_2 lobe in the mean T_2 distribution from that method. For MIML and NNLS-L this corresponds to bounds of 10-38ms and 10-32ms respectively.

In Table 6.2, we show the spatial Pearson correlations (with accompanying p-values) be-

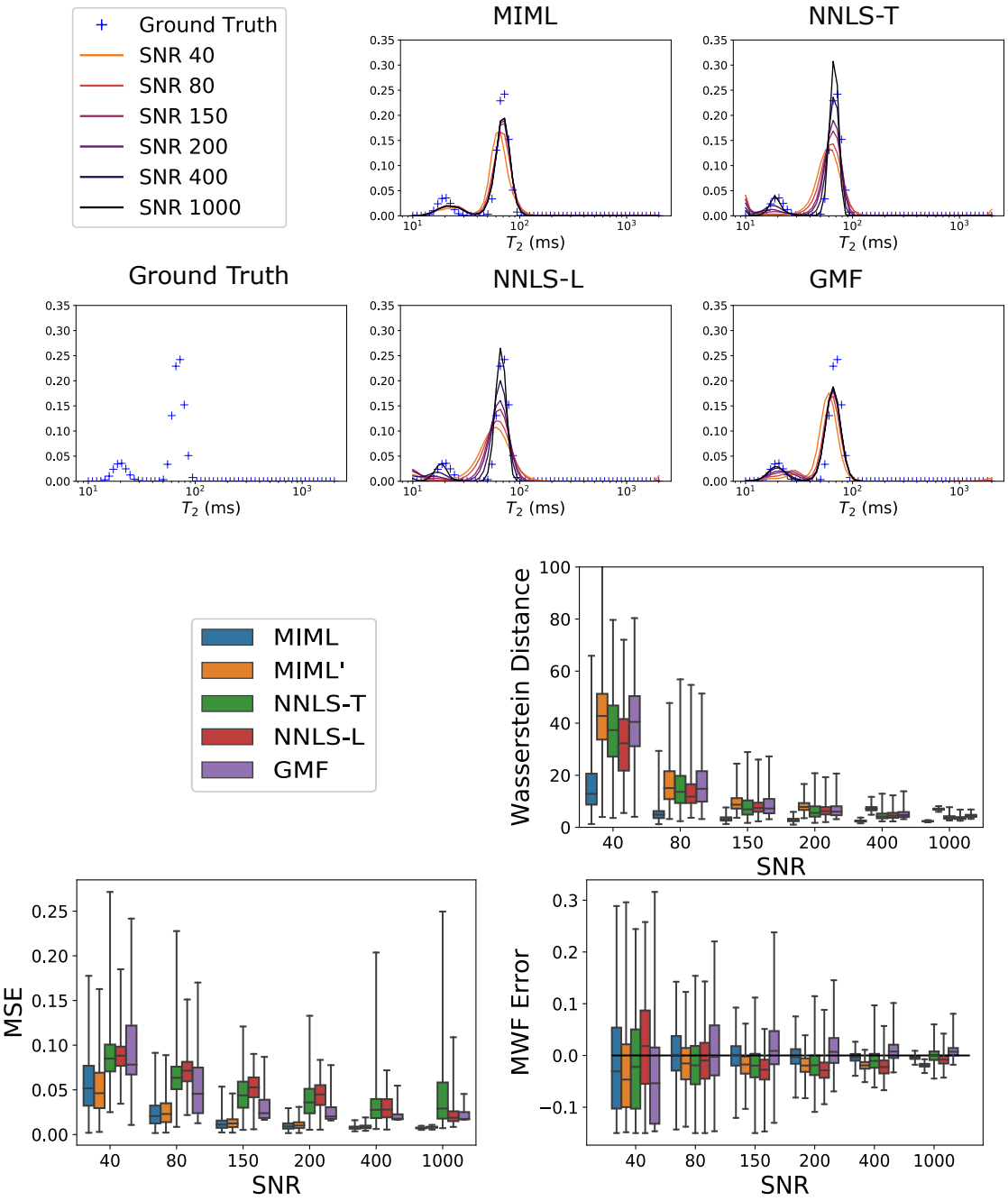


Figure 6.6: Mean reconstructed distributions (ground truth and from each method) over a range of SNRs as well as boxplots of the MSE and Wasserstein distance between the ground truth and reconstructed distributions from the results on the realistic, synthetic case. MIML produces the most robust reconstructions with respect to changing SNR, albeit with a consistently over-smoothed myelin water lobe. However, the other methods require high SNRs (1000) to resolve a myelin water lobe (still with distortions) close to the ground truth lobe as well as correct placement of the IE lobe. With regard to MSE and Wasserstein Distance, MIML performs the best, with the lowest median error and comparable or smaller interquartile range across the whole SNR range. With regard to MWF error (the ground truth MWF value is 0.15), MIML performs the best in the SNR range 80-400, with median errors closest to zero, and comparable or smaller interquartile ranges. MIML' performs similarly to MIML with respect to the MSE and significantly worse with respect to the Wasserstein Distance and MWF Error, showing the importance of using the Wasserstein Distance in the training of our method.

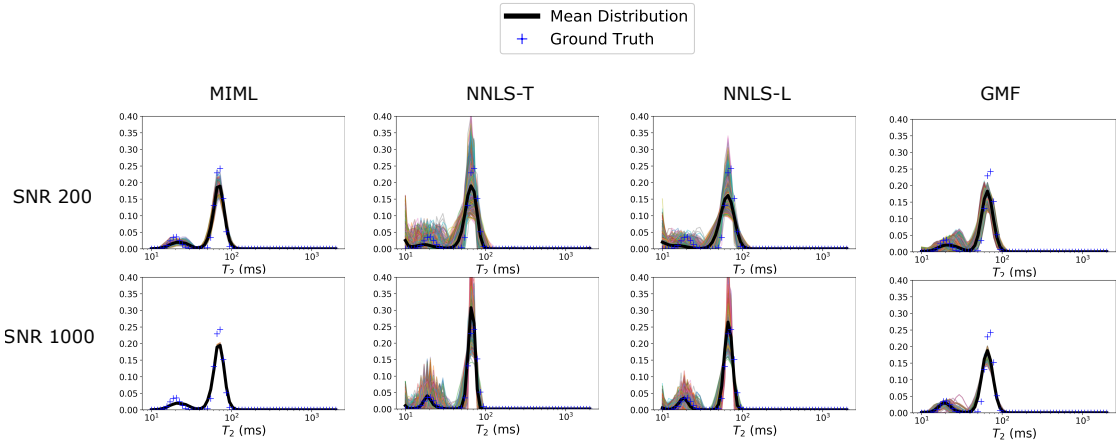


Figure 6.7: Distribution reconstructions from different noise realizations for SNRs 200 and 1000 on the realistic, synthetic case. The ground truth distribution is shown with blue crosses, and all the corresponding reconstructed distributions from different noise realizations are shown in color. The mean reconstructed distribution is shown in black Note that the reconstructions of NNLS-T and NNLS-L are highly sensitive to noise, even at SNR 1000, with large variability in the reconstructed distributions; this is in contrast to the stability and robustness of the reconstructions from MIML and, to a lesser extent, GMF. Note that for SNR 1000, MIML predicts virtually the same distribution for all the noisy signals. We use a logarithmic scale for the T_2 axis.

tween the MWF maps for each method and the histology map. In both cases, the MWF map from MIML has the highest correlation to the histology map. Only the correlation of NNLS-L changes between the two cases, increasing when using the fixed bounds. In Fig 6.8, we show the MWF maps corresponding to each case for the bounds, the histology map, and the reconstructed distributions for each method. MIML predicts higher values for the MWF than the other methods, particularly the NNLS methods. The MIML MWF map is smoother/less noisy than the other methods and corresponds better to the histology map. We can see from the mean distributions that all methods are, on average, able to recover the myelin water and IES water lobe in similar locations; however, the NNLS methods, in particular NNLS-L, produce more implausible, over-smoothed lobes in comparison to MIML and GMF. **Examining the reconstructed distributions, the influence of the model priors in MIML and GMF is evident, with clear separation between the myelin lobe and the IES lobe, while the NNLS methods produce distributions which are spread more uniformly across the T_2 axis.** We note that the small number of distributions with lobes in the range 200-1000ms and the lobes in the range 1000ms-2000ms can be attributed to the gray matter around the spine as well as CSF.

For all methods, the MWF values are significantly higher than those of the *in vivo* 3T scans we show later. However, this could be attributed to the differences resulting from the fact that *ex vivo* scan is of chemically treated spinal cord at 7T while the *in vivo* scans are of human brain at 3T.

Pearson Correlation of MWF Maps to Histology

	MIML	NNLS-T	NNLS-L	GMF
Tailored Bounds	(**0.54**,5.63E-81)	(0.44,8.58E-53)	(0.45,5.51E-55)	(0.39,1.43E-39)
Fixed Bounds	(**0.54**,2.04E-81)	(0.44,8.58E-53)	(0.49,1.13E-64)	(0.39,1.43E-39)

Table 6.2: Table of the spatial Pearson correlations (with p-values) between the MWF maps constructed from each method and the histology map of the myelin in a white matter mask. In bold are the highest correlations. We note that the histology map is not a map of the myelin water fraction, but a map of the fraction of pixels in the histology which correspond to the myelin tissue. In either case of fixed or tailored bounds, the MWF map from MIML has the highest spatial correlation to the histology.

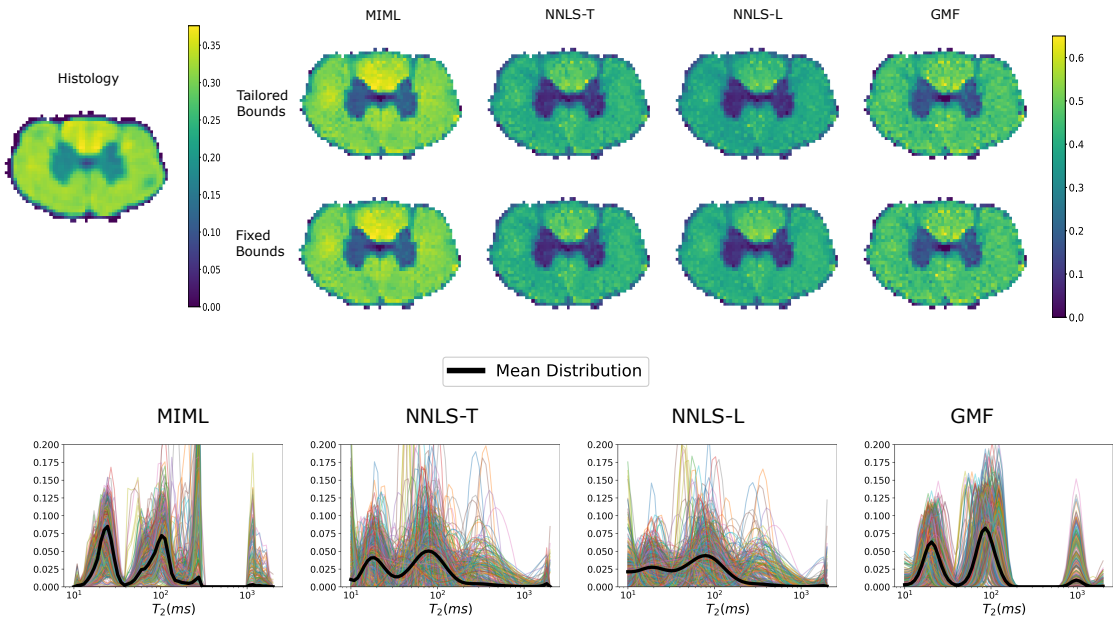


Figure 6.8: MWF maps from each method, the histology map, and the reconstructed distributions for each method on the ex-vivo data. We can see that the MIML MWF map is smoother than those of other maps and corresponds better to the histology map. The mean distribution is shown in black, and all the reconstructed distributions in color. All methods can recover the myelin water and IES water lobe in similar locations; however, the NNLS methods produce more smooth lobes in comparison to those of MIML and GMF. The effect of the model prior on MIML and GMF is clear, with unambiguous separation between the myelin lobe and the IES lobe. We emphasize that the histology map is a map of the fraction of the voxel which is occupied by myelin, not the MWF. We use a logarithmic scale for the T_2 axis.

Healthy Subjects

In Fig. 6.9 we show the MWF maps in axial, coronal, and sagittal slices for two healthy subjects with corresponding, registered MPRAGE images for comparison. In MPRAGE images, WM is hyperintense; hence, we treat the MPRAGE as a very rough proxy for the MWF map since MWF values are highest in the WM. **Although the MWF maps are fairly similar, the MWF map of MIML most accurately and smoothly conforms to the MPRAGE image.** The NNLS methods exhibit higher distortions, e.g. in the ventricles of subject 1, and difficulty in recovering the MWF in the frontal region of the brain. GMF produces maps comparable to the NNLS methods, albeit, looking noisier. We note that all methods exhibit lower MWFs in the frontal part of the brain as compared to other regions, which may stem from effects due to the gradient echo acquisition [177]. In Fig. 6.10, we show the reconstructed distributions over the WM voxels in the axial slices. Only MIML produces a mean WM distribution with two distinct, well-separated lobes corresponding to myelin water and the IES water as is expected from previous studies. Further, the peaks of the myelin water lobe and the IES water lobe correspond to the range expected at 3T. The NNLS methods recover the IES water lobe in line with expectations, but over-smooth the distribution in the region corresponding to myelin water, as was seen in the results on the synthetic data, with an implausible myelin water peak at 10ms. GMF also recovers the IE lobe in line with expectations, but produces a dispersed lobe in the myelin region. From the reconstructed distributions, we can again see the influence of the model priors on MIML and GMF, with the NNLS methods producing much more variable distributions. We note that the small component in the range 1000ms-2000ms for each method can be attributed to partial volume effects with the CSF.

In order to compare the MWF maps on regions of interest, and to conduct the scan-rescan analysis we did the following: in a first step, all the estimated MWF images for the 4 subjects were registered to the 'ICBM-DTI-81' white-matter tract labels atlas [178], [179] using the non-linear registration 'BSplineSyN' algorithm included in the ANTs software (https://github.com/ANTsX/ANTs). After visually inspecting the images, we removed small ROIs affected by registration errors and kept 44 tract labels showing a good anatomical agreement between the atlas and subject native spaces. Finally, the mean MWF value and the coefficient of variation of the MWF for each region of interest (ROI) was calculated for the scan and rescan maps from each method. A list of the ROIs can be found in Table 6.3.

In Fig. 6.11, we show boxplots of the mean MWF and the standard deviation of the MWF over the WM ROIs for each subject. We note that MIML results in a larger mean MWF across all subjects than the non-parametric methods; this can be explained by the underestimation of MWF by the non-parametric methods as is shown in the results on the realistic synthetic data as well as in [154]. However, the standard deviations of the MWF from MIML are generally comparable to that of the other methods, similar to that of NNLS-T and slightly higher than that of NNLS-L. In particular, the magnitude of the increase in the mean MWF using MIML as compared to the other methods is larger than the increase in the standard deviation of the MWF. This indicates that the smoothness of the MIML MWF map is comparable to that of

List of Brain ROIs used for Healthy Subjects

1	Middle cerebellar peduncle
2	Pontine crossing tract
3	Genu of corpus callosum
4	Body of corpus callosum
5	Splenium of corpus callosum
6	Fornix
7	Corticospinal tract R
8	Corticospinal tract L
13	Superior cerebellar peduncle R
14	Superior cerebellar peduncle L
15	Cerebral peduncle R
16	Cerebral peduncle L
17	Anterior limb of internal capsule R
18	Anterior limb of internal capsule L
19	Posterior limb of internal capsule R
20	Posterior limb of internal capsule L
21	Retrolenticular part of internal capsule R
22	Retrolenticular part of internal capsule L
23	Anterior corona radiata R
24	Anterior corona radiata L
25	Superior corona radiata R
26	Superior corona radiata L
27	Posterior corona radiata R
28	Posterior corona radiata L
29	Posterior thalamic radiation R
30	Posterior thalamic radiation L
31	Sagittal stratum R
32	Sagittal stratum L
33	External capsule R
34	External capsule L
35	Cingulum (cingulate gyrus) R
36	Cingulum (cingulate gyrus) L
37	Cingulum (hippocampus) R
38	Cingulum (hippocampus) L
39	Fornix/Stria terminalis R
40	Fornix/Stria terminalis L
41	Superior longitudinal fasciculus R
42	Superior longitudinal fasciculus L
43	Superior fronto-occipital fasciculus R
44	Superior fronto-occipital fasciculus L
45	Uncinate fasciculus R
46	Uncinate fasciculus L
47	Tapetum R
48	Tapetum L

Table 6.3: Here we show a table of the regions of interest (ROI) in the brain used to estimate the coefficient of variations in the scans, as well as the comparisons for the scan-rescan analysis

Mean and Standard Deviation of MWF Differences between Scan and Rescan WM ROIs

	MIML	NNLS-T	NNLS-L	GMF
Subject 1	(0.0067,0.0388)	(0.0093,0.0353)	(0.0094,**0.0305**)	(**0.0055**,0.0357)
Subject 2	(**0.0004**,0.0448)	(0.0012,0.0473)	(0.0010,0.0418)	(0.0044,**0.0454**)
Subject 3	(0.0191,0.0726)	(0.0134,0.0755)	(0.0176,0.0723)	(**0.0108**,**0.0712**)
Subject 4	(0.0104,0.0573)	(0.0107,0.0543)	(0.0103,**0.0504**)	(**0.0065**,0.0561)

Table 6.4: Table of the mean and standard deviation of the absolute difference between the mean MWF values of the scan and rescan in white matter ROIs for each method and for each healthy subject. In bold are the lowest values per subject. Overall, GMF has the smallest mean differences for 3/4 subjects with standard deviations comparable to those of other methods. The performance of MIML and the NNLS methods are overall quite similar; while MIML has the smallest mean difference on Subject 2, NNLS methods have smaller mean differences in Subjects 3 and 4.

Pearson Correlation and Linear Regression Coefficients between Scan and Rescan WM ROIs

	MIML	NNLS-T	NNLS-L	GMF
Subject 1	(0.92, 0.89,0.0069)	(0.91,0.887,0.0016)	(**0.93**,0.90,0.0004)	(**0.93**,0.85,0.0130)
Subject 2	(0.90,1.00,0.0002)	(0.87,0.98,0.0008)	(0.89,1.02,-0.0034)	(**0.91**,0.99,0.0045)
Subject 3	(**0.77**,0.74,0.0546)	(0.70,0.65,0.0527)	(0.72,0.69,0.0520)	(**0.77**,0.76,0.0433)
Subject 4	(0.87,0.87,0.0087)	(0.87,0.97,-0.0064)	(**0.89**,0.97,-0.0076)	(**0.89**,0.96,0.0001)

Table 6.5: Table of the spatial Pearson correlation and the linear regression coefficients (slope and intercept) between the mean MWF values of the scan and rescan in white matter ROIs for each method and each healthy subject. In bold are the highest Pearson correlations per subject. All Pearson correlations have p values less than 0.01. Overall, all methods perform quite similarly. However, GMF has the best correlations between scans (by a small margin), with MIML and the NNLS methods performing similarly.

other methods.

In Tables 6.4 and 6.5, we show the results of our scan-rescan analysis over all four healthy subjects; we show a table of the mean and standard deviation of the absolute difference between the mean MWF values of the scan and rescan in the specified ROIs as well as a table of the Pearson correlation and linear regression coefficients between the mean MWF values of the scan and rescan in the specified ROIs. We can see that in general, GMF provides the smallest mean differences and highest Pearson correlations. In particular, it is difficult to rank MIML and the NNLS methods as they perform better/worse on different subjects. We note that GMF's superior reproducibility may stem from the lower flexibility in the fitting of the MWF, as compared to MIML and the NNLS methods. However, overall, the reproducibility of the methods is quite similar.

