

Low Dimensional Methods for High Dimensional Magnetic Resonance Imaging

Frank Ong



Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2018-181

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-181.html>

December 14, 2018

Copyright © 2018, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Low Dimensional Methods for High Dimensional Magnetic Resonance Imaging

by

Frank Ong

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Associate Professor Michael Lustig, Chair

Professor Kannan Ramchandran

Professor Ming Gu

Fall 2018

The dissertation of Frank Ong, titled Low Dimensional Methods for High Dimensional Magnetic Resonance Imaging, is approved:

Chair	_____	Date	_____
	_____	Date	_____
	_____	Date	_____

University of California, Berkeley

Low Dimensional Methods for High Dimensional Magnetic Resonance Imaging

Copyright 2018
by
Frank Ong

Abstract

Low Dimensional Methods for High Dimensional Magnetic Resonance Imaging

by

Frank Ong

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Associate Professor Michael Lustig, Chair

Magnetic Resonance Imaging (MRI) is an amazing imaging modality in many aspects. It offers one of the best imaging contrast for visualizing soft tissues. It has no ionizing radiation at all. Its flexibility has also enabled many applications, including assessing blood flow, imaging brain activity via oxygenation contrast, and measuring tissue stiffness. Since MRI was invented, this imaging technology has saved numerous lives, and has been the frontier of biomedical and engineering research.

On the other hand, imaging speed remains a main limitation of MRI. Inherently, MRI takes time to collect measurements, and often requires minutes to complete a scan. In this regard, MRI is quite similar to early cameras: Subjects have to be motionless for minutes to obtain an image, which is uncomfortable to patients. This often leads to motion and motion artifacts. When severe motion artifacts occur, scans have to be repeated.

This dissertation aims to change that by developing techniques to reconstruct three-dimensional (3D) dynamic MRI from continuous acquisitions. An ideal 3D dynamic scan would be able to resolve all dynamics at a high spatiotemporal resolution. Subjects would not have to be motionless. The comprehensive information in the single scan would also greatly simplify clinical workflow. While this dissertation has not achieved this ideal scan yet, it proposes several innovations toward this goal. In particular, www.doi.org/10.6084/m9.figshare.7464485 shows a 3D rendering of a reconstruction result from this dissertation. Arbitrary slices at different orientation can be selected over time. Respiratory motion, contrast enhancements, and even slight bulk motion can be seen.

The main challenge in high resolution 3D dynamic MRI is that the reconstruction problem is inherently underdetermined and demanding of computation and memory. To overcome these challenges, this dissertation builds on top of many fundamental methods, including non-Cartesian imaging, parallel imaging and compressed sensing. In particular, this dissertation heavily relies on the compressed sensing framework, which has three components: 1) the image of interest has a compressed signal representation. 2) MRI can acquire (pseudo)-randomized samples in k-space, which provides incoherent encoding of the underlying image.

3) sparsity/compressibility can be efficiently enforced in reconstruction to recover the compressed representation from the undersampled measurements.

In this dissertation, I propose a multiscale low rank model that can compactly represent dynamic image sequences. The resulting representation can be applied beyond MRI, and is useful for other applications, such as motion separation in surveillance video. With the multiscale low rank representation, I propose a technique incorporating stochastic optimization to efficiently reconstruct 3D dynamic MRI. This makes it feasible to run such large-scale reconstructions on local workstations. To further speed up the reconstruction time, I propose accelerating the convergence of non-Cartesian reconstruction using a specially designed preconditioner. Finally, I leverage external undersampled datasets to further improve reconstruction quality using convolutional sparse coding.

Contents

Contents	i
List of Figures	iii
List of Tables	ix
1 Introduction	1
1.1 Outline	2
2 MRI Reconstruction Overview	5
2.1 Basic MRI Signal Equation	5
2.2 Basic MRI Reconstruction	7
2.3 View Sharing	8
2.4 Gating and Data Binning	9
2.5 Parallel Imaging	10
2.6 Compressed Sensing	12
3 Multiscale Low Rank Matrix Model	15
3.1 Introduction	15
3.2 Multiscale Low Rank Matrix Modeling	17
3.3 Problem Formulation and Convex Relaxation	20
3.4 Guidance on Choosing Regularization Parameters	22
3.5 Theoretical Analysis for Exact Decomposition	23
3.6 Theoretical Analysis for Approximate Decomposition	25
3.7 An Iterative Algorithm for Solving the Multiscale Low Rank Decomposition	26
3.8 Computational Complexity	28
3.9 Heuristics for translation invariant decomposition	29
3.10 Applications	31
3.11 Conclusion	36
3.12 Proof of Theorem 3.5.1	37
3.13 Proof of Theorem 3.6	41
4 High Resolution Volumetric Dynamic MRI from Non-Gated Acquisitions	47

4.1	Introduction	47
4.2	Forward Model with MSLR	48
4.3	Objective function using the Burer-Monteiro Heuristic	50
4.4	Stochastic Alternating Minimization	51
4.5	Methods	53
4.6	Results	55
4.7	Discussion	59
4.8	Conclusion	62
5	Accelerating Convergence for Non-Cartesian MRI Reconstructions	63
5.1	Introduction	63
5.2	Problem Setup	66
5.3	k-space Preconditioning	68
5.4	L2 optimized diagonal k-space preconditioner	70
5.5	Experiments	73
5.6	Discussion	80
5.7	Conclusion	83
5.8	Derivation for the Dual Problem	83
5.9	Derivation for L2 optimized diagonal preconditioner	84
6	Learning a Sparse Representation from Many Undersampled Datasets	85
6.1	Introduction	85
6.2	Problem statement and Objective function	86
6.3	Training using Stochastic Alternating Minimization	88
6.4	Reconstruction using the learned filters	89
6.5	Methods	89
6.6	Results	90
6.7	Discussion	93
6.8	Conclusion	94
7	Summary and Future Work	95
7.1	Summary of Contributions	95
7.2	Future work	96
	Bibliography	97

List of Figures

1.1	A 3D rendering of a result from Chapter 4 (full-length video available at www.doi.org/10.6084/m9.figshare.7464485). Arbitrary slices at different orientation can be selected over time. From the video, respiratory motion, contrast enhancements, and even slight bulk motion can be seen.	2
2.1	Throughout this dissertation, I consider the discrete Fourier approximate formulation of the MRI acquisition process.	5
2.2	Common MRI sampling trajectories: 2D Cartesian, radial and spiral trajectories, and 3D Cartesian, radial, and cones trajectories.	6
2.3	View sharing is a technique that fills in the missing k-space using samples from nearby time frames. In particular, this is often done in a sliding window manner.	8
2.4	Gridding reconstructions with various levels of view sharing of a real-time 2D cardiac dataset, acquired with a spiral trajectory. Compared to no view sharing, view sharing with five interleaves greatly reduce the undersampling artifacts. Dynamics over time can be seen. However, when pushed to use 50 interleaves, view sharing results in essentially static frames.	9
2.5	Gating and data binning exploit the periodicity nature of the underlying dynamics, such as cardiac and respiratory motion. In particular, k-space samples are acquired over many cycles. Then these measurements are sorted according to their corresponding phase in the cycle. With long enough scan time, the resulting gated k-space for each phase can even be fullsampled.	10
2.6	Gated reconstructions of the real-time cardiac dataset over six cardiac cycles, with and without simulated bulk motion. To simulate bulk motion, k-space measurements, and hence the underlying image, for the first cardiac cycle were rotated 30 degrees counterclock-wise. Motion artifacts can be seen for the corrupted image.	11
2.7	Parallel imaging utilizes additional spatial encodings provided by multichannel coil arrays to compensate for undersampling. In particular, each coil element is sensitive to particular region of the underlying object. This effectively multiplies the underlying image with spatially varying coil sensitivity maps.	12

2.8	Illustration of an example application of compressed sensing MRI. There are three components: 1) The wavelet transform is used to sparsely represent the underlying brain image. 2) Randomized k-space lines are acquired, which provides incoherent encoding of the underlying image. 3) ℓ_1 wavelet regularized reconstruction is used to efficiently recover the compressed representation from the undersampled measurements.	13
3.1	An example of the proposed multiscale low rank decomposition compared with other low rank methods. Each blob in the input matrix is a rank-1 matrix constructed from an outer product of hanning windows. Only the multiscale low rank decomposition exactly separates the blobs to their corresponding scales and represents each blob as compactly as possible.	16
3.2	Illustration of a multiscale matrix partition and its associated multiscale low rank modeling. Since the zero matrix is a matrix with the least rank, the multiscale modeling naturally extends to sparse matrices as 1×1 low rank matrices.	17
3.3	Illustration of the block reshape operator \mathcal{B}_b . \mathcal{B}_b extracts block b from the full matrix and reshapes it into an $m_i \times n_i$ matrix. Its adjoint operator \mathcal{B}_b^H takes an $m_i \times n_i$ matrix and embeds it into a full-size zero matrix.	18
3.4	Illustration of another multiscale matrix partition and its associated multiscale low rank modeling. Here, only the vertical dimension of the matrix is decimated. Since a $1 \times N$ matrix is low rank if and only if it is zero, the multiscale modeling naturally extends to group sparse matrices.	19
3.5	An example of the multiscale low rank decomposition in the presence of additive Gaussian noise by solving the convex program (3.1).	20
3.6	A conceptual illustration of how to obtain a multiscale low rank decomposition. First, I extract each block from the input matrix and perform a thresholding operation on its singular value to recover the significant components. Then, I subtract these significant components from the input matrix, thereby enabling the recovery of weaker, previously submerged components.	27
3.7	An example of the multiscale low rank decomposition with and without random cycle spinning. Each blob in the input matrix \mathbf{Y} is a rank-1 matrix constructed from an outer product of Hanning windows and is placed at random positions. Blocking artifacts can be seen in the decomposition without random cycle spinning while vastly diminished in the random cycle spinned decomposition.	30
3.8	Multiscale low rank versus low rank + sparse on faces with uneven illumination. Multiscale low rank decomposition recovers almost shadow-free faces, whereas low rank + sparse decomposition can only remove some shadows.	32

3.9	Multiscale low rank versus low rank + sparse decomposition on a surveillance video. For the multiscale low rank, body motion is mostly captured in the $16 \times 16 \times 16$ scale while fine-scale motion is captured in $4 \times 4 \times 4$ scale. The background video component is captured in the globally low rank component and is almost artifact-free. Low rank + sparse decomposition exhibits ghosting artifacts as pointed by the red arrow because they are neither globally low rank or sparse.	33
3.10	Multiscale low rank versus low rank + sparse decomposition on a dynamic contrast enhanced magnetic resonance image series. For the multiscale result, small contrast dynamics in vessels are captured in 4×4 blocks while contrast dynamics in the liver are captured in 16×16 blocks. The biggest block size captures the static tissues and interestingly the respiratory motion. In contrast, the low rank + sparse modeling could only provide a coarse separation of dynamics and static tissue, which result in neither truly sparse nor truly low rank components.	34
3.11	Multiscale low rank reconstructed matrix of the 100K MovieLens dataset. The extracted signal scale component captures the tendency that younger users rated Star Wars higher whereas the more senior users rated Gone with the Wind higher.	35
4.1	Illustration of the multi-scale low rank matrix model.	49
4.2	Illustration of the algorithm. Alternating minimization is performed stochastically using a random subset of the measurements.	52
4.3	Reconstructions of the MRXCAT phantom using LR, LLR, L+S and the proposed MSLR representations (full-length video at: https://doi.org/10.6084/m9.figshare.7466579). Frames during contrast injection are shown. The proposed method with MSLR shows the least amount of streaking artifacts, compared to other methods as pointed out by the red arrows. Overall, it also achieves the highest PSNR.	56
4.4	Reconstructions of the MRXCAT phantom using LR, LLR, L+S and the proposed MSLR representations (full-length video at: https://doi.org/10.6084/m9.figshare.7466579). Frames after contrast injection are shown, along with the temporal profiles indicated by the blue dashed lines. All methods display certain temporal artifacts, but overall LLR and MSLR show more accurate dynamics. The red arrows point to ghosting artifacts from other frames in all methods except the proposed MSLR. The yellow arrows point to contrast mismatch compared to the ground truth in all methods except LLR. Both LR and L+S also display blurring artifacts in the temporal profiles.	57

- 4.5 High frame-rate and low frame-rate reconstructions of the first DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453178>). In the high frame-rate reconstruction, contrast injection dynamics faster than respiration are nicely shown, such as those in the heart and in the blood vessels. From the video, slight bulk motion, pointed out by the white arrows, was observed when the contrast was injected. Respirator motion was also observed throughout the scan. In the low frame-rate reconstruction, the contrast injection dynamics were all combined into one frame. Because of the averaging effect over time, the contrast amplitude was also much less than the one in the high frame-rate reconstruction. More artifacts can also be seen in the images, even in the post-contrast image, which might be due to the respiratory motion. 58
- 4.6 The MSLR decomposition of the DCE reconstruction shown in Figure 4.5 (full-length video at: <https://doi.org/10.6084/m9.figshare.7453232>). The global scale mainly captures the respiratory motion, whereas finer scales mainly capture localized dynamics such as contrast injection dynamics in the heart and blood vessels. This matches the expectation that the local block-wise bases can capture localized dynamics more compactly than the global block-wise bases. 59
- 4.7 High frame-rate and low frame-rate reconstructions of the second DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453175>). In the high frame-rate reconstruction, the contrast injection dynamic from the left arm blood vessel can be visualized nicely. From the video, the contrast injection also caused a global shift of the subject body, creating severe artifacts afterwards. As soon as the body returned to the original position, the image quality improved, which is also seen in the post-contrast image in the figure. In the low frame-rate reconstruction, all frames with contrast injection dynamics were corrupted by the bulk motion. More artifacts can also be seen in the images, which might be due to the respiratory motion. 60
- 4.8 A respiratory cycle excerpt of the first lung dataset (full length video at: <https://doi.org/10.6084/m9.figshare.7453229>). From the video, irregular breathing with variable rates can be observed. Despite this, the pulmonary structures, such as blood vessels, can be visualized clearly when a cycle is extracted, as seen in the figure. The result matches the modeling assumption that global and recurrent dynamics can be well represented with LR models. At the same time, the proposed reconstruction does not require an explicit estimate of the respiratory signal that is needed in gating reconstruction, and hence is robust to errors associated with it. 61

4.9	A respiratory cycle excerpt of the second lung dataset (full length video at: https://doi.org/10.6084/m9.figshare.7453226). From the video, coughing can be observed throughout the scan, resulting in poor image quality whenever it occurs. However, when the subject returns to a more regular breathing pattern, image quality also improves significantly. This shows that MSLR, though not able to resolve sudden motion like coughing, is able to localize the artifacts in time. Thus, the extracted images from a single respiratory cycle shown in the figure are able to show certain detailed pulmonary structures, such as blood vessels.	61
5.1	Iteration progression for ℓ_1 wavelet regularized reconstruction of a 3D UTE lung dataset. Both FISTA and PDHG exhibit extreme blurring even after 100 iterations. In contrast, PDHG with the proposed preconditioner converges in about ten iterations, both visually and quantitatively in terms of minimizing the objective value.	65
5.2	Diagram of computing the proposed diagonal k-space preconditioner for the first channel.	72
5.3	Iteration progression for ℓ_2 regularized reconstruction of the liver dataset, comparing CG with and without Koolstra et al.'s preconditioning, and PDHG with and without the proposed preconditioning. Both visually and quantitatively in terms of objective value, methods with preconditioning converge faster than the non-preconditioned counterparts. Both preconditioned methods converge in less than ten iterations.	75
5.4	Iteration progression for ℓ_2 -regularized reconstruction of the brain dataset, comparing CG with and without Koolstra et al.s preconditioning, and PDHG with and without the proposed preconditioning	76
5.5	Iteration progression for ℓ_2 -regularized reconstruction of the cardiac dataset, comparing CG with and without Koolstra et al.s preconditioning, and PDHG with and without the proposed preconditioning	77
5.6	Iteration progression for ℓ_1 wavelet regularized reconstruction of the cardiac dataset, comparing FISTA and PDHG with and without the proposed preconditioning. Again, both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about ten iterations.	78
5.7	Iteration progression for ℓ_1 wavelet regularized reconstruction of the brain dataset, comparing FISTA and PDHG with and without the proposed preconditioning.	79
5.8	Iteration progression for ℓ_1 wavelet regularized reconstruction of the liver dataset, comparing FISTA and PDHG with and without the proposed preconditioning.	80
5.9	Iteration progression for total variation regularized reconstruction of the brain dataset, comparing PDHG with and without the proposed preconditioning. Both methods converged slower compared to the two other reconstruction experiments. Both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about 30 iterations.	81