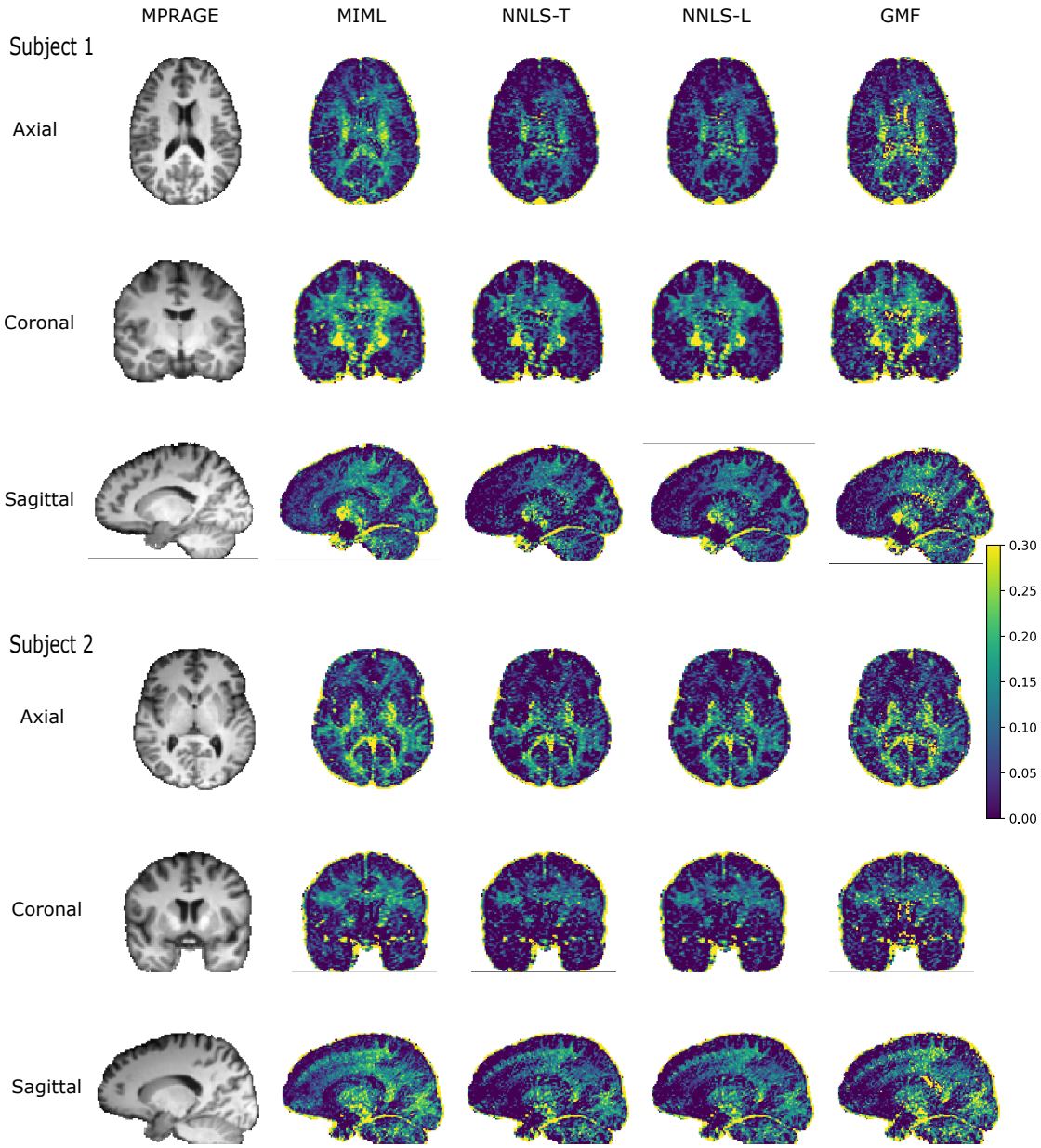


Figure 6.9: Example MWF maps produced from each method, in the axial, coronal, and sagittal planes of two healthy subjects. On the left, we show the corresponding MPRAGE slice. Compared to the MPRAGE (where WM is hyper-intense), we can see that MIML most accurately and smoothly reproduces the extent of white matter, which is consistent with WM having relatively high MWF values. Particularly, the NNLS methods struggle in MWF recovery in the frontal part of the brain. GMF produces comparable to better MWF recovery than the NNLS methods, but with a noisier map. In addition, MIML has the least distortion in the ventricles.

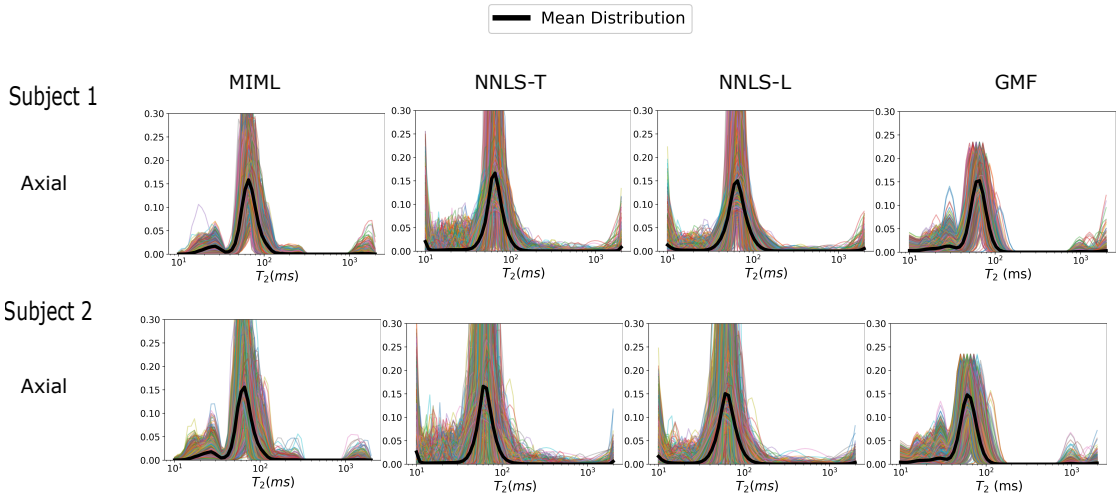


Figure 6.10: Reconstructed distributions (in color) in the WM voxels of the axial slices of two healthy subjects for each method. The mean distribution is shown in black. We note that only MIML produces a mean WM distribution with two distinct, well-separated lobes, and the myelin water peak in line with expectations at 3T. The NNLS methods and GMF recover the IE lobe well, but the myelin lobe is either irregular or appears at an extremely low T_2. Further, we see that NNLS-T/L produces a much more variable set distributions in contrast to those from MIML and GMF which are constrained by model priors. We use a logarithmic scale for the T_2 axis.

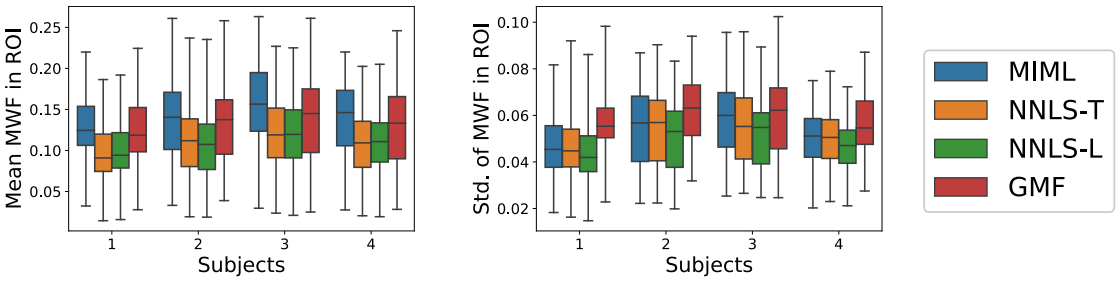


Figure 6.11: Boxplots of the mean MWF (left) and the standard deviation of the MWF (right) for each method over all the WM ROIs for each subject in the cohort of healthy subjects. We see that MIML produces a larger mean MWF across all subjects than the other methods. This is likely due to the underestimation of MWF by the non-parametric methods as is shown in the results on the realistic synthetic data as well as in [154]. The standard deviations of the MWF from MIML are generally comparable to that of the other methods, similar to that of NNLS-T and slightly higher than that of NNLS-L. We note that the magnitude of the increase in the mean MWF using MIML as compared to the other methods is larger than the increase in the standard deviation of the MWF. This indicates that the smoothness of the MIML MWF map is comparable to that of other methods.

MS Subject

In Fig. 6.12, we show the maps of the geometric mean T_2 in the IE range of 50-200ms as well as the MWF maps in an axial slice of a subject with MS. In addition, in Fig. 6.13, we zoom in on the lesions for better visualization. **For the mean T_2 maps, in all methods, all except one of the lesions can be clearly seen as hyperintensities i.e. with increased mean IE T_2. Further, the maps are similar across the methods, with the main differences residing in the ventricles.** Visualizing the lesions is far more difficult with MWF maps than with the mean T_2 maps, as the MWF maps are much noisier independent of the applied method. However, as with the healthy subjects, the MIML MWF map in both slices most smoothly and accurately conforms to the WM and the cortices, with the other methods exhibiting more variability and missing patches in the WM and worse delineation of the cortices; this occurs particularly in the frontal region. All three lesions can be seen on the MIML MWF map with minimal ambiguity; in particular, in lesions 1 and 3, we can clearly delineate the lesions from very close, adjacent structures. Concerning the NNLS methods, it appears that Lesion 1 is exaggerated in size and mixed with the adjacent structure, making it difficult to delineate the lesion as the dark region is extended far beyond the lesion region on the FLAIR image. In addition, due to poor contrast between the normal-appearing tissue and lesion tissue/noise, it is difficult to identify Lesion 2 unambiguously with the NNLS methods. As with Lesion 1, Lesion 3 can be seen but is connected to the adjacent grey matter, making localization problematic. Further, we can see that the MWF in the lesion is comparable to the MWF of the normal-appearing, contralateral brain region, due to the poor MWF reconstruction. The GMF MWF map resembles the MIML MWF map albeit noisier/ with greater variability.

In addition to the mean T_2 and MWF maps, in Fig. 6.14, we compare the T_2 distributions in the lesion masks to the T_2 distributions in the normal appearing, contralateral regions. **As in the healthy subjects, MIML consistently produces a mean distribution with two distinct, well-separated lobes corresponding to myelin water and the IES water as is expected from previous studies.** Further, the peaks of the myelin water lobe and the IES water lobe correspond to the range expected at 3T. The NNLS methods produce over-smoothed myelin water lobes with peaks occurring at implausibly low T_2 values. The IES water lobes are generally plausible. GMF produces more plausible myelin water lobes than those of the NNLS methods, but the lobes in the contralateral tissue are more variable, with the estimated mean and standard deviation of the myelin lobes varying significantly over the 3 regions of contralateral tissue. MIML reconstructs a diminished myelin water lobe in the lesions as compared to the normal-appearing tissue, reflecting lower MWF; this is in line with expectations of MS as a demyelinating disorder. In contrast, the distributions from the NNLS methods in Lesions 2/3 exhibit larger myelin water lobes in lesion tissue as compared to normal-appearing tissue, indicative of the poor MWF reconstruction in the normal-appearing tissue.

In conclusion, all methods perform similarly in detecting lesions from the mean T_2. However, MIML improves upon the NNLS methods and GMF in detecting lesions from MWF maps, by providing better contrast between lesions and normal appearing tissues, clearer delineation of

Average Computation for Whole Brain

	MIML	NNLS-T	NNLS-L	GMF
Time	34s	752s	701.2s	159382.4s

Table 6.6: Table of the average computation time for whole brain reconstructions for each method on the healthy subjects; all reconstructions were done using the same computer, with 16 threads. MIML is orders of magnitude faster than the other methods.

lesions from adjacent structures, and smoother, more plausible reconstructions overall in the WM. The comparison of the myelin water lobes of lesion and normal appearing tissue from MIML is consistent with the demyelinating nature of MS in contrast to that from the NNLS methods. Therefore, the performance of MIML meets or exceeds the performance of the other methods when used on a pathological case.

From our results on real data, we see that MIML generalizes to different machines, different magnetic field strengths, and different sequences since it is trained on a model of the signal decay which is agnostic to these differences; MIML's performance on the real data shows its potential for multi-component T_2 relaxometry at clinically achievable SNRs in high resolution scans.

6.4.3 Computation Time

Here we provide a brief overview of the computational cost of the different methods. For consistent comparison, we used one computer using Ubuntu 18.04 with an Intel Xeon CPU E5-1650v4 running at 3.6 GHz with 12 available threads to run parallelized whole-brain reconstructions on four of the healthy subjects (matrix size 144x126x84) using MIML, NNLS-T, NNLS-L, and GMF; we recorded the time to completion and show the average computation time for each method in Table 6.6. We can see that MIML is 1 to 4 orders of magnitude faster than the other methods.

6.5 Discussion

Overall, from our evaluation on synthetic data, an *ex vivo* scan and *in vivo* scans (healthy and pathological), we conclude that MIML provides fast, noise-robust, and plausible reconstructions of T_2 distributions, with potential for use in myelin water fraction mapping. We attribute the performance of our method to the blending of the advantages of machine learning, parametric, and non-parametric methods. We note that our approach is essentially using machine learning to solve the inverse problem of parametric approaches, albeit expressing the solution non-parametrically. We view our approach as an extension of the recent progress in using machine learning to solve inverse problems in many domains [180]. By using machine learning, our method is much faster than standard parametric or non-parametric

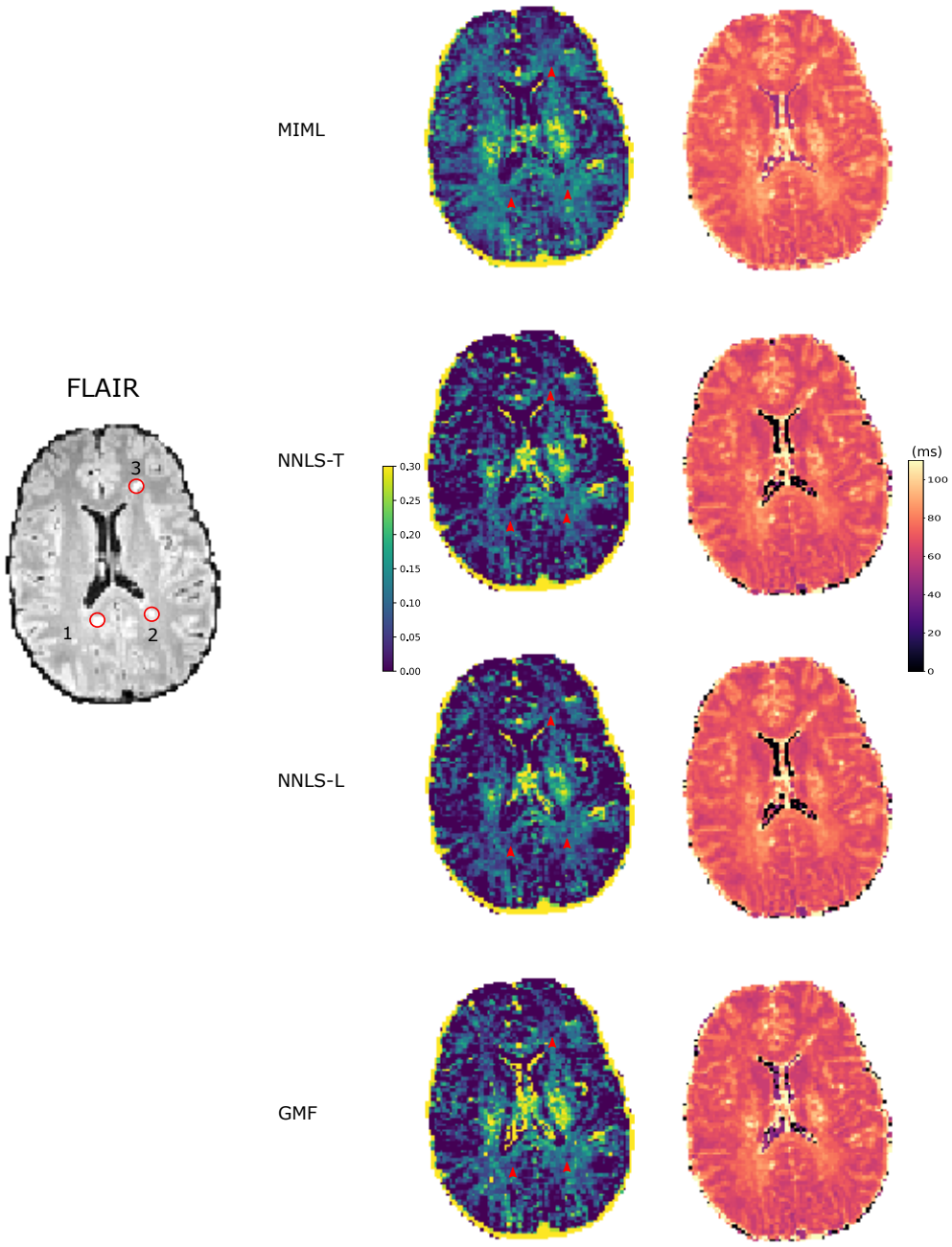


Figure 6.12: Anatomical FLAIR map where the white matter is hypointense (first column), maps of the MWF (second column), and maps of the geometric mean T_2 in the range corresponding to the IE space (50-200ms) (third column) for an axial slice in a subject with MS. We show the MS lesions on the FLAIR map marked in red and labeled numerically. Regarding the mean T_2 maps, we can see that the all lesions but Lesion 2 can be seen as hyperintensities, with the maps very similar across all methods. Regarding the MWF maps, as in the healthy subjects, MIML most smoothly and accurately reconstructs the WM, with the other methods exhibiting more noisy maps with missing patches. MIML provides the best lesion visualization due to better contrast between normal appearing tissue and lesions and a more smooth MWF map; in particular, lesions can clearly be delineated from close, adjacent structures in contrast to the NNLS methods (see Lesion 1, 3). See Fig. 6.13 for a closer look/analysis of the MWF maps compared to the lesions.