5.10	Iteration progression for total variation regularized reconstruction of the cardiac dataset, comparing PDHG with and without the proposed preconditioning.	82
5.11	Iteration progression for total variation regularized reconstruction of the liver dataset, PDHG with and without the proposed preconditioning.	83
6.1	Illustration of the convolutional sparse model for images.	87
6.2	Illustration of the training algorithm. In each iteration, a data index i is randomly picked in $\{1, \dots, I\}$, and alternating minimization performed.	89
6.3	Illustration of the reconstruction pipeline. When reconstructing new scans, the filters are fixed and only the coefficients are computed. The final image is reconstructed by convolving the coefficients with the filters.	90
6.4	Filters learned from fully-sampled datasets. Compared to wavelet filters, these filters display edge-like features with various orientations, instead of just horizontal and vertical. On the other hand, the filters notably lack smooth filters.	91
6.5	Filters learned from under-sampled datasets. These filters show similar structures as the ones learned from fullysampled datasets, with edge-like and impulse-like features. One difference is that the filters seem less noisy than the ones learned from fully-sampled datasets. This can also be seen in the phase of the filters.	92
6.6	A representative ℓ_1 -regularized reconstructions with wavelet transform, and the learned filters. In general, reconstructions with the dictionaries showed sharper edges, but increased noise artifacts as well.	92
6.7	A box-plot of PSNR difference between the learned reconstructions versus ℓ_1 -wavelet reconstruction over the 640 test slices.	93

List of Tables

5.1	Per-iteration computation time	74
5.2	Computation time for constructing preconditioners	74

Acknowledgments

First, I'd like to thank my advisor, Miki Lustig, for the amazing past seven and a half years. I first met Miki in the undergrad digital signal processing class EE 123. Originally, I was debating whether to take that or robotics. But as soon as I took the first class, I was blown away by Miki's lecture! I was impressed by the clarity, passion and excitement! That semester, I learned a few things about signal processing, and most importantly, I learned that Miki is a great person to work with. He has millions of ideas to try for any problem. He can explain different concepts with utmost clarity. And he always has a great way to put problems in the right perspective. Above all, Miki has the best kind of sense of humor, a quirky one. Since then, I have been working with Miki first as a undergrad, and then as a graduate student. During my PhD, Miki has given me lost of freedom to explore new ideas, which I've always been grateful for. Miki also gives great life advice and has shown me that hiking can sometimes be fun. And of course, because of Miki, I met my wife, Wenwen.

I'd also like to thank Kannan Ramchandran for the many engaging discussions we've had in many years. I met Kannan in an undergrad class EE 121 (Digital communication), and have worked together on sparse FFT algorithms. Kannan is a great teacher and has shown me the beauty of coding theory.

I want to thank Ming Gu, and Chunlei Liu for being in my dissertation and quals committee respectively. From Ming, I've learned how to look at problems through the lens of numerical linear algebra. From Chunlei, I've learned many MRI knowledge that I would get nowhere else, such as the ability to delay echos to effectively get TE \neq TR. Thank you!

I'd also like to thank past senior members in MikGroup: Martin, Anita, and Mariya. I learned how to do recon the right way from Martin. Because Martin's always right. I've always admired Anita, because she's just very cool. Every project she's involved in, she brought in discipline and style. Mariya was a great person to talk with, because every time her laughter would make me very happy.

I want to thank current senior members in MikGroup: Jon, and Zhiyong. I've always been impressed by Jon's ability to speak so clearly and give a talk whenever he wants. I've also learned a lot from our discussions on many technical topics. I'd like to thank Zhiyong for saving my life, when I got food poisoned in Israel. I enjoyed the many meals and home-grown Chinese vegetables I'd had at his place.

I'd also like to thank other ex-members in the group: Joe, Michal, Alan, Teresa, and Dara. I am also lucky enough to meet three of the latest grad students in Mikgroup: Ke, Suma and Ekin in the final semester of my PhD career. Thank you so much for the "low-rank Frank" cake! I hope you'll all have a happy PhD career.

I want to thank my Stanford collaborators: Shreyas, Joseph, Chris, Feiyu, Adam, David, and Neerav, and my UCSF collaborators: Peder Larson and Xucheng Zhu. I want to thank my Tao Zhang and Enhao Gong at Subtle Medical.

Finally, I'd like to thank my wife, Wenwen Jiang, for her support and many other things. Wenwen is infinitely more interesting than I am, but still chooses to be with me. I still can't believe we got married a month ago! Wenwen is my best collaborator both in research and

in life. Her Phd thesis inspires Chapter 4. And I'm continuing to be inspired by her every single day. Thank you!

Chapter 1

Introduction

Magnetic Resonance Imaging (MRI) is an amazing imaging modality in many aspects. It offers one of the best imaging contrast for visualizing soft tissues. It has no ionizing radiation at all. Its flexibility has also enabled many applications, including assessing blood flow, imaging brain activity via oxygenation contrast, and measuring tissue stiffness. Since MRI was invented, this imaging technology has saved numerous lives, and has been the frontier of biomedical and engineering research.

On the other hand, imaging speed remains a main limitation of MRI. Inherently, MRI takes time to collect measurements, and often requires minutes to complete a scan. In this regard, MRI is quite similar to early cameras: Subjects have to be motionless for minutes to obtain an image, which is uncomfortable to patients. This often leads to motion and motion artifacts. When severe motion artifacts occur, scans have to be repeated.

This dissertation aims to change that by developing techniques to reconstruct three-dimensional (3D) dynamic MRI from continuous acquisitions. An ideal 3D dynamic scan would be able to resolve almost all dynamics at a high spatiotemporal resolution. Subjects would not have to be motionless. The comprehensive information in the single scan would also greatly simplify clinical workflow. While this dissertation has not achieved this ideal scan yet, it proposes several innovations toward this goal. In particular, Figure 1.1 (full-length video available at www.doi.org/10.6084/m9.figshare.7464485) shows a 3D rendering of a reconstruction result from Chapter 4. Arbitrary slices at different orientation can be selected over time. From the video, respiratory motion, contrast enhancements, and even slight bulk motion can be seen.

The main challenge in high resolution 3D dynamic MRI is that the reconstruction problem is inherently underdetermined and demanding of computation and memory. To overcome these challenges, this dissertation builds on top of many fundamental methods, described in Chapter 2, including non-Cartesian imaging, parallel imaging and compressed sensing. In particular, this dissertation heavily relies on the compressed sensing framework, which has three components: 1) the image of interest has a compressed signal representation. 2) MRI can acquire (pseudo)-randomized samples in k-space, which provides incoherent encoding of the underlying image. 3) sparsity/compressibility can be efficiently enforced in reconstruction

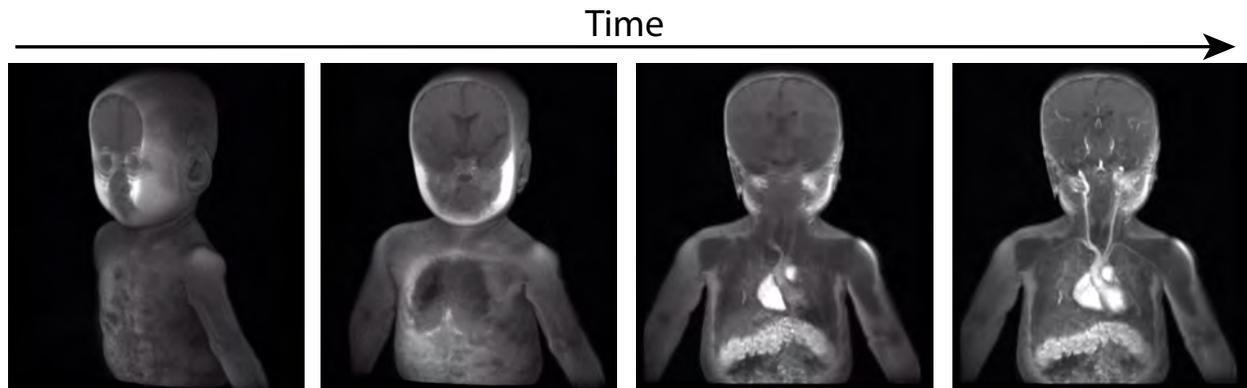


Figure 1.1: A 3D rendering of a result from Chapter 4 (full-length video available at www.doi.org/10.6084/m9.figshare.7464485). Arbitrary slices at different orientation can be selected over time. From the video, respiratory motion, contrast enhancements, and even slight bulk motion can be seen.

to recover the compressed representation from the undersampled measurements.

In Chapter 3, I propose a multiscale low rank model that can compactly represent dynamic image sequences. The resulting representation can be applied beyond MRI, and is useful for other applications, such as motion separation in surveillance video. With the multiscale low rank representation, in Chapter 4, I propose a technique incorporating stochastic optimization to efficiently reconstruct 3D dynamic MRI. This makes it feasible to run such large-scale reconstructions on local workstations. To further speed up the reconstruction time, in Chapter 5, I propose accelerating the convergence of non-Cartesian reconstruction using a specially designed preconditioner. Finally, in Chapter 6, I leverage external undersampled datasets to further improve reconstruction quality using convolutional sparse coding.

1.1 Outline

The structure of this dissertation is as follows:

MRI Reconstruction Overview

In this chapter, I give an overview of MRI reconstruction to understand the following chapters more completely. I start with the basic MRI signal equation and reconstruction. I then describe two commonly used methods for dynamic MRI reconstructions: view sharing and gating. Finally, I describe two essential techniques to reconstruct images from reduced measurements: parallel imaging and compressed sensing.

Multiscale Low Rank Matrix Model

In this chapter, I present a multi-scale low rank modeling that represents a data matrix as a sum of block-wise low rank matrices with increasing scales of block sizes. I then consider the inverse problem of decomposing the data matrix into its multiscale low rank components and approach the problem via a convex formulation. I show that under various incoherence conditions, the convex program recovers the multiscale low rank components either exactly or approximately. I demonstrate its effectiveness in four applications, including illumination normalization for face images, motion separation for surveillance videos, multi-scale modeling of DCE-MRI and collaborative filtering exploiting age information.

High Resolution Volumetric Dynamic MRI from Non-Gated Acquisitions

Leveraging the multiscale low rank model, I describe techniques to reconstruct high spatiotemporal resolution volumetric dynamic images from continuous non-gated acquisition. The problem considered is vastly underdetermined and computational demanding (trying to reconstruct hundreds of gigabytes of an image from a few gigabytes of measurements). To overcome these challenges, I propose three innovations: a compressed representation using multi-scale low rank matrix model to regularize the reconstruction problem, an objective function that directly optimize for the compressed representation to reduce memory usage, and stochastic optimization to reduce computation. I evaluated the proposed method in simulation, dynamic contrast enhanced imaging and lung imaging.

Accelerating Convergence for Non-Cartesian MRI Reconstructions

In this chapter, I propose a k-space preconditioning approach for accelerating the convergence of iterative MRI reconstruction from non-Cartesian k-space data. Existing methods either use sampling density compensation which sacrifices reconstruction accuracy, or circulant preconditioners, which increase per-iteration computation. The proposed approach overcomes both shortcomings. Through experiments, I show that in practice, the proposed method does accelerate convergence for parallel imaging and/or compressed sensing reconstructions.

Learning a Sparse Representation from Many Undersampled Datasets

Learning from existing datasets has the potential to improve reconstruction quality. However, existing machine learning based methods typically require many clean fully-sampled datasets as ground truths. Such datasets can be hard to come by, especially in dynamic imaging applications. In this chapter, I propose a method based on convolutional sparse coding that can learn filters from under-sampled datasets to sparsely represent MR images. I show that the filters learned from under-sampled datasets are similar to the filters learned from fully-sampled datasets, and improves upon wavelet transform for ℓ_1 -regularized reconstruction in terms of mean-squared error.

Summary and Future Work

Finally, I summarize the methods presented in this dissertation and outlines directions for future work.

Chapter 2

MRI Reconstruction Overview

In this chapter, I give an overview of MRI reconstruction to understand the following chapters more completely. I start with the basic MRI signal equation and reconstruction. I then describe two commonly used methods for dynamic MRI reconstructions: view sharing and gating. Finally, I describe two essential techniques to reconstruct images from reduced measurements: parallel imaging and compressed sensing.

For readers who are interested in how to arrive at the signal equation from the underlying physics, I recommend Nishimura's excellent book on Principles of Magnetic Resonance Imaging [1] for more detail.

2.1 Basic MRI Signal Equation

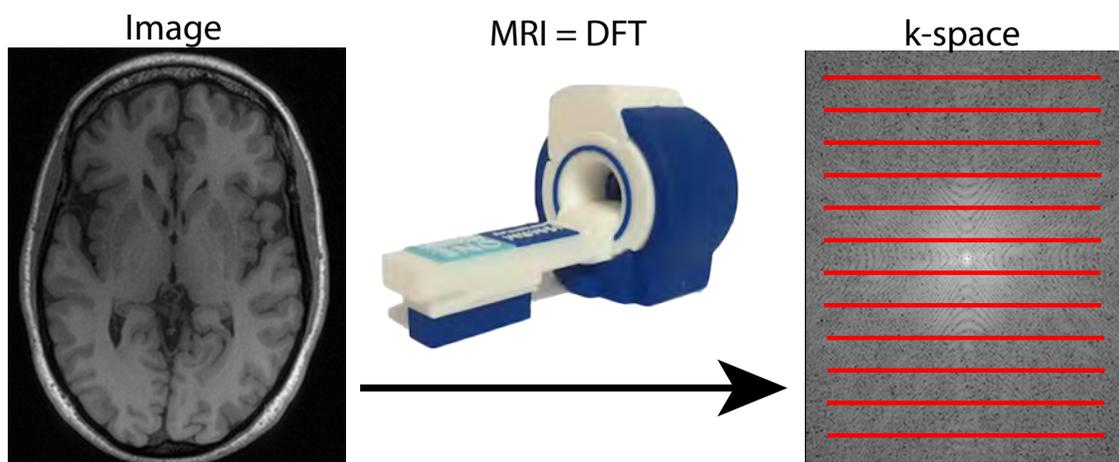


Figure 2.1: Throughout this dissertation, I consider the discrete Fourier approximate formulation of the MRI acquisition process.

Throughout this dissertation, I consider a discrete Fourier approximation of the MRI acquisition process. In particular, under this formulation and given an underlying image $\mathbf{x} \in \mathbb{C}^N$, an MRI scanner outputs the following measurements \mathbf{y} :

$$\mathbf{y}_i = \sum_{j=1}^N \mathbf{x}_j e^{-i2\pi \mathbf{k}_i^\top \mathbf{n}_j} + \mathbf{w}_j \text{ for } i = 1, \dots, M \quad (2.1)$$

where $\mathbf{n} \in \mathbb{R}^{3 \times N}$ and $\mathbf{k} \in \mathbb{R}^{3 \times M}$ represent a set of 3D spatial points and a set of 3D frequency points respectively, and $\mathbf{w} \in \mathbb{C}^M$ is a vector representing white Gaussian noise in the acquisition. A pictorial illustration is shown in Figure 2.1.

Because the spatial frequency points are denoted as \mathbf{k} , MRI acquisition is often described as sampling in “k-space”. Due to physical constraints, these k-space points can only be designed to sample along continuous lines. Each line is acquired one at a time and takes milliseconds to acquire. This results in minutes of acquisition time for 3D imaging.

When acquired on a an equispaced grid, the k-space sampling is called a Cartesian trajectory. Otherwise, it is referred to as non-Cartesian. In general, Cartesian trajectories are easier to reconstruct, but non-Cartesian ones have flexibility in their design. Figure 2.2 shows three commonly used two-dimensional (2D) and three-dimensional (3D) trajectories.