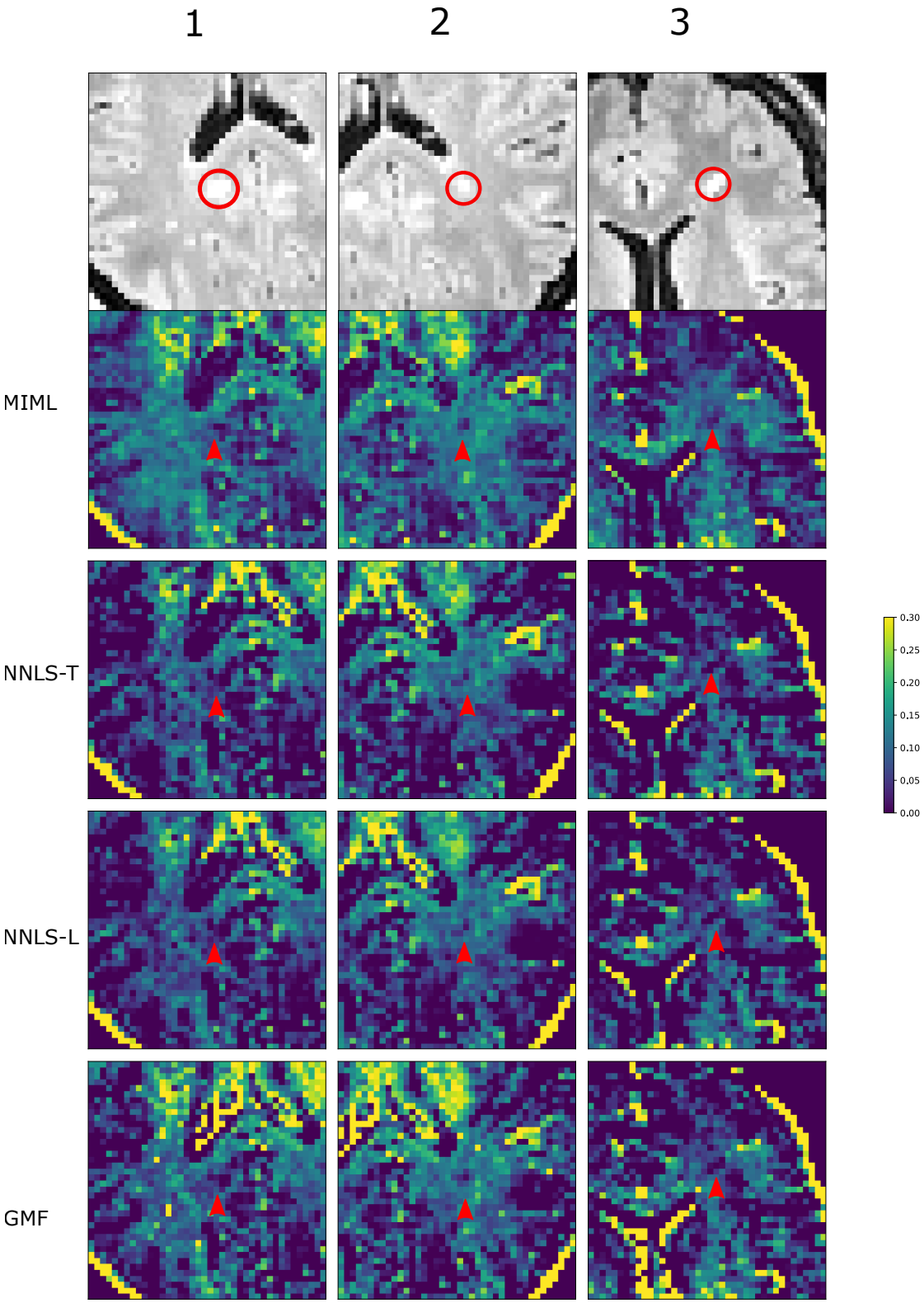


Figure 6.13: Here we zoom in on the lesions as well as the corresponding patches in the MWF map. We are able to see distinctly see all the lesions using the MIML MWF map; in particular, lesions 1 and 3 can be clearly distinguished from close, adjacent structures. Due to the lower contrast between normal appearing tissue and lesion tissue and noisier appearance in comparison to the MIML MWF map, lesions 1 and 2 are somewhat ambiguous on the NNLS maps; in particular, it appears the lesion 1 is exaggerated in size and mixed with the structure next to it. Similarly Lesion 3 is mixed with the structure next to it with the NNLS maps. The GMF MWF maps are similar to those of MIML, albeit noisier.

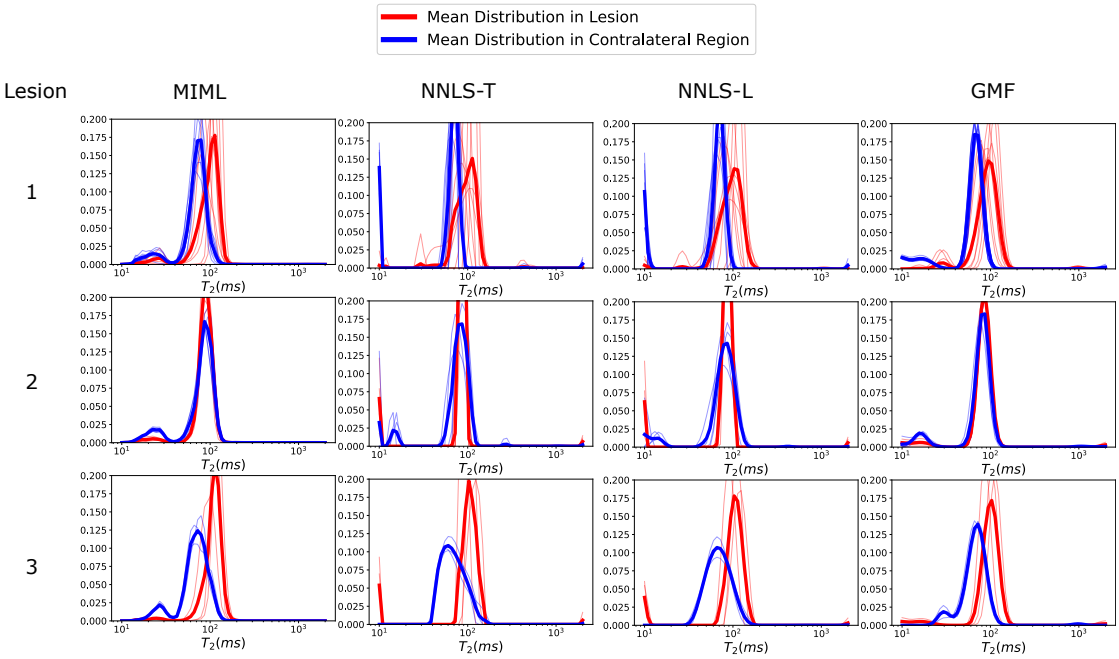


Figure 6.14: Here we compare the T_2 distributions within the lesion mask and the same mask translated to the normal appearing region contralateral to the lesion for each lesion and method. We can see that MIML consistently produces a mean distribution with two distinct, well-separated lobes, and the myelin water peak in line with expectations at 3T. In general, the NNLS methods and GMF recover the IE lobe well, with occasional noise, but the myelin water lobes are over-smoothed with peaks at implausibly low values or irregular. We note that MIML finds a diminished myelin water lobe in the lesion as compared to the normal appearing tissue, in line with expectations of MS as a demyelinating disorder; in contrast, the NNLS methods in Lesions 2/3 exhibit larger myelin water lobes in lesion tissue as compared to normal appearing tissue. GMF performs similarly to MIML in this regard, albeit with more irregular distributions, due to the more variable reconstruction in the myelin lobe in the contralateral tissue. We use a logarithmic scale for the T_2 axis.

approaches. By training on solely simulated data, our approach does not require expensive, *in vivo* acquisitions for training data, nor the need for multiple scans to adapt to different machines or sequences. Further, by simulating random flip angles in the dataset, our method is able to automatically account for the flip angle, in contrast to non-parametric methods which need to estimate the flip angle before fitting. Altogether, this allows for noise-robust reconstruction by training the network on simulated signals with an SNR range and noise model corresponding to those from clinical scans. By generating the simulations guided by biophysical models, we can simultaneously retain stability in the reconstruction by constraining the space of T_2 distributions while not being restricted to a specific number of water pools at inference time. Further, the produced distributions are implicitly constrained to have a plausible, lobular structure (as in parametric approaches), which makes the interpretation of parameters of interest such as the MWF consistent with past studies, in contrast to potential irregular distributions from non-parametric methods. The trained MIML model and code for generating the synthetic data and training the model is available at the following website: https://github.com/thomas-yu-epfl/Model_Informed_Machine_Learning.

However, our current approach has several limitations. First, while we attempted to be as comprehensive as possible in the simulated dataset, advances in biophysical modelling make it possible that there are additional relevant water pools to be estimated. For example, the Gaussian Mixture model we use assumes the symmetry of the mixture distributions, which may not be true in real distributions; in the case of skew, ground truth distributions, our method can result in a biased reconstruction. Second, while we fixed the Rician noise model for the training signals, with a fixed SNR range of 80-200, we note that in some sequences, more complex noise models such as the non-central chi distribution [181] with different SNR ranges may also be appropriate. Third, we only consider 32-echo sequences in this chapter. Fourth, we use a fixed, logarithmic T_2 discretization consisting of 60 points from 10ms to 2000ms for both our method and the NNLS methods. However, finer or coarser discretizations could also have been used. Finally, there may be relevant physical effects such as magnetization transfer [182], [183] which, if modelled in the dataset, could improve the reconstructions. In particular, it has recently been shown [184] that the T_2 of different compartments in white matter show an orientation dependence with respect to the main magnetic field, with concomitant effects on the estimation of the MWF, for example.

However, we highlight the flexibility and modularity of our approach for accounting for these limitations. Additional water pools can be easily added to the training dataset. The noise model and SNR range used in training can be swapped out for different noise models and SNRs. A sequence with a different number of echoes can be accommodated by reconstructing the dataset with the required number of echos and retraining the network. Different T_2 discretizations would simply require downsampling of the high resolution T_2 distributions in our dataset to match the new discretization, with subsequent retraining of the network. More advanced physical modelling can be added to the generation of new datasets. As the training of the network is quite fast (70s on a laptop GPU), the bottleneck for addressing these limitations is the dataset generation (1 hour on 46 CPU threads). However, while we generated

our dataset on CPU, GPU acceleration of the EPG formalism can potentially speedup dataset generation significantly [185].

As for future work: in this chapter, we did not study the impact of denoising the data on the reconstruction performance of the methods compared. This is first because in our overview of the literature, we found that presenting results on denoised data is not typical unless the subject of the paper is denoising. Second, the type of denoising, setting of denoising parameters, and accounting for potential biases due to denoising all require careful justification and study, which we felt was out of the scope of this paper, which introduces a proof of concept. However, we note that in the MS data, particularly for the NNLS methods, ostensibly normal appearing regions of the brain had unusually low MWF values, sometimes less than that predicted for the lesion. These areas of unusually low MWF values could also be seen in the scans of healthy subjects. These may be due to, in part, instability/ill-posedness in the estimation due to comparatively low SNRs in the *in vivo* scans; the *in vivo* scans we used have fairly high resolution (1.6-1.8mm) and are isotropic, while typical scans in the literature generally use much thicker slices ($\geq$ 2mm) along the axial direction [130], [147]. We note that both distributions and MWF maps from the NNLS methods were more plausible in the *ex vivo* scan, where the SNR was much higher. This is consistent with the observations in [154] concerning the noise dependence of NNLS methods. Future studies will be conducted to study the impact of denoising algorithms such as PCA denoising [186], or the NESMA filter [187] on MIML as well as other methods, and any effect this has on their comparison.

Our method, as well as the other methods compared to in this chapter, reconstruct the T_2 distribution in each voxel separately. However, there are parametric and non-parametric approaches to T_2 relaxometry which use spatial regularization [188]–[190]. These approaches assume that voxels spatially close to each other should also have similar reconstructions; hence, they perform reconstructions on groups of adjacent voxels simultaneously, with constraints that limit the variation of the reconstructions over the group. In addition, another approach estimates over groups of voxels by assuming the joint sparsity of the distributions in a region of interest [191]. In future work, we will study how regularization/simultaneous fitting over regions of interest can be incorporated into our machine-learning framework as well as its effects on distribution reconstruction.

In this paper, we tested our method on two types of sequences: a multi-echo spin echo sequence and a 3D gradient and spin echo sequence. While in principle our approach is agnostic to the sequence used, in the future we will further validate our method on data from other sequences such as the T_2 prepared gradient echo sequences [192].

Finally, we note that using more advanced neural networks such as Long short term memory (LSTM) networks [193], which are suitable for time series data, may offer improved reconstructions as well as potentially eliminating the need for fixed size inputs. In addition, while our synthetic dataset generation is based solely on the most common cases for biophysical modelling, we will investigate how to improve dataset generation i.e. the number of pools, the

maximum number of pools present per signal, etc. in order to optimize the generalization capabilities of the network while minimizing the ill-posedness of the reconstruction.

6.6 Conclusion

In this chapter, we presented Model-Informed Machine Learning (MIML), an approach for estimating T_2 distributions from MRI signals using a neural network trained on synthetic data derived from biophysical models. Through our evaluations on synthetic data, an *ex vivo* scan, as well as healthy and pathological *in vivo* data, we show that MIML provides more robust, accurate, and plausible T_2 distributions than standard parametric and non-parametric methods across a wide range of SNRs. We show that MWF maps derived from MIML show the highest conformity to anatomical scans, have the greatest correlation to a histological map of myelin volume, and improve upon the lesion visualization capabilities of other methods, with better contrast between lesions and normal-appearing tissue as well as clearer delineation between lesions and close adjacent structures. The code for generating the datasets and training the network is available at https://github.com/thomas-yu-epfl/Model_Informed_Machine_Learning.

7 Using Realistic and Bicubic Downsampling for Super-Resolution

The content of the following chapter is based on the postprint version of the article: "Benefiting from Bicubically Down-Sampled Images for Learning Real-World Image Super-Resolution" published in the Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision [194]. DOI: 10.1109/WACV48630.2021.00163.

7.1 Introduction

Similarly to Chapter 6, in this chapter we consider how realistic models can inform supervised methods for solving inverse problems where no, or only small amounts of ground truth data are available; in contrast to Chapter 6, instead of doing so through the synthetic generation of a large, realistic dataset, we leverage a small quantity of realistic, physically generated data such that we can improve upon and reuse existing supervised methods trained on large, unrealistic datasets.

In this chapter we shift away from MRI related inverse problems to the inverse problem corresponding to single image super resolution (SR), where we want to recover a high-resolution image from a low-resolution image; concretely, a general, analytical model for image degradation which is commonly assumed is

$$\mathbf{y} = (\mathbf{x} * \mathbf{k}) \downarrow_s + \mathbf{n} \tag{7.1}$$

where $\mathbf{y}$ is a low-resolution(LR) image, $\mathbf{x}$ is the corresponding high-resolution(HR) image, $*$ denotes convolution, $\mathbf{k}$ is a blur kernel, $\mathbf{n}$ is noise, and $\downarrow_s$ denotes downsampling by a factor s. We note that while MR super-resolution is a field in its own right, the realistic modelling considered in this chapter applies only to camera images, making transferability of this framework to MR super-resolution difficult. However, this problem has some connection to the undersampled MR image reconstruction problem in that in both cases, in some sense,

Figure 7.1: An example SR produced by our system on a real-world LR image, for which **no higher resolution/ground-truth is available**. Our method is compared against the RealSR [195] method, a state-of-the-art of real SR method trained in a supervised way on real low-resolution and high-resolution pairs. The low-resolution image is taken from HR images in the DIV2K validation set [196].

one needs to compensate for the lack of information of the input to produce a "high quality" image. In undersampled MR, one is implicitly trying to fill in the missing k-space values by reconstructing a high quality image. In super-resolution, one is trying to increase the resolution of a LR image, while maintaining high frequency detail, which is equivalent to filling in the Fourier components (or k-space) of the HR image corresponding to high frequencies.

In recent years, supervised, deep learning methods which implicitly imbed the above model through training sets have become the state of the art for solving the SR inverse problem. However, since the first introduction of deep learning for SR [197], the training datasets for proposed methods have been synthetically generated from bicubically downsampling high quality images [197]–[201]. Partly this was due to the desire for standardizing datasets for fair comparisons between methods. However, although the performance of these methods on bicubically downsampled images are quite impressive [202], [203], applying these methods on real-world LR images, with unknown degradations from cameras, cell-phones, etc. often lead to poor results [195], [204]. This indicates that training on datasets from bicubic downsampling,

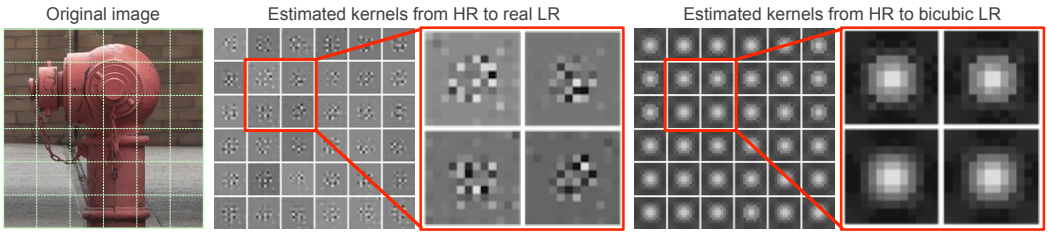


Figure 7.2: Downsampling kernels estimated patchwise on a RealSR [195] LR image and the same image bicubically downsampled from the HR image. Estimations were done using least squares optimization with regularization on the kernel using the LR and HR images, assuming the standard degradation model of kernel convolution followed by subsampling. We can see that the RealSR LR images are difficult to estimate with the standard image degradation model.

in general, is not realistic enough for generalizable performance. Therefore, there remains the real-world SR problem: to super-resolve LR images downsampled by unknown, realistic image degradations [205].

Recent works try to model realistic degradations by acquisition instead of artificial downsampling, such as hardware binning, where the downsampling corresponds to a coarser grid of photoreceptors of the camera [206], or camera focal length changes, which changes the apparent size of an object [195]. The latter, in particular, is compelling as a model for realistic downsampling as the resulting solution of the SR problem can be interpreted as ”zooming in” on in image, which is intuitively what is desired in SR. However, in terms of training datasets, these approaches propose a very limited number of physically real low and high-resolution pairs in comparison to the typically used synthetically generated datasets. Furthermore, as shown in [207], correct modeling of the image degradation is crucial for accurate super-resolution. However, as can be seen in Figure 7.2, even Equation 7.1 is not general enough to encompass the true downsampling model of [195]. Therefore, the real-world SR problem is in the regime where the model M is either unknown or cannot easily be expressed accurately in an analytical fashion. Therefore, a naive, implicit supervised approach may not have a sufficient amount of data for optimal learning.

Recently, there has been a push to account for more realistic image degradations through implicit supervised methods on physical generated datasets with real LR to HR pairs [195], [206], synthetically generating real LR to HR pairs through unsupervised learning or blind kernel estimation [204], [208], and simulating more complex image degradation models such as in Equation 7.1, with and without restrictions on $\mathbf{k}$ and $\downarrow_s$ [209], [210]. The pipelines of these approaches generally have the ultimate goal of training an end-to-end network to take as input a ”real" image and output a HR image. Although these approaches result in better reconstruction quality, the real challenge of the real-world LR to HR problem is not only limited to a lack of real LR and HR pairs; the large variety of degraded images and the difficulty in accurately modeling the degradations makes realistic SR even more ill-posed than SR based on bicubically down-sampled images [211].

Main idea We propose to address real world SR with a two-step approach, which we call Real Bicubic Super-Resolution (RBSR). RBSR generally decomposes the difficult problem of real world SR into two, sequential subproblems: **1-** Transformation of the wide variety of real LR images to a single, tractable LR space. **2-** Use of generic, pretrained SR networks **trained on bicubically downsampled datasets** with the transformed LR image as input.

We choose to transform real LR images to the common space of bicubically downsampled images because of two main advantages. First, bicubic images are tractably generated with the standard convolutional model of image degradation in Equation 7.1, therefore the inverse transform is less ill-posed compared to the cases of arbitrary/unknown degradations. Second, we can leverage the already impressive performance of SR networks trained on bicubically downsampled images, thanks to the availability of huge SR image datasets using bicubic kernels (see Figure 7.1); as bicubically downsampled images are still at least somewhat related to realistically downsampled images, we can still hope that the impressive performance of bicubic SR networks will transfer over to real-world images once we approximately convert inputs to bicubically downsampled inputs. As the transformation of realistic LR images to bicubically downsampled LR images is less complex than the transformation of realistic LR images to HR images, we are essentially splitting the SR inverse problem into two, more tractable inverse problems. In this way, we are able to combine all the available model-driven (small, realistic training datasets with intractable models M) and data-driven (networks pretrained on large datasets of unrealistic training data) information.

7.2 Contributions

1. We use adversarial training for a CNN-based image-to-image translation network, which we call a "bicubic look-alike generator", to map the distribution of real LR images to the easily modeled and well understood distribution of bicubically downsampled LR images. We use a SR network with the transformed LR image by our proposed bicubic look-alike generator as input to solve the **real-world super-resolution** problem.

2. To this end, and for the consistency of the bicubic look-alike generator, we propose a novel copying mechanism, where the network is fed with identical, bicubically downsampled images as both input and ground-truth during training; this way, the network loses its tendency to merely sharpen the input images, as realistic low-resolution images usually seem to be much smoother.

3. We train our bicubic look-alike generator by using an extended version of perceptual loss, where its feature extractor is specifically trained for SR task and on bicubically downsampled images. The proposed "bicubic perceptual loss" is shown to result in less artifacts.

4. We demonstrate the effectiveness of the proposed two-step approach by comparing it to an end-to-end setup, trained in the same setting. Furthermore, we show that our

proposed approach outperforms the state-of-the-art works in terms of both qualitative and quantitative results, as well as results of an extensive user study conducted on several real image datasets.

In summary, training models on paired datasets of real LR and HR pairs requires expensive collection of big datasets; in addition, training a single model on multiple degradations for SR is ill-posed/vulnerable to instability [211]. Training on synthetic datasets coming from analytical degradation models have the benefit of much larger datasets and an easier task for the network, at the cost of being less realistic. However, this approach still has the ill-posedness problem of training on multiple degradations. In RBSR, we try to simultaneously keep the added information from realistic LR images and the impressive performance of SR networks on single, well-defined degradations.

7.3 Related Work

The vast majority of prior work for Single image super-resolution (SISR) focuses on super-resolving low-resolution images which are artificially generated by bicubic or Gaussian downsampling as the degradation model. We consider that recent research on addressing real-world conditions can be broadly categorized into two groups. The first group proposes to physically generate new, real LR and HR pairs and/or learn from real LR images in supervised and unsupervised ways (Section 7.3.1). The second group extends the standard bicubic downsampling model, usually by more complex blur kernels, and generates new, synthetic LR and HR pairs (Section 7.3.2).

7.3.1 Real-World SR through real data

Some recent works [195], [212] propose to capture real LR/HR image pairs to train SR models under realistic settings. However, the amount of such data is limited. The authors in [195], [212] proposed to generate real, low-resolution images by taking two pictures of the same scene, with camera parameters all kept the same, except for a changing camera focal length. Hence, the image degradation corresponds to "zooming" out of a scene. They generate a dataset of real LR and HR pairs according to this procedure and show that bicubically trained SR models perform poorly on super-resolving their dataset. Since this model's image degradation can be modeled as convolution with a spatially varying kernel, they propose to use a kernel prediction network to super-resolve images. In [204], the authors perform unsupervised learning to train a generative adversarial network (GAN) to map bicubically downsampled images to the space of real LR images with two unpaired datasets of bicubically downsampled images and real LR images. They then train a second, supervised network to super-resolve real LR images, using the transformed bicubically downsampled images as the training data. In a similar work, [213] trains a GAN on face datasets, for the specific face SR task, but their approach relies on unrealistic blur-kernels.

In [214], the authors model image degradation as convolution over the whole image with a single kernel, followed by downsampling. Given a LR image, they propose a method to estimate the kernel used to downsample the image solely from subpatches of the image by leveraging the self-similarity present in natural images. This is done by training a GAN, where the generator produces the kernel and the discriminator is trained to distinguish between crops of the original image and crops which are downsampled from original image using this estimated kernel. This method relies on the accuracy of the standard convolutional model of downsampling, which is shown to not hold for RealSR images in Figure 7.2. Further, the estimation of the kernel and subsequent SR are quite time consuming in comparison to supervised learning based methods; the calculation of the kernel alone for a 1000×1000 image can take more than three minutes on a GTX 1080 TI. In addition, their method constrains the size of the input images to be "large enough" since they need to downsample the input images during training. In [215], the authors propose an unsupervised cycle-in-cycle GAN, where they create one module for converting real LR images to denoised, deblurred LR images and one module for SR using these Clean LR images. They then tune these networks simultaneously in an end-to-end fashion, which causes this intermediate representation of the LR image to deviate from their initial objective.

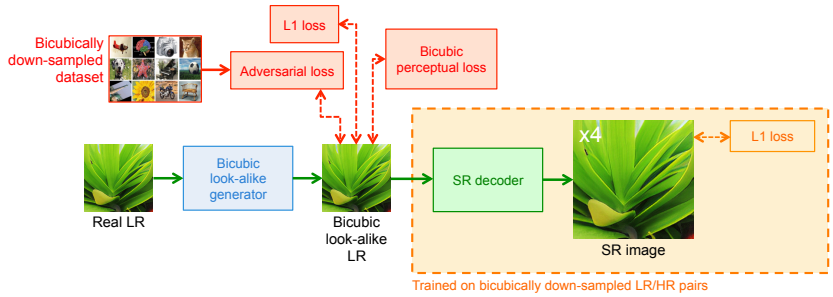


Figure 7.3: We propose a two-step pipeline for real world SR. First, we transform real LR images to bicubically downsampled looking images through our bicubic look-alike generator. We then pass the transformed image as input to a generic SR decoder trained on bicubically downsampled images.

7.3.2 Real World SR through extended models

In [210], the authors extend the bicubic degradation model by modeling image degradation as a convolution with an arbitrary blur kernel, followed by bicubic downsampling. They explicitly embed this unrealistic super-resolution model into an alternating iterative scheme where analytical deblurring is alternated with applying a SR network trained on bicubically downsampled images. Although this method generalizes to arbitrary kernels, one has to provide the kernel and the number of iterations as an input to the pipeline. In [209], the authors extend the bicubic degradation model by modeling image degradation as a convolution with a Gaussian blur kernel, followed by bicubic downsampling. They use an iterative scheme using only neural networks, where at each iteration the pipeline produces both the SR image and an estimate of the corresponding downsampling kernel. In [208], the authors also model

image degradations as convolution with a blur kernel followed by bicubic downsampling. They estimate the blur kernel using a pre-existing blind deblurring method on a set of "real" images which are bicubically upsampled; they use the same dataset of low quality cell-phone pictures used in [204]. They then train a GAN to generate new, realistic blur kernels using the blindly estimated blur kernels. Finally, they generate a large synthetic dataset using these kernels and train an end-to-end network on this dataset to perform SR. These three methods all rely on an analytical model for image degradation as well as being reliant on restrictive kernels or blind kernel estimation.

7.4 Methodology

7.4.1 Overall pipeline

RBSR consists of two steps; first, we use a Convolutional Neural Network (CNN)-based network, namely the bicubic look-alike image generator, whose objective is to take as input the real LR image and transform it into an image of the same size and content, but which looks as if it had been downsampled bicubically rather than with a realistic degradation. We call this output the bicubic look-alike image. Second, we use any generic SR network trained on bicubically downsampled data to take as input the transformed LR image and output the SR image. Figure 7.3 shows an overview of our proposed pipeline. We restrict the upsampling factor to four. In the following subsections, we describe each component of our pipeline in more detail.

7.4.2 Bicubic look-alike image generator

The bicubic look-alike image generator is a CNN, trained in a supervised manner. The main objective of this network is to transform real LR images to bicubic look-alike images. In this section, we present its architecture in detail. Then, we introduce a novel perceptual loss used to train it. Finally, we also introduce a novel copying mechanism used during training to make this transformation consistent.

Architecture

The architecture of the bicubic look-alike generator is shown in Figure 7.4. The generator is a feed-forward CNN, consisting of convolutional layers and several residual blocks, which has shown great capability in image-to-image translation tasks [216]. The real low-resolution image $I^{Real-LR}$ is passed through the first convolutional layer with a ReLU activation function with a 64 channel output. This output is subsequently passed through 8 residual blocks. Each block has two convolutional layers with 3×3 filters and 64 channel feature maps. Each one is followed by a ReLU activation. By using a long skip connection, the output of the final residual block is concatenated with the features of the first convolutional layer. Finally, the result is filtered by a last convolution layer to get the the 3-channel bicubic look-alike image

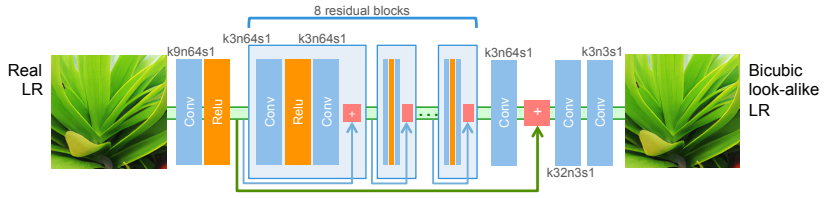


Figure 7.4: Schematic diagram of the bicubic-alike decoder. We train the decoder using our new bicubic perceptual loss, alongside standard L_1 and adversarial losses. In this schema, k, n and s correspond to kernel size, number of feature maps and stride size, respectively.

$(I^{Bicubic-LR})$.

Loss functions

In the bicubic look-alike generator, we use a loss function ($\mathscr{L}_{total}$) composed of three terms: 1- Pixel-wise loss ($\mathscr{L}_{pix.wise}$), 2- adversarial loss, and 3- our novel bicubic perceptual loss function ($\mathscr{L}_{bic.perc.}$). The overall loss function is given by:

$$\mathscr{L}_{total} = \alpha\mathscr{L}_{pix.wise} + \beta\mathscr{L}_{bic.perc.} + \gamma\mathscr{L}_{adv} \quad , \tag{7.2}$$

where α, β and γ are the corresponding weights of each loss term used to train our network. In the following, we present each term in detail:

• **Pixel-wise loss.** We use the L_1 norm of the difference between predicted and ground-truth images as this has been shown to improve results compared to the L_2 loss [217].

• **Adversarial loss.** This loss measures how well the image generator can fool a separate discriminator network, which originally was proposed to reconstruct more realistic looking images for different image generation tasks [218]–[221]. However, in our approach, as we are feeding the discriminator with bicubically downsampled images as the "real data", it results in images which are indistinguishable from bicubically downsampled images. The discriminator network used to calculate the adversarial loss is similar to the one presented in [219]; it consists of a series of convolutional layers with the number of channels of the feature maps of each successive layer increasing by a factor of two from that of the previous layer, up to 512 feature maps. The result is then passed through two dense layers, and finally, by a sigmoid activation function. The discriminator classifies the images as either "bicubically downsampled image" (real) or "generated image"(fake).

• **Bicubic perceptual loss.** Perceptual loss functions [86], [222] tackle the problem of blurred textures caused by optimization of using per-pixel loss functions and generally result in more photo-realistic reconstructions. In our approach, we take inspiration from this idea of perceptual similarity by introducing a novel perceptual loss.

However, instead of using a pre-trained classification network, e.g. VGG [223] for the high-level

feature representation, we use a pre-trained SR network trained on bicubically down-sampled LR/HR pairs. In particular, we use the output of the last residual block of our SR network, presented in Section 7.4.3, to map both HR and SR images into a feature space and calculate their distances. The bicubic perceptual loss term is formulated as:

$$\mathscr{L}_{bic.\_perc.} = \frac{1}{W_{i,j}\,H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \Big(\phi_k^{SR}\left(I^{Bicubic-LR}\right) - \phi_k^{SR}\left(I^{T-LR}\right) \Big)^2, \tag{7.3}$$

where $W_{i,j}$ and $H_{i,j}$ denote the dimensions of the respective feature maps. ϕ_k^{SR} indicates the output feature map of the k-th residual block from the SR decoder and I^{T-LR} denotes the transformed LR image. We conjecture that using a SR feature extractor, which is specifically trained for SR task and on bicubically down-sampled images, will better reflect features corresponding to the characteristics of bicubically downsampled images than using a feature extractor trained for image classification.

In Figure 7.5, we compare the effect of using the standard perceptual loss which uses a pre-trained classification network versus our bicubic perceptual loss. Note that the standard perceptual loss introduces artifacts in the transformed LR image which are avoided by the bicubic perceptual loss. Further, we see that using the bicubic perceptual loss produces sharper edges as compared to using just the L_1 loss.

Copying mechanism

Bicubically downsampled images are in general seem to be much sharper than realistic low-resolution images, therefore, training the bicubic look-alike network with only real LR images gives it a tendency to merely sharpen the input images instead of learning bicubic characteristics. To address this issue, we want the network to be consistent and apply minimal sharpening to already sharp images. To that end, we utilize a novel copying mechanism, where the network is periodically fed with identical, bicubically downsampled images as both input and output during training. This is done in order to prevent the network from just learning to sharpen images, as this can cause oversharpening or amplification of artifacts.

In Figure 7.6 we compare the outputs of the network trained with and without the copying mechanism. We can see clearly that training without the copying mechanism results in severe over-sharpening of the output image.

7.4.3 SR generator

The second step of our pipeline is to feed the output of our bicubic-like image generator as the input to any SR network trained on bicubically downsampled images. For simplicity, we use a network based on EDSR [224]. The EDSR architecture is composed of a series of residual blocks bookended by convolutional layers. Crucially, batch normalization layers are removed from these blocks for computational efficiency and artifact reduction. For simplicity, as well as decreasing training/inference time, we only use 16 residual blocks, as compared to the 32 residual blocks used in EDSR. This generator is trained on DIV2K training images (track 1: bicubically downsampled images and HR pairs) and by using the L_1 loss function.

7.4.4 Training parameters

Bicubic look-alike generator For the training data, as input, we use 400 RealSR [195] and 400 DIV2K Track 2 [196] LR images. The RealSR dataset contains real LR-HR pairs, captured by adjusting the focal length of a camera and taking pictures from the same scene. Track 2 images are downsampled using unknown kernels. As the desired output is the bicubic look-alike image, we use the bicubically downsampled RealSR and the bicubically downsampled DIV2K (track 1) images as the ground truth for the training inputs. In addition, as described in Section 7.4.2, we add 400 bicubically downsampled images from DIV2K, identical for both input and ground-truth, to make the generator consistent and avoid oversharpening or artifact amplification. We use the same 400 bicubically downsampled images from DIV2K as the real input of the discriminator. At each epoch, we randomly cropped the training images into 128×128 patches. The mini-batch size in all the experiments was set to 16. The training was done in two steps; first, the SR decoder was pre-trained for 1000 epochs with only the L_1 pixel-wise loss function. Then the proposed bicubic perceptual loss function, as well as the adversarial loss, were added and the training continued for 3000 more epochs. The weights of the L_1 loss, bicubic perceptual loss and adversarial loss function (α, β and γ) were set to 1.0, 3.0, and 1.0 respectively. The Adam optimizer [81] was used during both steps. The learning rate was set to 1×10^{-4} and then decayed by a factor of 10 every 800 epochs. We also alternately optimized the discriminator with similar parameters to those proposed by [219].

SR generator The SR decoder is also trained in a single step for 4000 epochs and using the L_1 loss function. For the training data, we only use track 1 images of DIV2K, which consists of 800 pairs of bicubically downsampled LR and HR images. Similar to the training of the bicubic look-alike generator, the Adam optimizer was used for the optimization process. The learning rate was set to 1×10^{-3} and then decayed by a factor of 10 every 1000 epochs.

End-to-end baseline To investigate the effectiveness of RBSR, which super-resolves a given input in two steps, we also fine-tune the EDSR architecture with the same datasets used to train the bicubic look-alike generator. This dataset consists of 400 RealSR and 400 DIV2K Track 2 LR and HR pairs. We further noticed that the inclusion of 400 bicubically downsampled LR and HR pairs in this dataset adds more robustness to the performance. In order to keep

the same number of parameters as in the RBSR pipeline, we increase the number of residual blocks of this end-to-end generator to 24. The training parameters used for this baseline is similar to the ones used in [224].

7.5 Experimental results

In this section, we compare RBSR to several SOTA algorithms (CVPR 2019, ICCV 2019) in real-world SR both qualitatively and quantitatively. We show standard distortion metrics for the datasets with ground truth, and we show a comprehensive user study conducted over six image datasets with varying image quality and degradations. In all cases, we use an upsampling factor of four.

We emphasize that the distortion metrics are not directly correlated to the perceptual quality as judged by human raters [219], [226]–[230]; the super-resolved images could have higher errors in terms of the PSNR and SSIM metrics, but still generate more appealing images. Moreover, the RealSR images represent only a limited group of realistic images from Nikon and Canon cameras. Therefore, we validate the effectiveness of our approach by qualitative comparisons and by an extensive user study in the following sections.

7.5.1 Test images

Lack of ground-truth in real-world SR

One of the main challenges of real-world SR is the lack of real low and high resolutions pairs, for both training and testing. As mentioned previously, most of the known benchmarks in super-resolution had no choice but using a known kernel to create a counterpart with lower resolution. To the best of our knowledge RealSR [195] is the only dataset with real images of the same scenes with different resolutions: their LR and HR images are generated by taking two camera pictures of the same scene, but changing the focal length of the camera between the two pictures. Hence, both are real images, but with the RealSR LR being degraded with the degradation from changing the focal length of the camera (zooming out). DIV2K Unknown kernel LR images [196] is another attempt to create pairs of real low and high-resolutions images. They generate synthetically real low and high resolution images by using unknown/random degradation operators.

Images without ground-truth

In addition to RealSR LR and DIV2K Unknown kernel datasets, we also evaluate our method on four datasets of real images, without having any ground-truth as this is the main focus of real-world SR: 1- RealSR [195] HR test images, 2- DIV2K HR [196] validation images (real), 3- DPED [225] Mobile Phone images, 4- TV Stream images (unknown, depending on the original content of the TV). The DPED Mobile Phone dataset is a dataset of real images where cell-phones were used to take pictures. The TV stream images are decoded images from an

actual TV channel stream at HD (1920 × 1080) resolution; our acquisition algorithm captured one image every ten minutes over a period of two days, to ensure that our these test images cover different types of content. We note that no information is available about their type of degradations, as the original resolutions of the contents before streaming are unknown. Further, we note that we only have the ground-truth high-resolution images for the DIV2K Unknown Kernels images and the RealSR LR images.

7.5.2 Quantitative results

In this work, calculating distortion metrics such as PSNR and SSIM is not possible for test images that truly reflect the real-world problem (original images from smartphones, TV streams, etc.), as in real cases the downsampling operator is not known and therefore no ground-truth is available. RealSR [195] is the only dataset with physically produced high and low-resolution image pairs.

Table 7.1 shows the SSIM and PSNR values estimated between super-resolved images of RealSR LR test images and their HR counterparts, using bicubic upsampling, EDSR-real [224], the RealSR network [195], DPSR [210] and our proposed method. The training details of each method is presented in Section 4.3 of the main manuscript. We also add the perception index (PI) metric to our evaluation; this index combines two no-reference image quality measures of Ma et al. [231] and NIQE [232] and was shown to have a higher correlation with human opinion than other commonly used metrics [226]. As PI is a no-reference metric, it can be also used for test images that have no ground-truth.

Dataset	Method	bicubic	SRResNet	RCAN	EDSR-real	DPSR	RealSR	**RBSR**
RealSR	SSIM	0.77	0.79	0.80	0.81	0.79	0.81	**0.82**
	PSNR	26.63	26.98	27.11	26.51	27.02	**28.05**	26.54
	PI	9.28	9.06	9.19	7.94	9.12	8.97	**7.76**
DIV2K HR	SSIM/PSNR	-	-	-	no ground-truth	-	-	-
	PI	10.02	9.62	9.81	9.01	9.36	9.19	**8.48**
DPED (cellphones)	SSIM/PSNR	-	-	-	no ground-truth	-	-	-
	PI	10.24	9.91	10.02	9.62	9.73	9.55	**7.92**
TV Streams	SSIM/PSNR	-	-	-	no ground-truth	-	-	-
	PI	11.52	10.71	10.64	**10.04**	11.19	10.32	10.15

Table 7.1: Comparison of bicubic interpolation, SRResNet [219], RCAN [233], EDSR [224], DPSR [210], RealSR [195] and RBSR (ours) on different presented test sets. Best measures (SSIM ↑, PSNR [dB] ↑, PI ↓) are highlighted in bold.