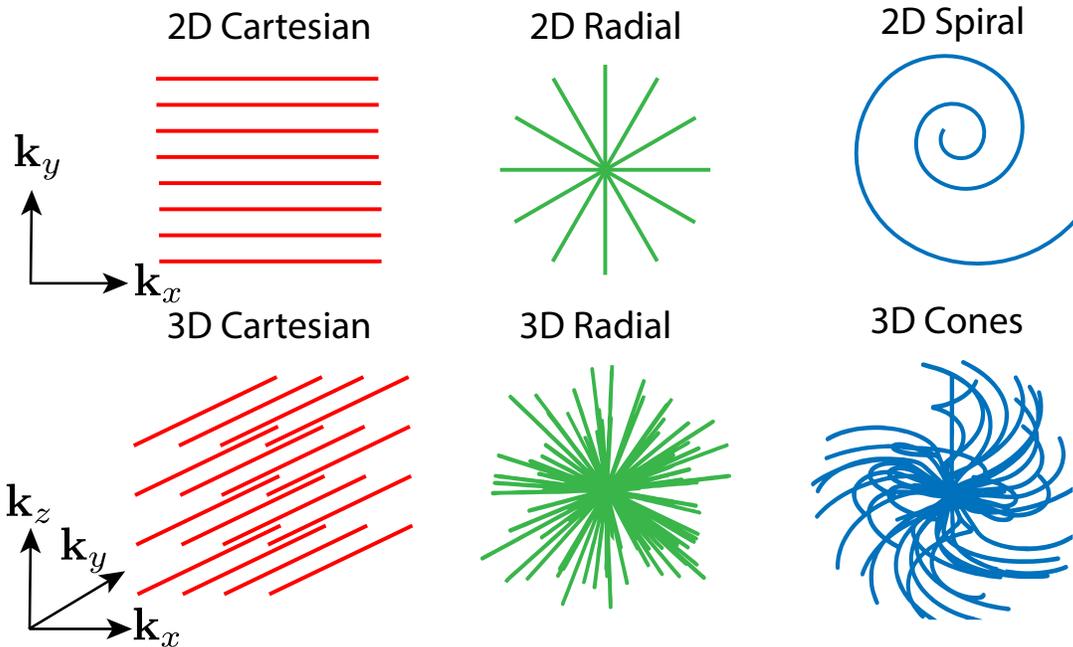


Figure 2.2: Common MRI sampling trajectories: 2D Cartesian, radial and spiral trajectories, and 3D Cartesian, radial, and cones trajectories.

In this dissertation, I often represent the signal equation in matrix form, which can be

succinctly represented as:

$$\mathbf{y} = \mathbf{F}\mathbf{x} + \mathbf{w} \quad (2.2)$$

where $\mathbf{F} \in \mathbb{C}^{M \times N}$ is a discrete Fourier transform operator.

2.2 Basic MRI Reconstruction

The goal of reconstruction is to recover an image $\hat{\mathbf{x}}$, which is as close as possible to the underlying image \mathbf{x} , consistent with the acquired measurements \mathbf{y} . Without additional information, the Fourier operator \mathbf{F} needs to be critical or overdetermined to form a well-posed reconstruction problem. For simplicity, in this section, I assume this is the case.

In particular, for fully-sampled or densely sampled Cartesian trajectory, the Fourier operator \mathbf{F} is unitary, that is $\mathbf{F}^{-1} = \mathbf{F}^H$. An image can be recovered as

$$\hat{\mathbf{x}} = \mathbf{F}^H \mathbf{y} \quad (2.3)$$

where \mathbf{F}^H performs the inverse discrete Fourier transform, and can be computed efficiently using the inverse Fast Fourier transform (FFT).

For non-Cartesian trajectories, the Fourier operator is no longer unitary. A general way to reconstruct is to consider the following problem,

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{F}\mathbf{x} - \mathbf{y}\|_2^2 = (\mathbf{F}^H \mathbf{F})^{-1} \mathbf{F}^H \mathbf{y}. \quad (2.4)$$

which is optimal when the data is corrupted with white Gaussian noise.

A direct matrix inverse is computationally prohibitive as the matrix is dense and the image size is on the order of at least tens of thousands. In practice, the minimization problem is approximated using iterative algorithms, such as the conjugate gradient method.

In addition, the Fourier operator for non-Cartesian trajectories cannot be computed directly using the FFT. Instead, it is approximately computed using the non-uniform Fast Fourier transform (NUFFT) to further speedup reconstruction time. The NUFFT operation is performed by multiplying with an apodization function, performing an oversampled FFT, and interpolating the grid onto the non-Cartesian points using a local interpolation kernel. Throughout this dissertation, I always use the NUFFT to approximately compute the Fourier operator for non-Cartesian trajectories. It is worth noting that the approximation error can be reduced to be arbitrarily small at the expense of additional computation.

However, even with the NUFFT, iterative methods may take many iterations to converge, resulting in long reconstruction time. Hence, a gridding reconstruction is often used to reconstruct an image in a single step, which performs the following operations:

$$\hat{\mathbf{x}} = \mathbf{F}^H \mathbf{D} \mathbf{y} \quad (2.5)$$

where \mathbf{D} is a diagonal matrix containing density compensation factors as diagonals, to reduce image blurring caused by the variable density sampling in non-Cartesian trajectories. In

some sense, the \mathbf{D} operator is equivalent to filtering in filtered back projection. Because of its efficiency and simplicity, gridding reconstruction is often the method of choice for basic non-Cartesian reconstructions, at the cost of reduced image quality and signal-to-noise-ratio.

2.3 View Sharing

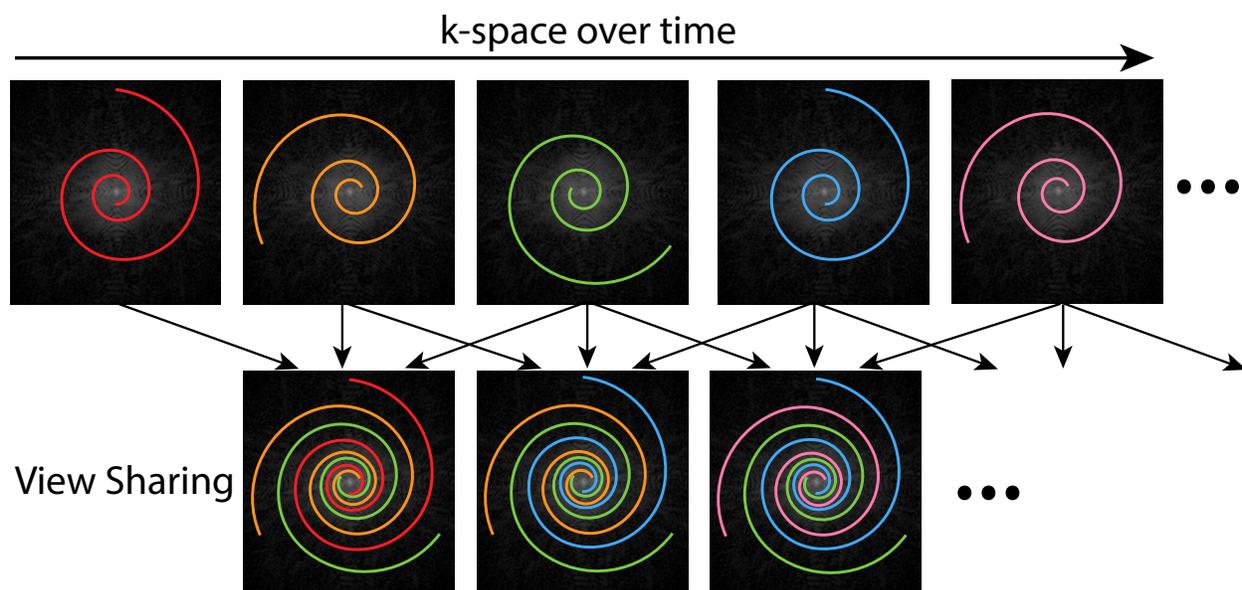


Figure 2.3: View sharing is a technique that fills in the missing k-space using samples from nearby time frames. In particular, this is often done in a sliding window manner.

Since each k-space line takes milliseconds to acquire, high resolution dynamic imaging can only cover k-space partially for each frame. View sharing [2, 3, 4] is a technique that fills in the missing k-space using samples from nearby time frames. In particular, this is often done in a sliding window manner, as shown in Figure 2.3. When the underlying dynamics are smooth, view sharing can often reduce undersampling artifacts, and results in reconstructions with higher signal-to-noise ratio. On the other hand, view sharing inherently blends k-space measurements along time and acts as a low pass filter, which reduces the temporal resolution.

For example, Figure 2.4 shows gridding reconstructions with various levels of view sharing of a real-time 2D cardiac dataset acquired with a spiral trajectory. Compared to no view sharing, view sharing with five interleaves reduce most of the undersampling artifacts. Dynamics over time can be seen. However, when pushed to use 50 interleaves, view sharing results in essentially static frames.

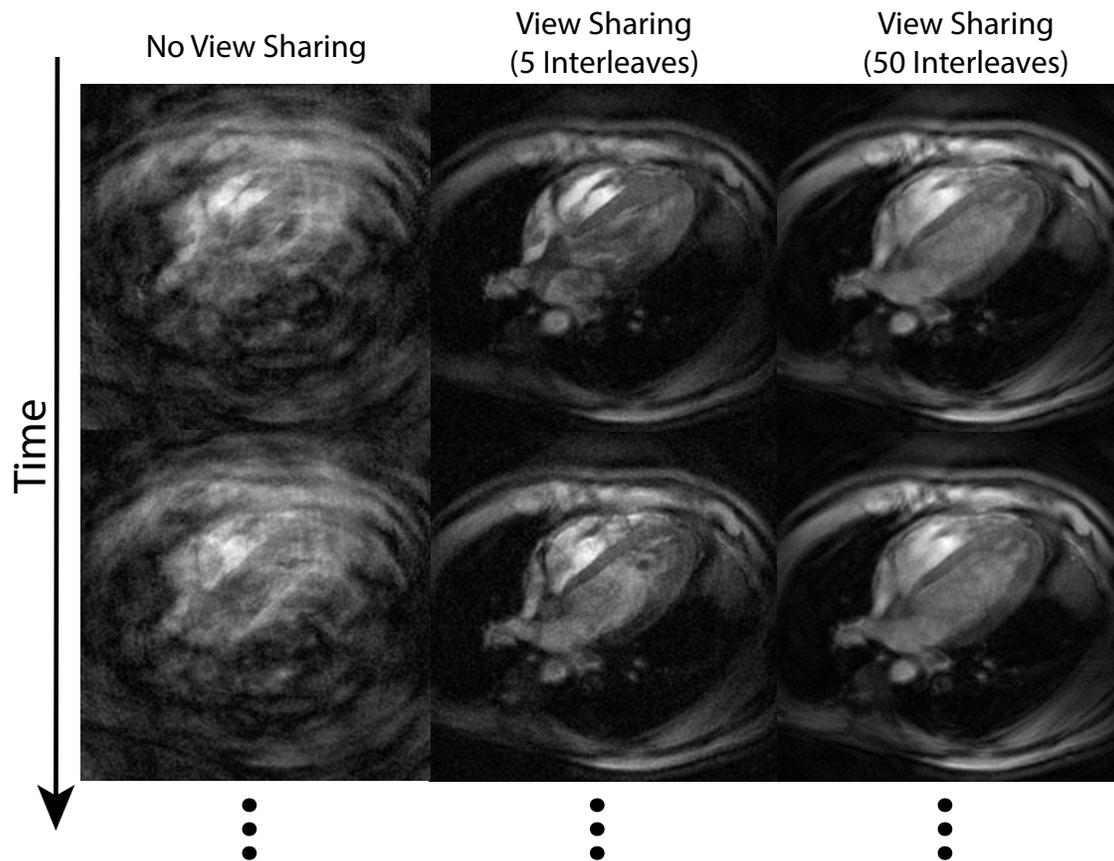


Figure 2.4: Gridding reconstructions with various levels of view sharing of a real-time 2D cardiac dataset, acquired with a spiral trajectory. Compared to no view sharing, view sharing with five interleaves greatly reduce the undersampling artifacts. Dynamics over time can be seen. However, when pushed to use 50 interleaves, view sharing results in essentially static frames.

2.4 Gating and Data Binning

Another class of dynamic image reconstruction technique consists of gating and data binning. These methods exploit the periodicity nature of the underlying dynamics, such as cardiac and respiratory motion. In particular, k -space samples are acquired over many cycles. These measurements are then sorted according to their corresponding phase in the cycle. If the dynamics remain periodic with long enough scan time, the resulting gated k -space for each phase can even be fully-sampled. Figure 2.5 shows an illustration.

Data sorting can be accomplished by leveraging external navigator signals, or navigators derived from the MR data itself, which is often referred to as self-gating. In particular, external navigators can be obtained from an electrocardiogram (ECG) for cardiac, and from bellows for respiratory signals. Self gating navigator signals can be extracted from k -space

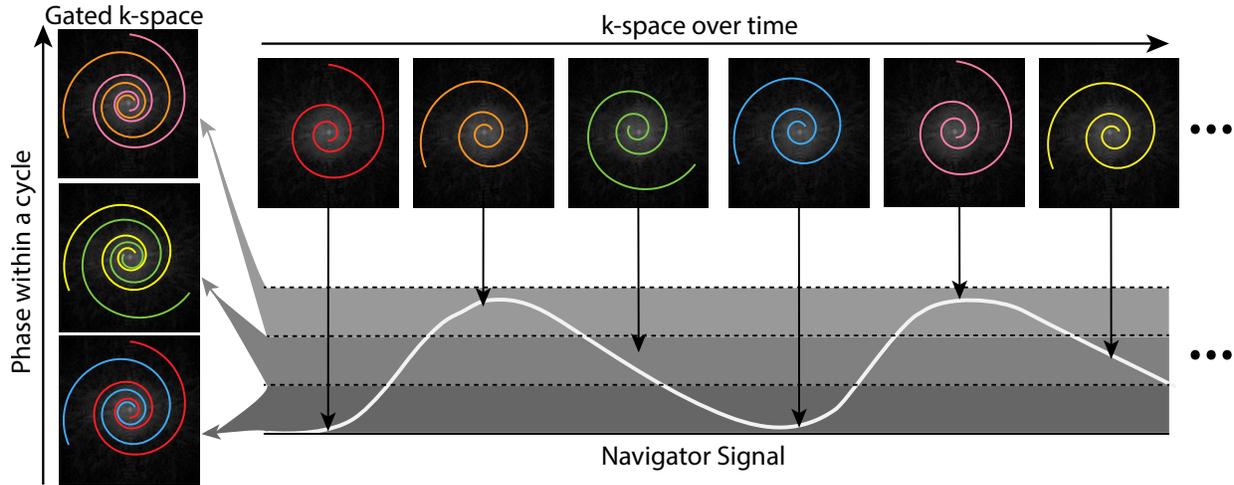


Figure 2.5: Gating and data binning exploit the periodicity nature of the underlying dynamics, such as cardiac and respiratory motion. In particular, k-space samples are acquired over many cycles. Then these measurements are sorted according to their corresponding phase in the cycle. With long enough scan time, the resulting gated k-space for each phase can even be fully sampled.

centers over time, or from a low resolution reconstructed dynamic image sequence [5].

Multiple navigator signals can be used simultaneously for gating and data binning. In particular, in XD-GRASP [6], measurements can be sorted according to both respiratory and cardiac phases. This results in a two-dimensional parametrized space for the underlying dynamics, resolving both cardiac and respiratory motion. Recently, MR multitasking [7] further incorporates this idea and bins measurements in up to five dimensions. These methods are often combined with parallel imaging and compressed sensing, as described in the following sections, to further reduce acquisition time.

The main drawback of gating and data binning methods is that when the periodic assumption does not hold, the reconstruction can break down and exhibit severe artifacts. Incompatible dynamics include bulk motion, coughing, and transient contrast enhancements. For example, Figure 2.6 shows gated reconstructions of the real-time cardiac dataset over six cardiac cycles, with and without simulated bulk motion. To simulate bulk motion, k-space measurements for the first cardiac cycle were rotated 30 degrees counter clock-wise. Motion artifacts can be seen for the motion corrupted image. To go beyond the periodicity assumption, in Chapter 4, I present a reconstruction method for high resolution 3D dynamic MRI from non-gated acquisitions.