7.5.3 Qualitative comparison

For the qualitative comparison, we compare the following real world SR algorithms: 1- RBSR (Ours), 2- EDSR-real: the EDSR [224] network trained end-to-end on the same data/settings as RBSR, 3- The pretrained RealSR network [195], and 4- The pre-trained DPSR network with default settings for real-world SR [210]. We compare with the end-to-end EDSR network in order to show the efficacy of splitting the problem into two steps. We compare to RealSR and DPSR as they are two of the most recent state-of-the-art algorithms. We use their pre-trained models along with the default settings for real images they provide[I,II]. In Figure 7.7, we show qualitative results on a random subset of the image datasets described in the previous sections.

7.5.4 User study

We also conducted a user study comprising forty one people in order to gauge the perceptual image quality of SR images using the image datasets described in the previous section. We chose five images randomly from each dataset, with thirty total images. For each image, the users were shown four SR versions of the image, each corresponding to the real-world SR algorithms being compared. Users were asked to select which SR image felt more realistic and appealing. The images were shown to users in a randomized manner.

Figure 7.9 shows a screenshot of the survey that we used to evaluate our proposed method.

We note that no reference image was shown, since the vast majority of the images had no ground truth. In sum, 41 people participated in user study.

As the datasets reflect a wide range of image quality, etc., we show the evaluations of the algorithms for each dataset separately. Our metric of evaluation for the algorithms is the percent of votes won. We show the results of the user study in Figure 7.8. We find that RBSR won the largest percent of votes over all six image datasets individually. RBSR decisively won the largest percentage of votes, by a margin of 10 to 55% from the second ranked algorithm, on the DIV2K HR, the RealSR-HR, the RealSR-LR, and the TV stream image datasets. The second place algorithm on these datasets alternated from RealSR, DPSR, and EDSR-Real, and RealSR respectively. We note that on the RealSR-LR dataset, for which the RealSR algorithm is tailored and trained, RBSR and EDSR-Real are the first and second place. This shows the efficacy of both the two step approach of RBSR and introducing bicubically downsampled images into the training dataset. On the DPED dataset, RBSR won by a small margin over DPSR.

[I] https://github.com/csjcai/RealSR
[II] https://github.com/cszn/DPSR

7.6 Additional Experiments

7.6.1 Generalizability of RBSR

Our proposed approach (RBSR) is a two step procedure. The first step transforms the real LR image using the bicubic look-alike generator. The second step uses any generic SR decoder trained on bicubically downsampled images, taking the transformed LR image as input. For the qualitative comparison and the user study, we used a pre-trained EDSR network for this second step. Here, we show the robustness and generalizability of our two step approach by replacing the EDSR network with pretrained ESRGAN and RCAN models. To do so, we compare the results of these models on real LR images and our transformed LR images obtained from the bicubic look-alike generator. Experimental results demonstrate that these SR methods generate more plausible results with greater perceptual quality when fed with transformed LR images instead of real LR images (see Figure 7.10).

7.6.2 Ablation study

In this section, we perform another study to investigate the effectiveness of each proposed component of the bicubic look-alike generator. We compare the performance of our network trained with the combinations of different settings such as different loss functions, and trainings with and without copying mechanism. These setting are listed in Table 7.2. We calculate PSNR and SSIM for each setting on RealSR [195] test set, the only available dataset with ground-truth for real-world SR task. For each setting, SSIM and PSNR values are calculated after upsampling the picture by a fixed $\times 4$ SR decoder and comparing it to the RealSR ground-truth.

Name	Description	SSIM	PSNR
$RBSR_{MSE}$	only $\mathscr{L}_{MSE}$ loss	0.788	27.69
$RBSR_{E}$	only $\mathscr{L}_1$ loss	0.792	**27.95**
$RBSR_{EP}$	$\mathscr{L}_1 + \mathscr{L}_{perceptual}$	0.811	26.98
$RBSR_{EPA}$	$\mathscr{L}_1 + \mathscr{L}_{perceptual} + \mathscr{L}_{adversarial}$	0.798	26.60
$RBSR_{EBA}$	$\mathscr{L}_1 + \mathscr{L}_{bicubic\,perceptual} + \mathscr{L}_{adversarial}$	**0.835**	26.73
$RBSR$	$\mathscr{L}_1 + \mathscr{L}_{bicubic\,perceptual} + \mathscr{L}_{adversarial}$ + Copying mechanism	0.820	26.54

Table 7.2: Comparing the effect of each proposed component of the bicubic look-alike generator on LR and HR images of [195] test set. Best measures (SSIM ↑, PSNR [dB] ↑) are highlighted in bold. As mentioned earlier, **these metrics are not directly correlated to the perceptual quality, therefore, we chose our best baseline based on qualitative comparison shown in Figure 5 and Figure 6 of the manuscript, comparing $RBSR_{EPA}$ to $RBSR_{EBA}$ and $RBSR_{EBA}$ to $RBSR$, respectively.**

As already emphasized, the distortion metrics are not directly correlated to the perceptual

quality as judged by human raters, therefore, we chose our best baseline based on qualitative comparisons.

7.6.3 Computational cost

We compared our two step approach (RBSR), our end-to-end comparison (EDSR-real), RealSR [195], and DPSR [210]. In terms of computational cost, both RealSR and DPSR have different disadvantages. RealSR's network calculations take place in the high-resolution space, incurring a heavy memory overhead cost. For example, running the model on CPU requires 19 GB of RAM for an image of size 1200×1200, which is the maximum possible. DPSR is an iterative algorithm, requiring multiple forward passes and multiple deblurring steps in order to converge to an acceptable solution; DPSR uses an iterative approach by default for real LR images. Hence, these two algorithms have either high memory overhead or high computation time overhead. In contrast, RBSR requires two forward passes per input image. The first network is relatively lightweight, as it operates exclusively in the LR space. The second network can be any generic SR decoder for bicubically downsampled images. The complete pipeline (using EDSR as the SR decoder) reconstructs 1024×768 pixel images at 26.9 FPS, using a GeForce GTX 1080 Ti. Our end-to-end setting (EDSR-real) reconstructs the same size images at 33.7 FPS using the same GPU.

7.7 Conclusion

In this chapter, we have shown that the challenges of super resolution on realistic images can be partly alleviated by decomposing the typical SR inverse problem solution into two sub-problems. First, is the conversion of real LR images to bicubic look-alike images using our novel copying mechanism and bicubic perceptual loss. Second, is the super-resolution of the converted images using any generic network trained on bicubically downsampled images. Each sub-problem addresses a different aspect of the real-world SR problem. Converting real low-resolution images to bicubic look-alike images allows us to handle and model the variety of realistic image degradations. The super-resolution of bicubically downsampled images allows for the application of state-of-the-art super-resolution models, which have achieved impressive results on images with well defined degradations. In this way, we can leverage both the latest advances in model-based approaches, with realistic albeit small training datasets, and the latest advances in data-driven approaches, which achieve incredible results on large, albeit unrealistic training datasets. We show that our approach (RBSR) outperforms the SOTA in real-world SR both qualitatively and quantitatively using a comprehensive user study over a variety of real image datasets. In particular, we can see that RBSR, which implicitly embeds a realistic, albeit analytically challenging SR model, is able to beat DPSR, a supervised approach which explicitly embeds a conventional albeit unrealistic SR model in its iterative solution.

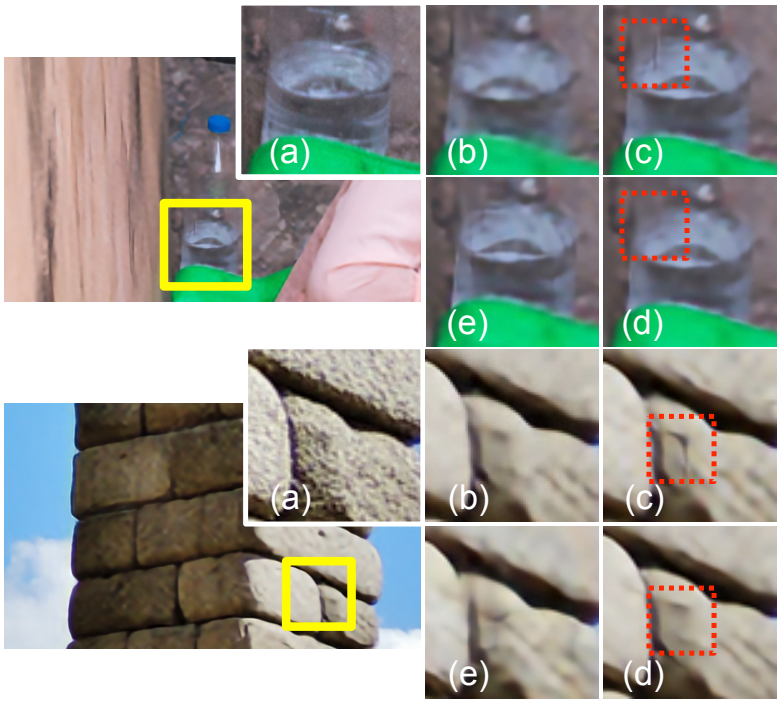


Figure 7.5: The effectiveness of using bicubic perceptual loss: (a) HR image, (b) Only L1 loss, (c) perceptual loss, (d) bicubic perceptual loss, and (e) bicubic perceptual loss + adversarial loss. Red boxes show how using bicubic perceptual loss (c) decreases artifacts comparing to using conventional perceptual losses (d), while still producing sharper edges comparing to only using L_1 loss.

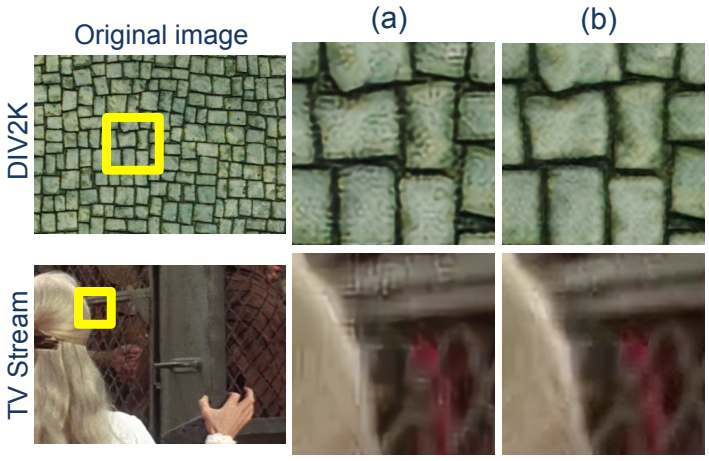


Figure 7.6: Example images generated without (a) and with (b) the copying mechanism during training. We can clearly see that without the copying mechanism, resulting images suffer from oversharpening and artifact amplification.

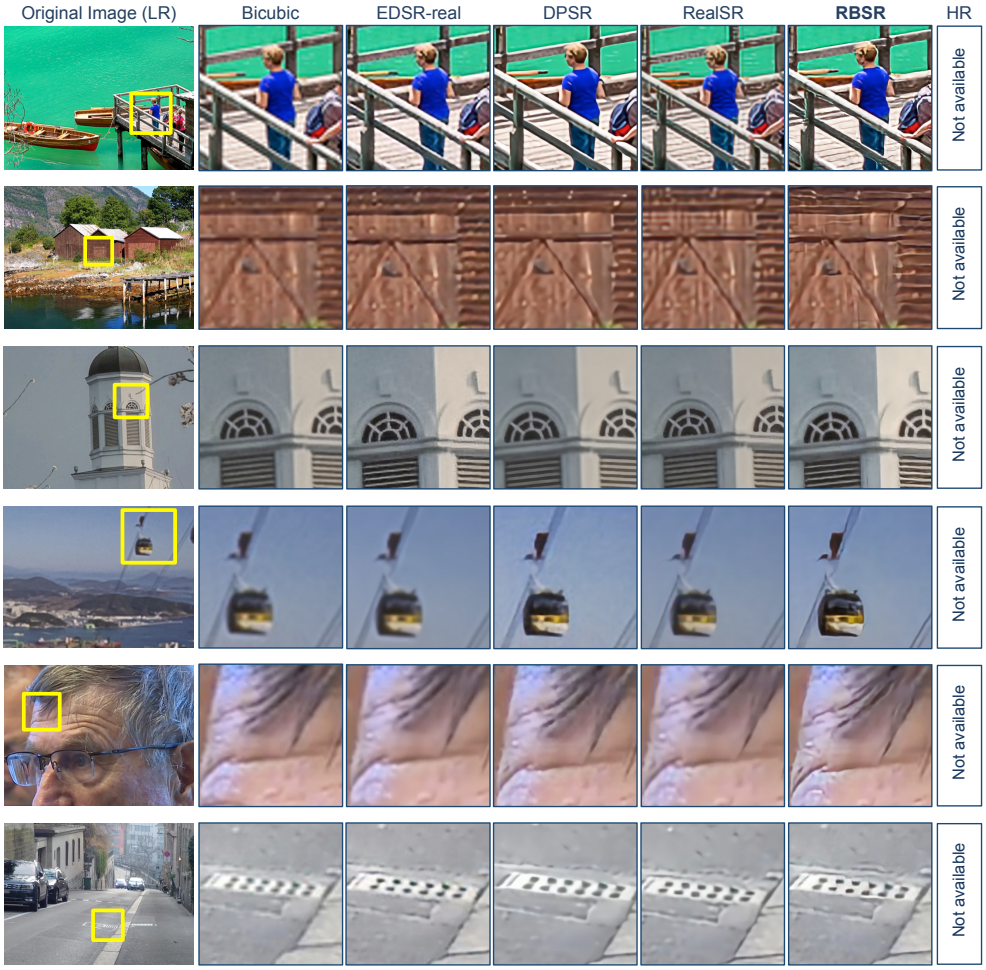


Figure 7.7: Qualitative results of ×4 SR on a subset of the DIV2k [196] (Rows 1-2), RealSR HR [195] (Rows 3-4), TV Streams (Row 5), and DPED cell-phone images [225] (Row 6). Results from left to right: bicubic, EDSR [224] fine-tuned with real LR and HR pairs, DPSR [210], RealSR [195], and RBSR (ours). Please note that no ground-truth is available for these images. **Zoom in for the best view.**

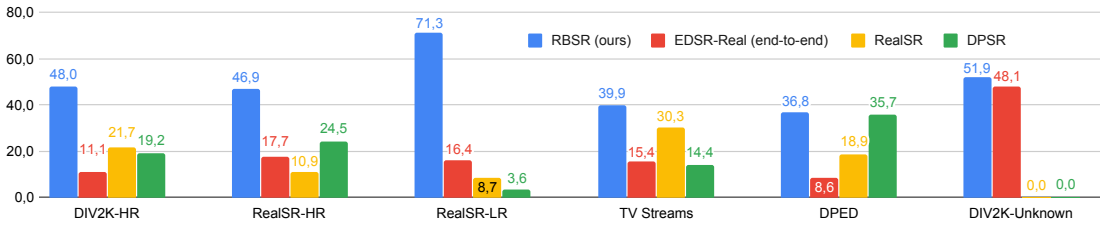


Figure 7.8: Results of the user study comprising forty one people, comparing EDSR [224], fine-tuned with real LR and HR pairs, DPSR [210], RealSR [195], and RBSR (ours), on six different datasets: DIV2K HR [196], RealSR [195] HR, RealSR LR, TV Stream images, DPED [225] Mobile Phone images, and DIV2K Unknown Kernel LR.

Figure 7.9: Example screenshot of our online survey to perform a user study and compare our method to state-of-the-art real-world SR approaches. In total, 41 people participated in this survey.

(a) RCAN [233]

(b) ESRGAN [234]

Figure 7.10: Comparison results of RCAN (a) and ESRGAN (2) methods on original images from the RealSR dataset and our transformed LR images, generated by our bicubic look-alike generator (BLG). Experimental results demonstrate that these SR methods generate more plausible results with greater perceptual quality when fed with transformed LR images instead of real LR images.

8 Discussion and Conclusion

8.1 General Discussion

In the preceding chapters, we concluded each section with a discussion specific to the inverse problem considered. Here, we make a few general observations.

8.1.1 Self-Supervised vs. Supervised Approaches

We note that the self-supervised methods of Chapters 4 and 5 are independent of the specific model M of the inverse problem, though of course their success greatly relies on being embedded in traditional, model-driven schemes (Hamiltonian MCMC and variational optimization); being more flexible, these methods can applied to all inverse problems, for example, where noise plays a significant role. In contrast, the realistic modelling approaches in Chapters 6 and 7 are entirely specific to the model M of the inverse problem by design; furthermore, for these approaches we used standard supervised learning methods. Therefore, the introduction of a self-supervised component along with realistic modelling could further enhance performance; for example, combining the self-supervised noise reduction of Noise2Self with supervised super-resolution models.

8.1.2 Validation

While we focused exclusively on validation in Chapter 4, validation is a key component/concern in all the inverse problems we considered, since, in all cases, there is little to no real ground truth data. Other than Chapter 5, where we only validated on synthetic data, the validations heavily relied on qualitative analyses through visual inspection and quality ratings **in the absence of ground truth**. Therefore, in some sense, these analyses more consider the plausibility, consistency, and conformity to expected content from prior knowledge of the reconstructions. In the case of real-world super-resolution, this is entirely appropriate as one of the primary applications of real-world super-resolution is simply to produce images which ”look good”, e.g. for television or personal use; extreme fidelity to the ground truth is not

critical for this purpose. In contrast, fidelity to the ground truth is critical for medical images since the purpose of the image is not merely to look plausible, but to reflect the actual physical state of the imaged region since the purpose is medical diagnosis and research. Hence, while qualitative analysis, particularly by radiologists and MR experts, can be a reasonable proxy for physical validation, it should be taken with a grain of salt. Furthermore, even when ground truth data is available, making quantitative analysis possible, we showed in Chapters 4 that commonly used quantitative metrics (PSNR, SSIM, MSE) can be misleading, with counter-intuitive results which have little correlation with qualitative analysis **where ground truth is available**. While we showed alternative quantitative metrics that yield more intuitive results (e.g. Wasserstein distance for probability distributions, perceptual distance for images), these have to be carefully tailored to the problem at hand. Therefore, while fidelity to the ground truth is important, establishing universal, robust ways in which fidelity is measured in inverse problems is still an open problem.

8.1.3 Properties of Measurement Data and Solutions

Finally, we recall the list of fundamental questions (2.1.1) introduced in the introductory chapter, applying to all inverse problems. Throughout the thesis, we have addressed the questions of how realistic the model is, computational expense, the effect of noise in the measurements, and the myriad ways to solve inverse problems. However, two questions in particular remain outstanding: the theoretical number of measurements/amount of data needed for successful recovery and theoretical limits on what solutions can be recovered.

Data Requirements

With model-driven methods, such as compressed sensing, there are proven, theoretical requirements of the measurement data in order for successful recovery; the theory of compressed sensing proves that sparse signals can be reconstructed, with high probability, from a small set of **random** measurements (small in comparison to the Nyquist-Shannon limit). Hence, the measurement data is required to be randomly sampled. However, it is hard, in general, to theoretically characterize requirements of the measurement data. For example, in undersampled MRI, suppose the goal is to recover spatial features of a certain size; e.g. if one wants to detect small tumors or lesions in a brain scan. Then there is clearly an upper bound on the acceleration factor; at some point there is simply not enough information in the measured data to recover this information. However, other than through experiment, it is difficult to see how to establish a sharp bound. This is important as with machine learning algorithms, if the acceleration factor is "too high", one runs the risk of producing a plausible image with no indication that small features have been omitted. There is a similar problem in multi-component T_2 relaxometry: there is clearly a lower bound on the number of echos necessary to resolve a lobe of a specific size. In the respective chapters, we fixed the acceleration factor and number of echos; however, it would be desirable to have some theoretical estimates of the number of measurements required to solve the inverse problem to a given

specification. Note that we would expect looser requirements on machine learning methods since they can take advantage of learning from training data; however, as previously stated, this runs the concomitant risk of simply outputting an image from the implicit prior of the training data when there is insufficient measurement data.