2.5 Parallel Imaging

Parallel imaging is now routinely used to reduce scan time by acquiring less k-space data. It utilizes additional spatial encodings provided by multichannel coil arrays to compensate

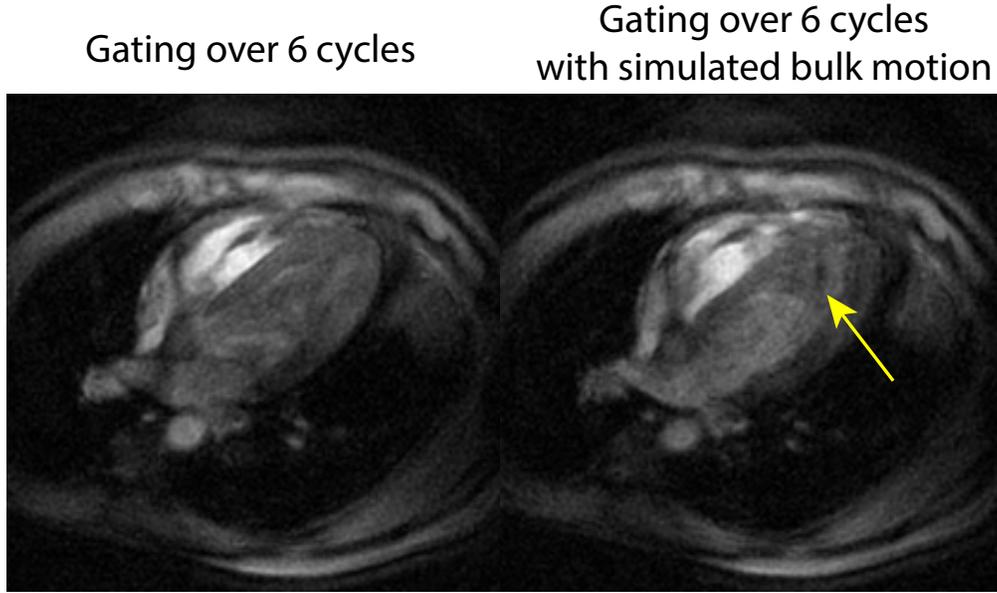


Figure 2.6: Gated reconstructions of the real-time cardiac dataset over six cardiac cycles, with and without simulated bulk motion. To simulate bulk motion, k-space measurements, and hence the underlying image, for the first cardiac cycle were rotated 30 degrees counterclock-wise. Motion artifacts can be seen for the corrupted image.

for undersampling. In particular, each coil element is sensitive to a local region of the underlying object. This can be represented as the underlying image being multiplied with spatially varying coil sensitivity maps. Figure 2.7 illustrates how multichannel coil arrays change the imaging process.

Concretely, with multichannel coil arrays, the signal equation (2.1) becomes

$$\mathbf{y}_{ic} = \sum_{j=1}^N \mathbf{s}_{cj} \mathbf{x}_j e^{-i2\pi \mathbf{k}_i^\top \mathbf{n}_j} + \mathbf{w}_{ic} \text{ for } i = 1, \dots, M, \text{ and } c = 1, \dots, C \quad (2.6)$$

where $\{\mathbf{s}_c \in \mathbb{C}^N\}_{c=1}^C$ represents the multichannel coil sensitivity maps.

In matrix form, this can be represented as:

$$\mathbf{y} = \mathbf{F}\mathbf{S}\mathbf{x} + \mathbf{w} \quad (2.7)$$

where $\mathbf{S} \in \mathbb{C}^{NC \times N}$ is a sensitivity map operator.

Many algorithms have been developed to reconstruct images from undersampled parallel imaging data. Among them, SENSE [8, 9] is a general method that is compatible with arbitrary k-space trajectories, and considers the following reconstruction problem,

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{F}\mathbf{S}\mathbf{x} - \mathbf{y}\|_2^2. \quad (2.8)$$

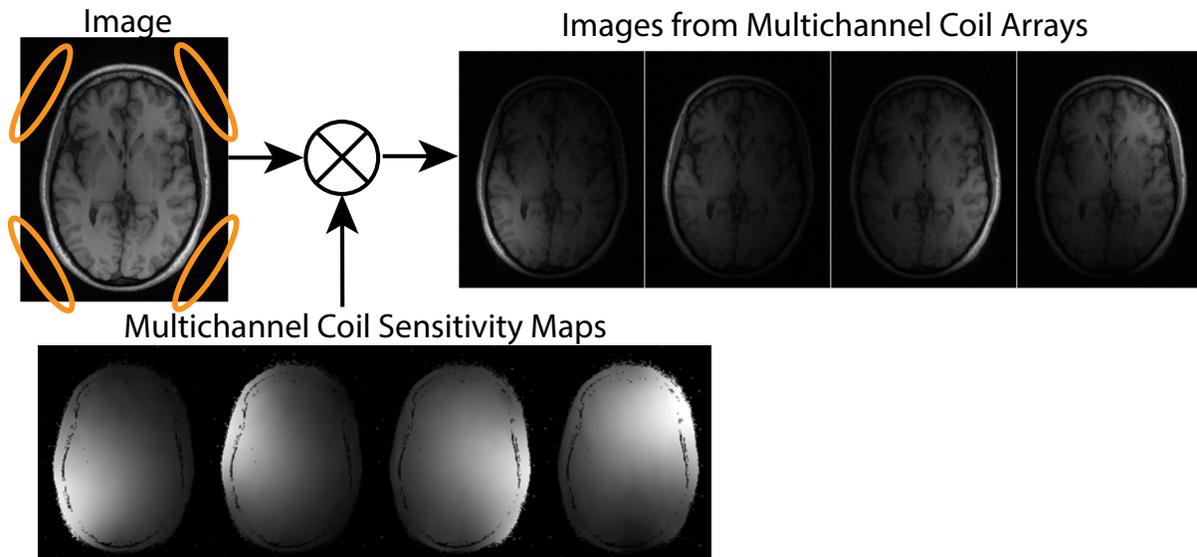


Figure 2.7: Parallel imaging utilizes additional spatial encodings provided by multichannel coil arrays to compensate for undersampling. In particular, each coil element is sensitive to particular region of the underlying object. This effectively multiplies the underlying image with spatially varying coil sensitivity maps.

For Cartesian parallel imaging with uniform sampling, the reconstruction can be decoupled for solving for a few pixels at a time. This can be efficiently solved in a single step. For non-Cartesian parallel imaging, iterative reconstruction is used in practice to reconstruct images.

Similar to the basic non-Cartesian reconstruction, iterative methods for non-Cartesian parallel imaging might take many iterations to converge. Unlike the basic non-Cartesian reconstruction, there are no single step techniques to approximate non-Cartesian parallel imaging reconstruction for arbitrary trajectories. Therefore, long reconstruction time is a limitation to the clinical adoption of non-Cartesian parallel imaging. In Chapter 5, I propose an efficient method to speed up iteration convergence for non-Cartesian reconstruction.

2.6 Compressed Sensing

Compressed sensing offers a separate direction to parallel imaging to accelerate acquisition by exploiting properties of the underlying image. In particular, there are three components for the application of compressed sensing MRI [10]: 1) the image of interest is compressible using a signal transform. 2) MRI can acquire (pseudo)-randomized samples in k-space, which provides incoherent encoding of the underlying image. 3) sparsity/compressibility can be efficiently enforced in reconstruction to recover the compressed representation from the