Properties of Solution Methods

With model-driven methods based on the variational framework of Chapter 2, there are usually theoretical guarantees on the convergence of the optimization algorithms; while these **do not** guarantee that the method converges to the correct solution, they are still desirable as they show that solutions, unique or otherwise, exist and will be reached. In contrast, data-driven methods, while showing impressive empirical performance, generally have comparatively less theoretical guarantees, particularly for end-to-end supervised methods that implicitly embed M. Indeed, other than the neural network sampler of Chapter 5 and DeepDecoder of Chapter 4, which can be shown to satisfy theoretical guarantees on the validity as a MCMC sampler and convergence for compressed sensing respectively, none of the other data-driven/hybrid methods proposed or examined enjoy such theoretical guarantees. In Chapter 2, we noted that theoretical analysis of data-driven methods exists and is gaining popularity, but is usually specific to a network architecture/learning strategy. In the future, it would be desirable to have more general theoretical analyses of data-driven methods.

8.2 Future Work and Conclusion

- In Chapter 4, we engaged in a rigorous validation of self-supervised methods for image reconstruction from undersampled MR measurements, where, in contrast to the previous literature, we focused primarily on prospectively accelerated data, the clinically relevant scenario. Encouragingly, we found that self-supervised methods have high potential for generalization, as well as evidence that no-reference image quality metrics could be useful as quantitative metrics when no ground truth is available. However, we also found that commonly used quantitative metrics do not necessarily reflect perceptual quality. Furthermore, we confirmed that retrospective reconstruction quality cannot necessarily be taken as a reliable proxy for prospective reconstruction quality. For the future, more work should be done to investigate quantitative metrics which reflect perceptual quality. Furthermore, as the end goal of the images is for diagnosis, more work should be done to add standardize validation over, for example, the detection of pathological features in the images.

- In Chapter 5, we proposed to solve inverse problems in a probabilistic framework using a neural network enhanced MCMC sampler which is trained in a self-supervised way. We found that our proposed method was more robust and accurate than the state of the art in solving a difficult joint diffusometry-relaxometry problem. For the future, more work should be done to accelerate the MCMC sampling by optimizing the burnout

period/number of samples and potentially modifying the algorithm in a more GPU friendly way. Furthermore, our method, in conjunction with realistic modelling, could be used to solve the joint diffusometry-relaxometry problem in a more well-posed way by adding additional constraints.

- In Chapter 6, we created a large, realistic dataset for training a multi-layer perceptron to solve the inverse problem of multi-component T_2 relaxometry, using realistic signal modelling and realistic priors on the solutions. Furthermore, we introduced a novel loss function tailored for probability distributions. We demonstrated that our method was more robust, accurate, and orders of magnitude faster than the state of the art. For the future, more work should be done to incorporate additional physical modelling into the dataset generation, to make the modelling even more realistic. Furthermore, there is potential to combine different types of acquisitions to simultaneously solve for the spectrum along multiple parameters (T_2, T_1, Diffusion, Magnetization transfer, etc) as more realistic modelling needs to incorporate effects on the signal from these parameters.

- In Chapter 7, we proposed a two step pipeline for real-world super resolution, where we first convert real low-resolution images to a bicubically downsampled version using a neural network, then we put this as input to any generic SR network which has been trained on bicubically downsampled images. This allowed us to leverage both a small realistic dataset as well as much larger, unrealistic datasets for solving the super-resolution inverse problem, while also being able to take advantage of existing SR approaches without any modifications. For the future, more work should be done to expand the realistic dataset with additional realistic downsampling, as we concentrated mainly on focal length changes. Furthermore, the validation of real-world super resolution algorithms is still heavily based on user perception of image quality, as no ground truth is usually available. While quantitative validation using unrealistic datasets with ground truth is still an option, there is no guarantee that image quality will be robust to realistic inputs or that commonly used quantitative metrics will correspond to perceptual quality. More work should be done to standardize the validation of real-world super resolution algorithms such that it does not have to heavily rely on different subjective judgments from different people or quantitative evaluation on unrealistic datasets.

A naive observer first introduced to machine learning and the deep learning revolution might remark on the declining usefulness of modelling, analytical solutions, and traditional algorithms; after all, one can simply acquire a huge dataset of training pairs and train a neural network which outperforms traditional approaches on every metric without having to deal with these complexities. To some extent, this is true for some subjects, such as image classification. However, when faced with complex problems for which no or little ground truth data is available or even feasible to acquire, one must use all the tools at one's disposal, model-driven and data-driven. In this thesis, we have addressed four inverse problems, arising from MRI and computer vision, for which large, realistic datasets for training standard, supervised machine

learning methods are unavailable, with concomitant constraints and limitations on their solution and validation. We showcased two broad strategies for dealing with this lack of data: self-supervised learning and the application of realistic modelling such that traditional supervised methods can be trained with large, realistic datasets which are synthetically generated or large, unrealistic datasets augmented by a comparatively small quantity of realistic, physically acquired data. In doing so, we showed that model-driven and data-driven methods can be combined in a variety of different ways to robustly and accurately solve inverse problems, though the details and structure of the combination depend heavily on the specific of the inverse problem addressed.

Bibliography

[1] M. Bertero and P. Boccacci, *Introduction to inverse problems in imaging*. CRC press, 2020.

[2] C. R. Vogel, *Computational methods for inverse problems*. SIAM, 2002.

[3] R. W. Brown, Y.-C. N. Cheng, E. M. Haacke, M. R. Thompson, and R. Venkatesan, *Magnetic resonance imaging: physical principles and sequence design*. John Wiley & Sons, 2014.

[4] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, "Deep learning techniques for inverse problems in imaging", *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 39–56, 2020.

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

[6] C. Clason, *Regularization of inverse problems*, 2021. arXiv: 2001.00617 [math.FA].

[7] M. Benning and M. Burger, *Modern regularization methods for inverse problems*, 2018. arXiv: 1801.09922 [math.NA].

[8] M. Kac, "Can one hear the shape of a drum?", *The american mathematical monthly*, vol. 73, no. 4P2, pp. 1–23, 1966.

[9] C. Gordon, D. L. Webb, and S. Wolpert, "One cannot hear the shape of a drum", *Bulletin of the American Mathematical Society*, vol. 27, no. 1, pp. 134–138, 1992.

[10] K. Pepper. "Isospectral shapes". (Aug. 2007), [Online]. Available: https://commons.wikimedia.org/wiki/File:Isospectral_drums.svg.

[11] H. Weyl, "Über die asymptotische verteilung der eigenwerte", *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, vol. 1911, pp. 110–117, 1911.

[12] Z. Lu and J. M. Rowlett, "One can hear the corners of a drum", *Bulletin of the London Mathematical Society*, vol. 48, no. 1, pp. 85–93, 2016.

[13] J. Hadamard, "Sur les problèmes aux dérivées partielles et leur signification physique", *Princeton university bulletin*, pp. 49–52, 1902.

[14] A. N. Tihonov, "Solution of incorrectly formulated problems and the regularization method", *Soviet Math.*, vol. 4, pp. 1035–1038, 1963.

[15] A. N. Tikhonov, A. Goncharsky, V. Stepanov, and A. G. Yagola, *Numerical methods for the solution of ill-posed problems*. Springer Science & Business Media, 1995, vol. 328.

[16] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[17] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements", *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 59, no. 8, pp. 1207–1223, 2006.

[18] D. L. Donoho, "Compressed sensing", *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[19] C. E. Shannon, "Communication in the presence of noise", *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[20] M. R. Hestenes, E. Stiefel, *et al.*, *Methods of conjugate gradients for solving linear systems*, 1. National Bureau of Standards, Washington, DC, 1952, vol. 49.

[21] K. Levenberg, "A method for the solution of certain non-linear problems in least squares", *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[22] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[23] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting", *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[24] J. Douglas and H. H. Rachford, "On the numerical solution of heat conduction problems in two and three space variables", *Transactions of the American mathematical Society*, vol. 82, no. 2, pp. 421–439, 1956.

[25] P. L. Combettes and J.-C. Pesquet, *Proximal splitting methods in signal processing*, 2010. arXiv: 0912.3522 [math.OC].

[26] L. Condat, "A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms", *Journal of optimization theory and applications*, vol. 158, no. 2, pp. 460–479, 2013.

[27] J.-J. Moreau, "Proximité et dualité dans un espace hilbertien", *Bulletin de la Société mathématique de France*, vol. 93, pp. 273–299, 1965.

[28] V. A. Morozov, "The error principle in the solution of operational equations by the regularization method", *USSR Computational Mathematics and Mathematical Physics*, vol. 8, no. 2, pp. 63–87, 1968.

[29] P. C. Hansen, "Analysis of discrete ill-posed problems by means of the l-curve", *SIAM review*, vol. 34, no. 4, pp. 561–580, 1992.

[30] E. Soubies, F. Soulez, M. T. McCann, *et al.*, "Pocket guide to solve inverse problems with globalbioim", *Inverse Problems*, vol. 35, no. 10, p. 104 006, 2019.

[31] J. Adler, H. Kohr, A. Ringh, *et al.*, *Odlgroup/odl: odl 0.7.0*, version v0.7.0, Sep. 2018. DOI: 10.5281/zenodo.1442734. [Online]. Available: https://doi.org/10.5281/zenodo.1442734.

[32] M. SIMEONI and P. del Aguila Pla, *Matthieumeo/pycsou: pycsou 1.0.6*, version v1.0.6, Apr. 2021. DOI: 10.5281/zenodo.4715243. [Online]. Available: https://doi.org/10.5281/zenodo.4715243.

[33] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: residual learning of deep cnn for image denoising", *IEEE transactions on image processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[34] W. Ring, "Structural properties of solutions to total variation regularization problems", *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 34, no. 4, pp. 799–810, 2000.

[35] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation", *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 492–526, 2010.

[36] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms", *IEEE Transactions on Signal Processing*, vol. 61, no. 5, pp. 1072–1086, 2012.

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

[38] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain.", *Psychological review*, vol. 65, no. 6, p. 386, 1958.

[39] K. Fukushima and S. Miyake, "Neocognitron: a self-organizing neural network model for a mechanism of visual pattern recognition", in *Competition and cooperation in neural nets*, Springer, 1982, pp. 267–285.

[40] Y. LeCun, B. Boser, J. S. Denker, *et al.*, "Backpropagation applied to handwritten zip code recognition", *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[41] G. Cybenko, "Approximation by superpositions of a sigmoidal function", *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[42] K. Hornik, "Approximation capabilities of multilayer feedforward networks", *Neural networks*, vol. 4, no. 2, pp. 251–257, 1991.

[43] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: can plain neural networks compete with bm3d?", in *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, pp. 2392–2399.

[44] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks", *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

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

[46] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for mr image reconstruction", in *International Conference on Information Processing in Medical Imaging*, Springer, 2017, pp. 647–658.

[47] K. Hammernik, T. Klatzer, E. Kobler, *et al.*, "Learning a variational network for reconstruction of accelerated mri data", *Magnetic resonance in medicine*, vol. 79, no. 6, pp. 3055–3071, 2018.

[48] J. Sun, H. Li, Z. Xu, *et al.*, "Deep admm-net for compressive sensing mri", *Advances in neural information processing systems*, vol. 29, 2016.

[49] T. Meinhardt, M. Moller, C. Hazirbas, and D. Cremers, "Learning proximal operators: using denoising networks for regularizing inverse imaging problems", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1781–1790.

[50] J. Rick Chang, C.-L. Li, B. Poczos, B. Vijaya Kumar, and A. C. Sankaranarayanan, "One network to solve them all–solving linear inverse problems using deep projection models", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5888–5897.

[51] R. Liu, S. Cheng, Y. He, X. Fan, Z. Lin, and Z. Luo, "On the convergence of learning-based iterative methods for nonconvex inverse problems", *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 12, pp. 3027–3039, 2019.

[52] X. Chen, J. Liu, Z. Wang, and W. Yin, "Theoretical linear convergence of unfolded ista and its practical weights and thresholds", *arXiv preprint arXiv:1808.10038*, 2018.

[53] R. Liu, S. Cheng, L. Ma, X. Fan, and Z. Luo, "Deep proximal unrolling: algorithmic framework, convergence analysis and applications", *IEEE Transactions on Image Processing*, vol. 28, no. 10, pp. 5013–5026, 2019.

[54] W. Gerlach and O. Stern, "Der experimentelle nachweis der richtungsquantelung im magnetfeld", *Zeitschrift für Physik*, vol. 9, no. 1, pp. 349–352, 1922.

[55] ——, "Das magnetische moment des silberatoms", *Zeitschrift für Physik*, vol. 9, no. 1, pp. 353–355, 1922.

[56] I. I. Rabi, S. Millman, P. Kusch, and J. R. Zacharias, "The molecular beam resonance method for measuring nuclear magnetic moments. the magnetic moments of Li_3^6, Li_3^7, and F_9^{19}", *Physical review*, vol. 55, no. 6, p. 526, 1939.

[57] F. Bloch, W. W. Hansen, and M. Packard, "Nuclear induction", *Phys. Rev.*, vol. 69, pp. 127–127, 3-4 Feb. 1946. DOI: 10.1103/PhysRev.69.127. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.69.127.

[58] E. M. Purcell, H. C. Torrey, and R. V. Pound, "Resonance absorption by nuclear magnetic moments in a solid", *Phys. Rev.*, vol. 69, pp. 37–38, 1-2 Jan. 1946. DOI: 10.1103/PhysRev.69.37. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.69.37.

[59] P. C. Lauterbur, "Image formation by induced local interactions: examples employing nuclear magnetic resonance", *Nature*, vol. 242, no. 5394, pp. 190–191, 1973.

[60] F. Knoll, J. Zbontar, A. Sriram, *et al.*, "Fastmri: a publicly available raw k-space and dicom dataset of knee images for accelerated mr image reconstruction using machine learning", *Radiology: Artificial Intelligence*, vol. 2, no. 1, e190007, 2020.

[61] I. C. L. Biomedical Image Analysis Group. "Ixi dataset". (2006), [Online]. Available: https://brain-development.org/ixi-dataset/ (visited on 12/30/2021).

[62] T. Yu, T. Hilbert, G. F. Piredda, *et al.*, *Validation and generalizability of self-supervised image reconstruction methods for undersampled mri*, 2022. DOI: 10.48550/ARXIV.2201.12535. [Online]. Available: https://arxiv.org/abs/2201.12535.

[63] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, "Sense: sensitivity encoding for fast mri", *Magnetic Resonance in Medicine*, vol. 42, no. 5, pp. 952–962, 1999.

[64] M. A. Griswold, P. M. Jakob, R. M. Heidemann, *et al.*, "Generalized autocalibrating partially parallel acquisitions (grappa)", *Magnetic Resonance in Medicine*, vol. 47, no. 6, pp. 1202–1210, 2002.

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

[66] F. Knoll, K. Hammernik, C. Zhang, *et al.*, "Deep-learning methods for parallel magnetic resonance imaging reconstruction: a survey of the current approaches, trends, and issues", *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 128–140, 2020.

[67] K. Hammernik and F. Knoll, "Chapter 2 - machine learning for image reconstruction", in *Handbook of Medical Image Computing and Computer Assisted Intervention*, ser. The Elsevier and MICCAI Society Book Series, S. K. Zhou, D. Rueckert, and G. Fichtinger, Eds., Academic Press, 2020, pp. 25–64, ISBN: 978-0-12-816176-0. DOI: https://doi.org/10.1016/B978-0-12-816176-0.00007-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128161760000077.

[68] J. Liu, Y. Sun, C. Eldeniz, W. Gan, H. An, and U. S. Kamilov, "Rare: image reconstruction using deep priors learned without groundtruth", *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 6, pp. 1088–1099, 2020.

[69] B. Yaman, S. A. H. Hosseini, S. Moeller, J. Ellermann, K. Uğurbil, and M. Akçakaya, "Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data", *Magnetic Resonance in Medicine*, vol. 84, no. 6, pp. 3172–3191, 2020.

[70] R. Heckel and P. Hand, "Deep decoder: concise image representations from untrained non-convolutional networks", *Proceedings of the International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=rylV-2C9KQ.

[71] M. Akçakaya, B. Yaman, H. Chung, and J. C. Ye, "Unsupervised deep learning methods for biological image reconstruction", *arXiv preprint arXiv:2105.08040*, 2021.

[72] K. Epperson, A. Sawyer, M. Lustig, *et al.*, "Creation of fully sampled mr data repository for compressed sensing of the knee.", *Proceedings of Society for MR Radiographers and Technologists, 22nd Annual Meeting. Salt Lake City, Utah, USA.*, 2013.

[73] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9446–9454, 2018.

[74] R. Heckel and M. Soltanolkotabi, "Compressive sensing with un-trained neural networks: gradient descent finds a smooth approximation", *Proceedings of the International Conference on Machine Learning*, pp. 4149–4158, 2020.

[75] J. Batson and L. Royer, "Noise2self: blind denoising by self-supervision", *Proceedings of the International Conference on Machine Learning*, pp. 524–533, 2019.

[76] K. P. Pruessmann, M. Weiger, P. Börnert, and P. Boesiger, "Advances in sensitivity encoding with arbitrary k-space trajectories", *Magnetic Resonance in Medicine*, vol. 46, no. 4, pp. 638–651, 2001.

[77] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation", in *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.

[78] F. Ong and M. Lustig, "Sigpy: a python package for high performance iterative reconstruction", *Proceedings of the International Society of Magnetic Resonance in Medicine, Montréal, QC*, vol. 4819, 2019.

[79] M. Uecker, P. Lai, M. J. Murphy, *et al.*, "Espirit—an eigenvalue approach to autocalibrating parallel mri: where sense meets grappa", *Magnetic Resonance in Medicine*, vol. 71, no. 3, pp. 990–1001, 2014.

[80] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: an imperative style, high-performance deep learning library", *Advances in neural information processing systems*, vol. 32, pp. 8026–8037, 2019.

[81] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[82] M. Z. Darestani and R. Heckel, "Accelerated mri with un-trained neural networks", *IEEE Transactions on Computational Imaging*, vol. 7, pp. 724–733, 2021.

[83] E. Mussard, T. Hilbert, C. Forman, R. Meuli, J.-P. Thiran, and T. Kober, "Accelerated mp2rage imaging using cartesian phyllotaxis readout and compressed sensing reconstruction", *Magnetic Resonance in Medicine*, vol. 84, no. 4, pp. 1881–1894, 2020.

[84] D. Salomon, *Data compression: the complete reference*. Springer Science & Business Media, 2004.

[85] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[86] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution", in *European conference on computer vision*, Springer, 2016, pp. 694–711.

[87] J. P. Woodard and M. P. Carley-Spencer, "No-reference image quality metrics for structural mri", *Neuroinformatics*, vol. 4, no. 3, pp. 243–262, 2006.

[88] Z. Zhang, G. Dai, X. Liang, S. Yu, L. Li, and Y. Xie, "Can signal-to-noise ratio perform as a baseline indicator for medical image quality assessment", *IEEE Access*, vol. 6, pp. 11 534–11 543, 2018.

[89] Z. Wang, H. R. Sheikh, and A. C. Bovik, "No-reference perceptual quality assessment of jpeg compressed images", *Proceedings of the International Conference on Image Processing*, vol. 1, pp. I–I, 2002.

[90] A. Mittal, A. K. Moorthy, and A. C. Bovik, "Blind/referenceless image spatial quality evaluator", *Proceedings of the Forty Fifth ASILOMAR Conference on Signals, Systems and Computers*, pp. 723–727, 2011.

[91] N. Venkatanath, D. Praneeth, M. C. Bh, S. S. Channappayya, and S. S. Medasani, "Blind image quality evaluation using perception based features", *Proceedings of the Twenty First National Conference on Communications (NCC)*, pp. 1–6, 2015.

[92] A. Lugmayr, M. Danelljan, R. Timofte, *et al.*, "NTIRE 2020 challenge on real-world image super-resolution: methods and results", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops*, pp. 2058–2076, 2020. DOI: 10.1109/CVPRW50498.2020.00255. [Online]. Available: https://openaccess.thecvf.com/content%5C_CVPRW%5C_2020/html/w31/Lugmayr%5C_NTIRE%5C_2020%5C_Challenge%5C_on%5C_Real-World%5C_Image%5C_Super-Resolution%5C_Methods%5C_and%5C_Results%5C_CVPRW%5C_2020%5C_paper.html.

[93] M. J. Muckley, B. Riemenschneider, A. Radmanesh, *et al.*, "Results of the 2020 fastmri challenge for machine learning mr image reconstruction", *IEEE Transactions on Medical Imaging*, vol. 40, no. 9, pp. 2306–2317, 2021.

[94] Y. Blau and T. Michaeli, "The perception-distortion tradeoff", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6228–6237, 2018.

[95] P. M. Adamson, B. Gunel, J. Dominic, *et al.*, "Ssfd: self-supervised feature distance as an mr image reconstruction quality metric", *NeurIPS 2021 Workshop on Deep Learning and Inverse Problems*, 2021.

[96] K. Wang, J. I. Tamir, and S. X. Yu, "High-fidelity reconstruction with instance-wise discriminative feature matching loss", *Proc. Intl. Soc. Mag. Reson. Med. 28*, 2019.

[97] F. Knoll, K. Hammernik, E. Kobler, T. Pock, M. P. Recht, and D. K. Sodickson, "Assessment of the generalization of learned image reconstruction and the potential for transfer learning", *Magnetic Resonance in Medicine*, vol. 81, no. 1, pp. 116–128, 2019.

[98] K. Hammernik, J. Schlemper, C. Qin, J. Duan, R. M. Summers, and D. Rueckert, "Systematic evaluation of iterative deep neural networks for fast parallel mri reconstruction with sensitivity-weighted coil combination", *Magnetic Resonance in Medicine*, vol. 86, no. 4, pp. 1859–1872, 2021. DOI: https://doi.org/10.1002/mrm.28827. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.28827. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.28827.

[99] V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen, "On instabilities of deep learning in image reconstruction and the potential costs of ai", *Proceedings of the National Academy of Sciences*, vol. 117, no. 48, pp. 30 088–30 095, 2020.

[100] M. Z. Darestani, A. S. Chaudhari, and R. Heckel, "Measuring robustness in deep learning based compressive sensing", *International Conference on Machine Learning*, Proceedings of Machine Learning Research, vol. 139, M. Meila and T. Zhang, Eds., pp. 2433–2444, 2021. [Online]. Available: http://proceedings.mlr.press/v139/darestani21a.html.

[101] M. P. Recht, J. Zbontar, D. K. Sodickson, *et al.*, "Using deep learning to accelerate knee mri at 3 t: results of an interchangeability study", *American journal of Roentgenology*, vol. 215, no. 6, p. 1421, 2020.

[102] M. Roux, T. Hilbert, M. Hussami, F. Becce, T. Kober, and P. Omoumi, "Mri t2 mapping of the knee providing synthetic morphologic images: comparison to conventional turbo spin-echo mri", *Radiology*, vol. 293, no. 3, pp. 620–630, 2019.

[103] R. Zhao, B. Yaman, Y. Zhang, *et al.*, "Fastmri+: clinical pathology annotations for knee and brain fully sampled multi-coil mri data", *arXiv preprint arXiv:2109.03812*, 2021.

[104] A. D. Desai, A. M. Schmidt, E. B. Rubin, *et al.*, "Skm-tea: a dataset for accelerated mri reconstruction with dense image labels for quantitative clinical evaluation", *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[105] T. Yu, M. Pizzolato, G. Girard, J. Rafael-Patino, E. J. Canales-Rodriguez, and J.-P. Thiran, "Robust biophysical parameter estimation with a neural network enhanced hamiltonian markov chain monte carlo sampler", *International Conference on Information Processing in Medical Imaging*, pp. 818–829, 2019.

[106] S. K. Sengijpta, *Fundamentals of Statistical Signal Processing: Estimation theory*, 1995.

[107] N. Balakrishnan and V. B. Nevzorov, *A Primer on Statistical Distributions*. John Wiley & Sons, 2004.

[108] M. Betancourt, "A conceptual introduction to Hamiltonian Monte Carlo", *arXiv preprint arXiv:1701.02434*, 2017.

[109] M. Ledoux, *The Concentration of Measure Phenomenon*, 89. American Mathematical Soc., 2001.

[110] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications", 1970.

[111] E. J. Canales Rodriguez, M. Pizzolato, Y. Aleman-Gomez, *et al.*, "Unified multi-modal characterization of microstructural parameters of brain tissue using diffusion MRI and multi-echo T2 data", in *Joint Annual Meeting ISMRM-ESMRMB*, 2018.

[112] E. Kaden, N. D. Kelm, R. P. Carson, M. D. Does, and D. C. Alexander, "Multi-compartment microscopic diffusion imaging", *NeuroImage*, vol. 139, pp. 346–359, 2016.

[113] A. L. MacKay and C. Laule, "Magnetic Resonance of Myelin Water: An in vivo Marker for Myelin", *Brain Plasticity*, vol. 2, no. 1, pp. 71–91, 2016.

[114] D. Levy, M. D. Hoffman, and J. Sohl-Dickstein, "Generalizing Hamiltonian Monte Carlo with Neural Networks", *arXiv preprint arXiv:1711.09268*, 2017.

[115] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

[116] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, "Hybrid Monte Carlo", *Physics Letters B*, vol. 195, no. 2, pp. 216–222, 1987.

[117] M. D. Hoffman and A. Gelman, "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo.", *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1593–1623, 2014.

[118] M. Girolami and B. Calderhead, "Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods", *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 73, no. 2, pp. 123–214, 2011.

[119] R. Fick, D. Wassermann, and R. Deriche, "Mipy: An Open-Source Framework to improve reproducibility in Brain Microstructure Imaging", in *OHBM 2018-Human Brain Mapping*, 2018, pp. 1–4.

[120] A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. M. Blei, "Automatic Differentiation Variational Inference", *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 430–474, 2017.

[121] A. L. Beam, S. K. Ghosh, and J. Doyle, "Fast Hamiltonian Monte Carlo using GPU Computing", *Journal of Computational and Graphical Statistics*, vol. 25, no. 2, pp. 536–548, 2016.

[122] E. T. McKinnon and J. H. Jensen, "Measuring intra-axonal t2 in white matter with direction-averaged diffusion mri", *Magnetic resonance in medicine*, vol. 81, no. 5, pp. 2985–2994, 2019.

[123] Y. Ji, B. Gagoski, W. S. Hoge, Y. Rathi, and L. Ning, "Accelerated diffusion and relaxation-diffusion mri using time-division multiplexing epi", *Magnetic Resonance in Medicine*, vol. 86, no. 5, pp. 2528–2541, 2021.

[124] B. Lampinen, F. Szczepankiewicz, J. Mårtensson, *et al.*, "Towards unconstrained compartment modeling in white matter using diffusion-relaxation mri with tensor-valued diffusion encoding", *Magnetic resonance in medicine*, vol. 84, no. 3, pp. 1605–1623, 2020.

[125] L. Ning, B. Gagoski, F. Szczepankiewicz, C.-F. Westin, and Y. Rathi, "Joint relaxation-diffusion imaging moments to probe neurite microstructure", *IEEE transactions on medical imaging*, vol. 39, no. 3, pp. 668–677, 2019.

[126] J. Martin, A. Reymbaut, M. Schmidt, *et al.*, "Nonparametric d-r1-r2 distribution mri of the living human brain", *NeuroImage*, vol. 245, p. 118 753, 2021.

[127] T. Yu, E. J. Canales-Rodriguez, M. Pizzolato, *et al.*, "Model-informed machine learning for multi-component t2 relaxometry", *Medical Image Analysis*, vol. 69, p. 101 940, 2021, ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2020.101940. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841520303042.

[128] R. Menon and P. Allen, "Application of continuous relaxation time distributions to the fitting of data from model systmes and excised tissue", *Magnetic resonance in medicine*, vol. 20, no. 2, pp. 214–227, 1991.

[129] F. Bloch, "Nuclear induction", *Physical review*, vol. 70, no. 7-8, p. 460, 1946.

[130] T. Prasloski, B. Mädler, Q.-S. Xiang, A. MacKay, and C. Jones, "Applications of stimulated echo correction to multicomponent t2 analysis", *Magnetic resonance in medicine*, vol. 67, no. 6, pp. 1803–1814, 2012.

[131] J. Hennig, "Multiecho imaging sequences with low refocusing flip angles", *Journal of Magnetic Resonance (1969)*, vol. 78, no. 3, pp. 397–407, 1988.

[132] K. J. Layton, M. Morelande, D. Wright, P. M. Farrell, B. Moran, and L. A. Johnston, "Modelling and estimation of multicomponent $T\_\{2\}$ distributions", *IEEE transactions on medical imaging*, vol. 32, no. 8, pp. 1423–1434, 2013.

[133] D. Neumann, M. Blaimer, P. M. Jakob, and F. A. Breuer, "Simple recipe for accurate t 2 quantification with multi spin-echo acquisitions", *Magnetic Resonance Materials in Physics, Biology and Medicine*, vol. 27, no. 6, pp. 567–577, 2014.

[134] G. F. Piredda, T. Hilbert, J.-P. Thiran, and T. Kober, "Probing myelin content of the human brain with mri: a review", *Magnetic Resonance in Medicine*, 2020.

[135] A. L. MacKay and C. Laule, "Myelin water imaging", *eMagRes*, 2007.

[136] A. Mackay, K. Whittall, J. Adler, D. Li, D. Paty, and D. Graeb, "In vivo visualization of myelin water in brain by magnetic resonance", *Magnetic resonance in medicine*, vol. 31, no. 6, pp. 673–677, 1994.

[137] K. P. Whittall, A. L. Mackay, D. A. Graeb, R. A. Nugent, D. K. Li, and D. W. Paty, "In vivo measurement of t2 distributions and water contents in normal human brain", *Magnetic resonance in medicine*, vol. 37, no. 1, pp. 34–43, 1997.

[138] V. Vasilescu, E. Katona, V. Simplaceanu, and D. Demco, "Water compartments in the myelinated nerve. iii. pulsed nmr result", *Experientia*, vol. 34, no. 11, pp. 1443–1444, 1978.

[139] R. Menon, M. Rusinko, and P. Allen, "Proton relaxation studies of water compartmentalization in a model neurological system", *Magnetic resonance in medicine*, vol. 28, no. 2, pp. 264–274, 1992.

[140] A. Raj, S. Pandya, X. Shen, E. LoCastro, T. D. Nguyen, and S. A. Gauthier, "Multi-compartment t2 relaxometry using a spatially constrained multi-gaussian model", *PLoS One*, vol. 9, no. 6, 2014.

[141] Y. P. Du, R. Chu, D. Hwang, *et al.*, "Fast multislice mapping of the myelin water fraction using multicompartment analysis of t decay at 3t: a preliminary postmortem study", *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 58, no. 5, pp. 865–870, 2007.

[142] T. Yu, M. Pizzolato, E. J. Canales-Rodriguez, and J.-P. Thiran, "Robust t 2 relaxometry with hamiltonian mcmc for myelin water fraction estimation", in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, IEEE, 2019, pp. 1813–1817.

[143] S. Chatterjee, O. Commowick, O. Afacan, S. K. Warfield, and C. Barillot, "Multi-compartment model of brain tissues from t2 relaxometry mri using gamma distribution", in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 141–144.

[144] A. Akhondi-Asl, O. Afacan, R. V. Mulkern, and S. K. Warfield, "T 2-relaxometry for myelin water fraction extraction using wald distribution and extended phase graph", in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2014, pp. 145–152.

[145] M. Björk, D. Zachariah, J. Kullberg, and P. Stoica, "A multicomponent t2 relaxometry algorithm for myelin water imaging of the brain", *Magnetic resonance in medicine*, vol. 75, no. 1, pp. 390–402, 2016.

[146] M. Prange and Y.-Q. Song, "Quantifying uncertainty in nmr t2 spectra using monte carlo inversion", *Journal of Magnetic Resonance*, vol. 196, no. 1, pp. 54–60, 2009.

[147] E. Alonso-Ortiz, I. R. Levesque, and G. B. Pike, "Mri-based myelin water imaging: a technical review", *Magnetic resonance in medicine*, vol. 73, no. 1, pp. 70–81, 2015.

[148] C. L. Lawson and R. J. Hanson, *Solving least squares problems*. Siam, 1995, vol. 15.

[149] K. P. Whittall and A. L. MacKay, "Quantitative interpretation of nmr relaxation data", *Journal of Magnetic Resonance (1969)*, vol. 84, no. 1, pp. 134–152, 1989.

[150] R. M. Kroeker and R. M. Henkelman, "Analysis of biological nmr relaxation data with continuous distributions of relaxation times", *Journal of Magnetic Resonance (1969)*, vol. 69, no. 2, pp. 218–235, 1986.

[151] C. Laule, E. Leung, D. K. Li, *et al.*, "Myelin water imaging in multiple sclerosis: quantitative correlations with histopathology", *Multiple Sclerosis Journal*, vol. 12, no. 6, pp. 747–753, 2006.

[152] S. J. Graham, P. L. Stanchev, and M. J. Bronskill, "Criteria for analysis of multicomponent tissue t2 relaxation data", *Magnetic Resonance in Medicine*, vol. 35, no. 3, pp. 370–378, 1996.

[153] T. Andrews, J. L. Lancaster, S. J. Dodd, C. Contreras-Sesvold, and P. T. Fox, "Testing the three-pool white matter model adapted for use with t2 relaxometry", *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 54, no. 2, pp. 449–454, 2005.

[154] V. Wiggermann, I. M. Vavasour, S. Kolind, A. L. MacKay, G. Helms, and A. Rauscher, "Non-negative least squares computation for in vivo myelin mapping using simulated multi-echo spin-echo t2 decay data", *NMR in Biomedicine*, e4277, 2020.

[155] D. Kumar, T. D. Nguyen, S. A. Gauthier, and A. Raj, "Bayesian algorithm using spatial priors for multiexponential t2 relaxometry from multiecho spin echo mri", *Magnetic resonance in medicine*, vol. 68, no. 5, pp. 1536–1543, 2012.

[156] A. Raj, S. Pandya, X. Shen, E. LoCastro, T. D. Nguyen, and S. A. Gauthier, "Multi-compartment t2 relaxometry using a spatially constrained multi-gaussian model", *PLoS One*, vol. 9, no. 6, e98391, 2014.

[157] J. Lee, D. Lee, J. Y. Choi, D. Shin, H.-G. Shin, and J. Lee, "Artificial neural network for myelin water imaging", *Magnetic resonance in medicine*,

[158] H. Liu, Q.-S. Xiang, R. Tam, *et al.*, "Myelin water imaging data analysis in less than one minute", *NeuroImage*, vol. 210, p. 116 551, 2020, ISSN: 1053-8119.

[159] T. Prasloski, A. Rauscher, A. L. MacKay, *et al.*, "Rapid whole cerebrum myelin water imaging using a 3d grase sequence", *Neuroimage*, vol. 63, no. 1, pp. 533–539, 2012.

[160] C. Villani, "The wasserstein distances", in *Optimal Transport*, Springer, 2009, pp. 93–111.

[161] C. Laule, I. M. Vavasour, S. H. Kolind, *et al.*, "Long t2 water in multiple sclerosis: what else can we learn from multi-echo t2 relaxation?", *Journal of neurology*, vol. 254, no. 11, pp. 1579–1587, 2007.

[162] J. P. Wansapura, S. K. Holland, R. S. Dunn, and W. S. Ball Jr, "Nmr relaxation times in the human brain at 3.0 tesla", *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 9, no. 4, pp. 531–538, 1999.

[163] A. Ramdas, N. G. Trillos, and M. Cuturi, "On wasserstein two-sample testing and related families of nonparametric tests", *Entropy*, vol. 19, no. 2, p. 47, 2017.

[164] Martín Abadi, Ashish Agarwal, Paul Barham, *et al.*, *TensorFlow: large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015. [Online]. Available: http://tensorflow.org/.

[165] G. Van Rossum *et al.*, *Python reference manual*.

[166] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python", *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: https://doi.org/10.1038/s41592-019-0686-2.

[167] J. Cohen-Adad, M. Does, T. DUVAL, *et al.*, *White matter microscopy database*, Mar. 2020. DOI: 10.17605/OSF.IO/YP4QG. [Online]. Available: osf.io/yp4qg.

[168] M.-T. Vuong, T. Duval, J. Cohen-Adad, and N. Stikov, "On the precision of myelin imaging: characterizing ex vivo dog spinal cord.", 2017, p. 3760.

[169] A. Zaimi, M. Wabartha, V. Herman, P.-L. Antonsanti, C. S. Perone, and J. Cohen-Adad, "Axondeepseg: automatic axon and myelin segmentation from microscopy data using convolutional neural networks", *Scientific reports*, vol. 8, no. 1, pp. 1–11, 2018.

[170] G. F. Piredda, T. Hilbert, E. J. Canales-Rodrigez, *et al.*, "Fast and high-resolution myelin water imaging: accelerating multi-echo grase with caipirinha", *Magnetic Resonance in Medicine,*, 2020.

[171] M. Brant-Zawadzki, G. D. Gillan, and W. R. Nitz, "Mp rage: a three-dimensional, t1-weighted, gradient-echo sequence–initial experience in the brain.", *Radiology*, vol. 182, no. 3, pp. 769–775, 1992.

[172] B. De Coene, J. V. Hajnal, P. Gatehouse, *et al.*, "Mr of the brain using fluid-attenuated inversion recovery (flair) pulse sequences.", *American journal of neuroradiology*, vol. 13, no. 6, pp. 1555–1564, 1992.

[173] F. La Rosa, A. Abdulkadir, M. J. Fartaria, *et al.*, "Multiple sclerosis cortical and wm lesion segmentation at 3t mri: a deep learning method based on flair and mp2rage", *NeuroImage: Clinical*, p. 102 335, 2020.

[174] I. R. Levesque, P. S. Giacomini, S. Narayanan, *et al.*, "Quantitative magnetization transfer and myelin water imaging of the evolution of acute multiple sclerosis lesions", *Magnetic resonance in medicine*, vol. 63, no. 3, pp. 633–640, 2010.

[175] S. H. Kolind, B. Mädler, S. Fischer, D. K. Li, and A. L. MacKay, "Myelin water imaging: implementation and development at 3.0 t and comparison to 1.5 t measurements", *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 62, no. 1, pp. 106–115, 2009.

[176] C. Laule, P. Kozlowski, E. Leung, D. K. Li, A. L. MacKay, and G. W. Moore, "Myelin water imaging of multiple sclerosis at 7 t: correlations with histopathology", *Neuroimage*, vol. 40, no. 4, pp. 1575–1580, 2008.

[177] E. Alonso-Ortiz, I. R. Levesque, R. Paquin, and G. B. Pike, "Field inhomogeneity correction for gradient echo myelin water fraction imaging", *Magnetic resonance in medicine*, vol. 78, no. 1, pp. 49–57, 2017.

[178] K. Oishi, K. Zilles, K. Amunts, *et al.*, "Human brain white matter atlas: identification and assignment of common anatomical structures in superficial white matter", *Neuroimage*, vol. 43, no. 3, pp. 447–457, 2008.

[179] S. Mori, K. Oishi, H. Jiang, *et al.*, "Stereotaxic white matter atlas based on diffusion tensor imaging in an icbm template", *Neuroimage*, vol. 40, no. 2, pp. 570–582, 2008.

[180] J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks", *Inverse Problems*, vol. 33, no. 12, p. 124 007, 2017.

[181] S. Aja-Fernández and G. Vegas-Sánchez-Ferrero, "Statistical analysis of noise in mri", *Switzerland: Springer International Publishing*, 2016.

[182] J. G. Sled, "Modelling and interpretation of magnetization transfer imaging in the brain", *Neuroimage*, vol. 182, pp. 128–135, 2018.

[183] S. J. Malik, R. P. A. Teixeira, and J. V. Hajnal, "Extended phase graph formalism for systems with magnetization transfer and exchange", *Magnetic resonance in medicine*, vol. 80, no. 2, pp. 767–779, 2018.

[184] C. Birkl, J. Doucette, M. Fan, E. Hernández-Torres, and A. Rauscher, "Myelin water imaging depends on white matter fiber orientation in the human brain", *Magnetic resonance in medicine*, vol. 85, no. 4, pp. 2221–2231, 2021.

[185] D. Wang, J. Ostenson, and D. S. Smith, "Snapmrf: gpu-accelerated magnetic resonance fingerprinting dictionary generation and matching using extended phase graphs", *Magnetic Resonance Imaging*, vol. 66, pp. 248–256, 2020.

[186] M. D. Does, J. L. Olesen, K. D. Harkins, *et al.*, "Evaluation of principal component analysis image denoising on multi-exponential mri relaxometry", *Magnetic resonance in medicine*, vol. 81, no. 6, pp. 3503–3514, 2019.

[187] M. Bouhrara, D. A. Reiter, M. C. Maring, J.-M. Bonny, and R. G. Spencer, "Use of the nesma filter to improve myelin water fraction mapping with brain mri", *Journal of Neuroimaging*, vol. 28, no. 6, pp. 640–649, 2018.

[188] C. El-Hajj, S. Moussaoui, G. Collewet, and M. Musse, "Multi-exponential transverse relaxation times estimation from magnetic resonance images under rician noise and spatial regularization", *IEEE Transactions on Image Processing*, 2020.

[189] D. Hwang and Y. P. Du, "Improved myelin water quantification using spatially regularized non-negative least squares algorithm", *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 30, no. 1, pp. 203–208, 2009.

[190] D. Kumar, H. Hariharan, T. D. Faizy, *et al.*, "Using 3d spatial correlations to improve the noise robustness of multi component analysis of 3d multi echo quantitative t2 relaxometry data", *NeuroImage*, vol. 178, pp. 583–601, 2018.

[191] M. Nagtegaal, P. Koken, T. Amthor, *et al.*, "Myelin water imaging from multi-echo t2 mr relaxometry data using a joint sparsity constraint", *NeuroImage*, p. 117 014, 2020.

[192] T. D. Nguyen, C. Wisnieff, M. A. Cooper, *et al.*, "T2prep three-dimensional spiral imaging with efficient whole brain coverage for myelin water quantification at 1.5 tesla", *Magnetic Resonance in Medicine*, vol. 67, no. 3, pp. 614–621, 2012.

[193] S. Hochreiter and J. Schmidhuber, "Long short-term memory", *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[194] M. S. Rad*, T. Yu*, C. Musat, H. K. Ekenel, B. Bozorgtabar, and J.-P. Thiran, "Benefiting from bicubically down-sampled images for learning real-world image super-resolution", *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1590–1599, Jan. 2021.

[195] J. Cai, H. Zeng, H. Yong, Z. Cao, and L. Zhang, "Toward real-world single image super-resolution: a new benchmark and a new model", in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[196] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, *et al.*, "Ntire 2017 challenge on single image super-resolution: methods and results", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jul. 2017.

[197] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution", in *European conference on computer vision*, Springer, 2014, pp. 184–199.

[198] J. Van Ouwerkerk, "Image super-resolution survey", *Image and vision Computing*, vol. 24, no. 10, pp. 1039–1052, 2006.

[199] Z. Wang, J. Chen, and S. C. Hoi, "Deep learning for image super-resolution: a survey", *arXiv preprint arXiv:1902.06068*, 2019.

[200] S. Anwar, S. Khan, and N. Barnes, "A deep journey into super-resolution: a survey", *arXiv preprint arXiv:1904.07523*, 2019.

[201] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: a persistent memory network for image restoration", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4539–4547.

[202] X. Wang, K. Yu, S. Wu, *et al.*, "Esrgan: enhanced super-resolution generative adversarial networks", in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 0–0.

[203] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 065–11 074.

[204] A. Lugmayr, M. Danelljan, and R. Timofte, "Unsupervised learning for real-world super-resolution", in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[205] A. Lugmayr, M. Danelljan, R. Timofte, *et al.*, "Aim 2019 challenge on real-world image super-resolution: methods and results", *arXiv preprint arXiv:1911.07783*, 2019.

[206] T. Köhler, M. Bätz, F. Naderi, A. Kaup, A. K. Maier, and C. Riess, "Benchmarking super-resolution algorithms on real data", *CoRR*, vol. abs/1709.04881, 2017. arXiv: 1709.04881. [Online]. Available: http://arxiv.org/abs/1709.04881.

[207] N. Efrat, D. Glasner, A. Apartsin, B. Nadler, and A. Levin, "Accurate blur models vs. image priors in single image super-resolution", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2832–2839.

[208] R. Zhou and S. Susstrunk, "Kernel modeling super-resolution on real low-resolution images", in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2433–2443.

[209] J. Gu, H. Lu, W. Zuo, and C. Dong, "Blind super-resolution with iterative kernel correction", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1604–1613.

[210] K. Zhang, W. Zuo, and L. Zhang, "Deep plug-and-play super-resolution for arbitrary blur kernels", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1671–1681.

[211] ——, "Learning a single convolutional super-resolution network for multiple degradations", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3262–3271.

[212] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu, "Camera lens super-resolution", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1652–1660.

[213] A. Bulat, J. Yang, and G. Tzimiropoulos, "To learn image super-resolution, use a gan to learn how to do image degradation first", in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 185–200.

[214] S. Bell-Kligler, A. Shocher, and M. Irani, *Blind super-resolution kernel estimation using an internal-gan*, 2019. arXiv: 1909.06581 [cs.CV].

[215] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 701–710.

[216] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks", in *Advances in neural information processing systems*, 2017, pp. 700–708.

[217] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for neural networks for image processing", *arXiv preprint arXiv:1511.08861*, 2015.

[218] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets", in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[219] C. Ledig, L. Theis, F. Huszár, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network", *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 105–114, 2016.

[220] B. Bozorgtabar, M. S. Rad, H. Kemal Ekenel, and J. Thiran, "Using photorealistic face synthesis and domain adaptation to improve facial expression analysis", in *2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019)*, 2019, pp. 1–8.

[221] B. Bozorgtabar, M. S. Rad, H. K. Ekenel, and J.-P. Thiran, "Learn to synthesize and synthesize to learn", *Computer Vision and Image Understanding*, vol. 185, pp. 1–11, 2019, ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2019.04.010.

[222] C. Ledig, L. Theis, F. Huszár, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[223] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *CoRR*, vol. abs/1409.1556, 2014. arXiv: 1409.1556. [Online]. Available: http://arxiv.org/abs/1409.1556.

[224] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution", *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1132–1140, 2017.

[225] A. Ignatov, N. Kobyshev, R. Timofte, K. Vanhoey, and L. Van Gool, "Dslr-quality photos on mobile devices with deep convolutional networks", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3277–3285.

[226] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "2018 PIRM challenge on perceptual image super-resolution", *CoRR*, vol. abs/1809.07517, 2018. arXiv: 1809.07517. [Online]. Available: http://arxiv.org/abs/1809.07517.

[227] M. S. M. Sajjadi, B. Schölkopf, and M. Hirsch, "Enhancenet: single image super-resolution through automated texture synthesis", *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4501–4510, 2016.

[228] M. S. Rad, B. Bozorgtabar, U.-V. Marti, M. Basler, H. K. Ekenel, and J.-P. Thiran, "Srobb: targeted perceptual loss for single image super-resolution", in *The IEEE International Conference on Computer Vision (ICCV)*, Oct. 2019.

[229] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform", *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018.

[230] M. S. Rad, B. Bozorgtabar, C. Musat, *et al.*, "Benefiting from multitask learning to improve single image super-resolution", *Neurocomputing*, vol. 398, pp. 304–313, 2020, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2019.07.107.

[231] C. Ma, C. Yang, X. Yang, and M. Yang, "Learning a no-reference quality metric for single-image super-resolution", *CoRR*, vol. abs/1612.05890, 2016. arXiv: 1612.05890.

[232] A. Mittal, R. Soundararajan, and A. Bovik, "Making a completely blind image quality analyzer", *Signal Processing Letters, IEEE*, vol. 20, pp. 209–212, Mar. 2013. DOI: 10.1109/LSP.2012.2227726.

[233] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks", in *ECCV*, 2018.

[234] X. Wang, K. Yu, S. Wu, *et al.*, "Esrgan: enhanced super-resolution generative adversarial networks", in *ECCV Workshops*, 2018.

Publications

Articles in peer-reviewed journals

1. **Yu, T.**, Canales-Rodríguez, E. J., Pizzolato, M., Piredda, G. F., Hilbert, T., Fischi-Gomez, E., ..., Thiran, J. P. (2021).
   Model-informed machine learning for multi-component T2 relaxometry. *Medical Image Analysis*, 69, 101940. doi: 10.1016/j.media.2020.101940.

2. La Rosa, F., **Yu, T.**, Barquero, G., Thiran, J. P., Granziera, C., Cuadra, M. B.
   MPRAGE to MP2RAGE UNI translation via generative adversarial network improves the automatic tissue and lesion segmentation in multiple sclerosis patients. *Computers in Biology and Medicine*. 2021;132:104297. doi: 10.1016/j.compbiomed.2021.104297

3. Canales-Rodríguez EJ., Pizzolato M., **Yu T.**, Piredda GF., Hilbert T., Radua J., Kober T., Thiran JP.
   Revisiting the T_2 spectrum imaging inverse problem: Bayesian regularized non-negative least squares. *NeuroImage*. 2021;244:118582. doi: 10.1016/j.neuroimage.2021.118582.

4. Canales-Rodríguez EJ., Pizzolato M., Piredda GF., Hilbert T., Kunz N., Pot C., **Yu T.**, Salvador R., Pomarol-Clotet E., Kober T., Thiran JP., Daducci A.
   Comparison of non-parametric T_2 relaxometry methods for myelin water quantification. *Med Image Anal*. 2021;101959. doi: 10.1016/j.media.2021.101959.

5. Khawam, M., De Dumast, P., Deman, P., Kebiri, H., **Yu, T.**, Tourbier, S., ..., Koob, M.
   Fetal brain biometric measurements on 3D super-resolution reconstructed T2-weighted MRI: an intra-and inter-observer agreement study. *Frontiers in pediatrics*. 2021:9. doi: 10.3389/fped.2021.639746

6. Schilling, K. G., Rheault, F., Petit, L., Hansen, C. B., Nath, V., Yeh, F. C., ..., **Yu T.** ..., Descoteaux, M.
   Tractography dissection variability: what happens when 42 groups dissect 14 white matter bundles on the same dataset?. *NeuroImage*. (2021); 243:118502. doi: 10.1016/j.neuroimage.2021.118502

Articles currently under revision in peer-reviewed journals

1. Lajous, H., Roy, C. W., Hilbert, T., de Dumast, P., Tourbier, S., Alemán-Gómez, Y., ..., **Yu T.** ..., Cuadra, M. B.
   FaBiAN: A Fetal Brain magnetic resonance Acquisition Numerical phantom. (Under Review in *Scientific Reports*)

Articles submitted to peer-reviewed journals

1. **Yu, T.**, Hilbert T, Piredda GF, Canales-Rodríguez, E.,Kober T., Thiran JP.
   Validation and Generalizability of Self-Supervised Image Reconstruction Methods for Undersampled MRI
   Submitted to *The Journal of Machine Learning for Biomedical Imaging*

Articles in proceedings of international conferences

1. **Yu T.**, Pizzolato M., Girard G., Rafael-Patino J., Canales-Rodríguez E.J., Thiran JP.
   Robust Biophysical Parameter Estimation with a Neural Network Enhanced Hamiltonian Markov Chain Monte Carlo Sampler. *Information Processing in Medical Imaging*. IPMI 2019. Lecture Notes in Computer Science, vol 11492. Springer, Cham. doi: 10.1007/978-3-030-20351-1_64

2. Rad, M. S. †, **Yu, T.**†, Musat, C., Ekenel, H. K., Bozorgtabar, B., Thiran, J. P.
   Benefiting from bicubically down-sampled images for learning real-world image super-resolution. Proceedings of the *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021 (pp. 1590-1599). Link.
   † Rad and Yu contributed equally.

3. **Yu, T.**, Pizzolato, M., Canales-Rodríguez, E. J., Thiran, J. P.
   Robust T_2 relaxometry with hamiltonian MCMC for myelin water fraction estimation. IEEE 16th *International Symposium on Biomedical Imaging* (ISBI 2019) (pp. 1813-1817). IEEE. 10.1109/ISBI.2019.8759446

4. Lajous, H., Hilbert, T., Roy, C. W., Tourbier, S., de Dumast, P., **Yu, T.**, ..., Cuadra, M. B.
   T_2 Mapping from Super-Resolution-Reconstructed Clinical Fast Spin Echo Magnetic Resonance Acquisitions. *International Conference on Medical Image Computing and Computer-Assisted Intervention* (MICCAI 2020) (pp. 114-124). Springer, Cham. 10.1007/978-3-030-59713-9_12

Articles in proceedings of conference workshops

1. Rad, M. S., **Yu T.**, Bozorgtabar, B., Thiran, J. P.
   Test-Time Adaptation for Super-Resolution: You Only Need to Overfit on a Few More Images

Proceedings of *the IEEE/CVF International Conference on Computer Vision (ICCV) Workshop on Advanced Image Manipulation* (AIM), 2021, pp. 1845-1854. Link.

2. Patino Lopez, J. R., **Yu, T.**, Delvigne, V., Barakovic, M., Pizzolato, M., Girard, G., ..., Thiran, J. P.
   DWI Simulation-Assisted Machine Learning Models for Microstructure Estimation. *Computational Diffusion MRI, MICCAI Workshop*, Shenzhen, China, October 2019 (No. CONF). Springer Nature. 10.1007/978-3-030-52893-5_11

Abstracts in proceedings of international conferences

1. Rafael-Patino, J, **Yu T.**, Andersson, M., Kjer, H.M., Dahl, V.A., Pacureanu, A., Dahl, A.B., Dyrby T.B., Thiran, J.P.
   Phantoms for Diffusion Simulations:Multi-Objective Differential Evolution for Realistic Numerical (MODERN) Phantoms., 2019 Proc. Intl. Soc. Mag. Reson. Med. 27 (2019), Montreal,Canada, 2019

2. Rafael-Patino, J. Girard, G., Fischi, E., Romascano, D., **Yu, T.**, Pizzolato, M., Ramirez-Manzanares, A., Canales Rodríguez, E., Thiran, J.P.
   Multi-diffusion and Multi-T2 weighted Monte-Carlo Simulations. 2020 OHBM ANNUAL MEETING

Thomas Yu

Contact Information

Mobile: +1 484-202-0608
E-mail: thomas.yu@epfl.ch
Address: Route du Bois 61, Ecublens 1024, Vaud

Biographical Information

Citizenship: United States of America

Education

École Polytechnique Fédérale de Lausanne, Department of Electrical Engineering

PhD Student in Medical Imaging **May 2018-Current (Anticipated Graduation: May 2022)**

Erasmus Mundus Joint Master Program in Mathematical Modelling
Joint Student at the University of Hamburg(primary), University of L'Aquila(secondary), Local Degree from L'Aquila: 110/110 e lode **September 2015-September 2017**

Yale University
Student in Applied Physics PhD Program (Left for EM Master Program) **July 2014-November 2014**

The University of Chicago, Chicago, Illinois
Bachelor of Arts in Physics, Bachelor of Science in Mathematics with Honors, **September 2011-June 2014**

Honours and Awards

Deans List (2012,2013,2014)

Full Scholarship(tuition, travel expenses, living expenses) from the Erasmus Mundus program(from EU) for a Joint Master's degree. (2015-2017)

Marie Skłodowska-Curie Actions PhD Fellowship (2018-2021) Agreement No 765148, TRABIT

Research Experience

Research Internship at Siemens Healthineers, Lausanne (**November 2020-Current**)

- Research focuses on self-supervised approaches to reconstructing MR images directly from raw data from the scanner (k-space). Working on both new methods as well as a rigorous validation of existing methods.

PhD Research in the group of Prof. Jean-Philippe Thiran, EPFL (**May 2018-Current**)

- Research focuses on combining machine learning and model-driven solutions to inverse problems mainly in the context of medical image reconstruction and computer vision. E.g. Reconstructing quantitative maps of diffusion/T_2 relaxation in MRI, Single image super-resolution, T_2 Spectrum Imaging.

Research Internship with Prof. Dr. Alexander Schlaefer, Technical University of Hamburg. **October 2017- April 2018**

- Machine Learning applied to treatment selection in HDR brachytherapy for cancer treatment.

Master Thesis with Prof. Dr. Alexander Schlaefer, Technical University of Hamburg. Co-supervised by Prof. Dr. Christina Brandt at University of Hamburg **April 2017- September 2017**

- Subject of Master Thesis: Robust Inverse Planning in High Dose Rate Brachytherapy using robotic needle insertion.
- I learned/used linear, nonlinear, and stochastic optimization. For programming I used Python, C++, and Matlab for different parts of the thesis. In particular, I used C++ for CGAL(Computational geometry library) and Python for PyMC3(probabilistic programming library).

Prof. A. Douglas Stone Research Group, Yale University **July 2014-November 2014**

- Learned dynamical systems at the level of Perko as well as how to use the Python package PyDSTool(for numerical solutions of differential equations and bifurcation analysis) in order to study bifurcation and chaos in lasers.

Prof. David Schuster Research Group, University of Chicago **October 2011- June 2014**

- Worked with Walter Lawrence (affiliated with James Franck Institute) on designing a circuit of parallel arrays of Josephson Junctions to act as a stable, flux controlled current switch for possible use in experimentation in Schuster Lab using simulations in Mathematica.
- Worked with Walter Lawrence on designing a superconducting qubit that minimized decoherence times using simulation and optimization of quantum circuits in Python.
- Construction of decade resistor boxes and construction/implementation of an Arduino relaybox system to remotely control RF switches for use in experiments.

Funded Participant of the 2012/2013 University of Chicago VIGRE Math Research Experience for Undergraduates **June 2012/2013- August 2012/2013**

- Research topics: Calculus of variations, Riemannian Geometry, and Mathematics of General Relativity

SKILLS

Python, C++, and Matlab. Familiarity with Microsoft Word, Excel, etc.

Languages Known: English(Native), Korean (fluent), French (Basic)

REFERENCES

David Schuster, Professor of Physics at the University of Chicago, David.Schuster@uchicago.edu

Alexander Schlaefer, Professor of Medical Technology at the Technical University of Hamburg, schlaefer@tuhh.de

Christina Brandt, Professor of Mathematics at the University of Hamburg, Christina.Brandt@uni-hamburg.de

Jean-Philippe Thiran, Full Professor in Electrical Engineering at École Polytechnique Fédérale de Lausanne
jean-philippe.thiran@epfl.ch

Tobias Kober, Director MR RD Switzerland - Advanced Clinical Imaging Technology
tobias.kober@siemens-healthineers.com