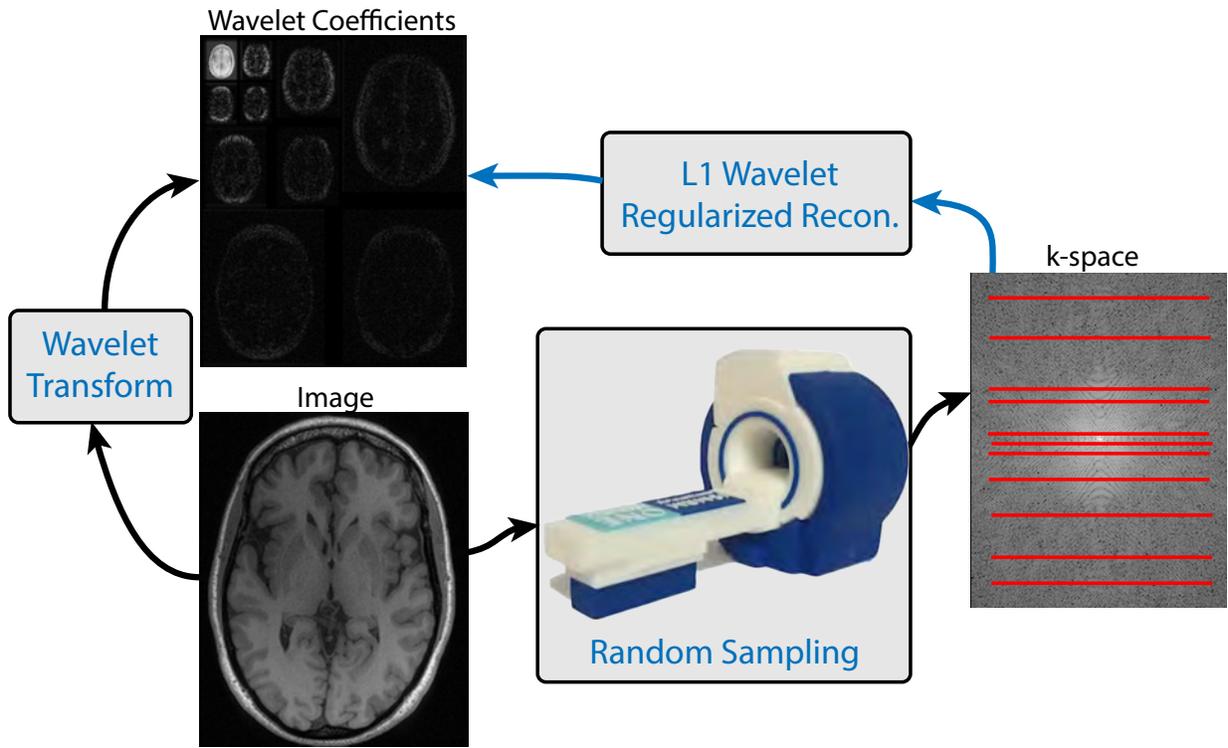


Figure 2.8: Illustration of an example application of compressed sensing MRI. There are three components: 1) The wavelet transform is used to sparsely represent the underlying brain image. 2) Randomized k-space lines are acquired, which provides incoherent encoding of the underlying image. 3) ℓ_1 wavelet regularized reconstruction is used to efficiently recover the compressed representation from the undersampled measurements.

undersampled measurements.

Figure 2.8 illustrates an example compressed sensing MRI application. Here, the wavelet transform is used to sparsely represent the underlying brain image. Randomized k-space lines are acquired, which provide incoherent encoding of the underlying image. Finally, ℓ_1 wavelet regularized reconstruction is used to efficiently recover the compressed representation from the undersampled measurements. In particular, let $\Psi \in \mathbb{C}^{N \times N}$ be a wavelet transform operator, then the ℓ_1 wavelet regularized reconstruction considers the following optimization problem

$$\hat{\mathbf{x}} \in \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{F}\mathbf{S}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|\Psi\mathbf{x}\|_1 \quad (2.9)$$

which can be approximately solved using iterative algorithms, such as the Fast Iterative Soft-Thresholding Algorithm (FISTA) [11].

Each of the three components in compressed sensing MRI can be designed and tailored to specific applications. In Chapter 3, I focus on developing a compressed representation for

dynamic images using the multiscale low rank matrix model. Using this representation, in Chapter 4, I develop an efficient algorithm for 3D dynamic MRI using stochastic optimization. Finally, in Chapter 6, I leverage external undersampled datasets to learn a sparsifying transform using convolutional sparse coding.

Chapter 3

Multiscale Low Rank Matrix Model

3.1 Introduction

Signals and systems often exhibit different structures at different scales. Such multiscale structure has inspired a wide variety of multiscale signal transforms, such as wavelets [12], curvelets [13] and multiscale pyramids [14], that can represent natural signals compactly. Moreover, their ability to compress signal information into a few significant coefficients has made multiscale signal transforms valuable beyond compression and are now commonly used in signal reconstruction applications, including denoising [15], compressed sensing [16, 17], and signal separation [18, 19, 20].

On the other hand, low rank methods are commonly used instead when the signal subspace needs to be estimated as well. In particular, low rank methods have seen great success in applications, such as biomedical imaging [21], face recognition [22] and collaborative filtering [23], to recover the signal subspace and compactly represent the signal at the same time. Recent convex relaxation techniques [24] have further enabled low rank model to be adaptable to various signal processing tasks, including matrix completion [25], system identification [26] and phase retrieval [27].

Here I present a multiscale low rank matrix decomposition method that incorporates multiscale structures with low rank methods. The additional multiscale structure allows me to obtain a more accurate and compact signal representation than conventional low rank methods whenever the signal exhibits multiscale structures (see Figure 3.1). To capture data correlation at multiple scales, I model the data matrix as a sum of block-wise low rank matrices with increasing scales of block sizes (more detail in Section 3.2) and consider the inverse problem of decomposing the matrix into its multiscale components.

I propose a convex formulation to perform the multiscale low rank matrix decomposition, and provide a theoretical analysis in Section 3.5 that extends the rank-sparsity incoherence results in Chandrasekaran et al. [28]. I show that the proposed convex program decomposes the data matrix into its multiscale components exactly under a deterministic incoherence condition. In addition, in Section 3.6, I provide a theoretical analysis on approximate multi-

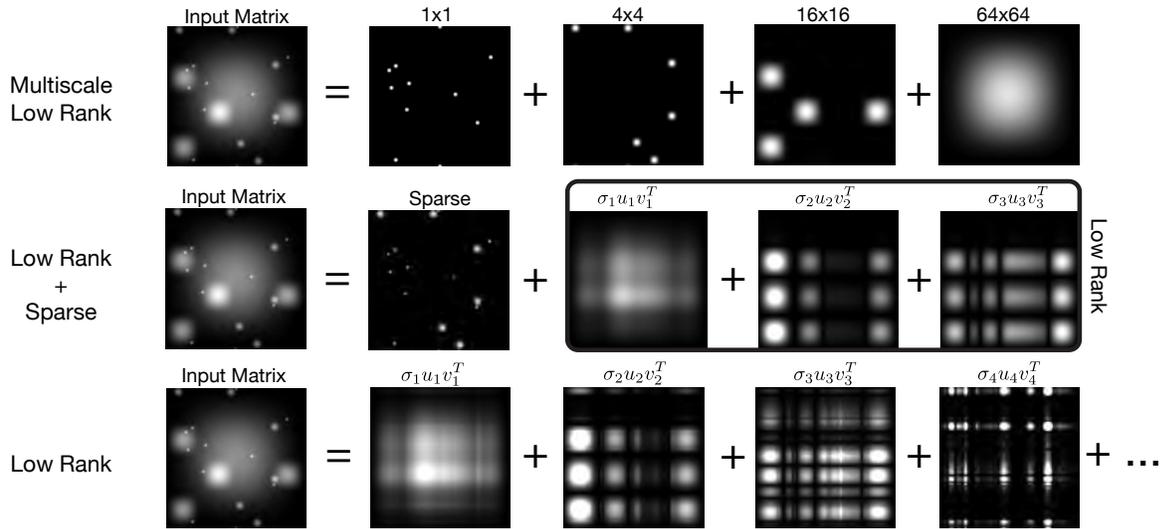


Figure 3.1: An example of the proposed multiscale low rank decomposition compared with other low rank methods. Each blob in the input matrix is a rank-1 matrix constructed from an outer product of hanning windows. Only the multiscale low rank decomposition exactly separates the blobs to their corresponding scales and represents each blob as compactly as possible.

scale low rank matrix decomposition in the presence of additive noise that extends the work of Agarwal et al. [29].

I also investigate the practical performance and applications of the proposed multiscale low rank decomposition. I provide practical guidance on choosing regularization parameters for the convex method in Section 3.4 and describe heuristics to perform cycle spinning [30] to reduce blocking artifacts in Section 3.9. In addition, I applied the multiscale low rank decomposition on real datasets and considered four applications of the multiscale low rank decomposition: illumination normalization for face images, motion separation for surveillance videos, compact modeling of the dynamic contrast enhanced magnetic resonance imaging and collaborative filtering exploiting age information. (See Section 3.10 for more detail). The results show that the proposed multiscale low rank decomposition provides intuitive multiscale decomposition and compact signal representation for a wide range of applications.

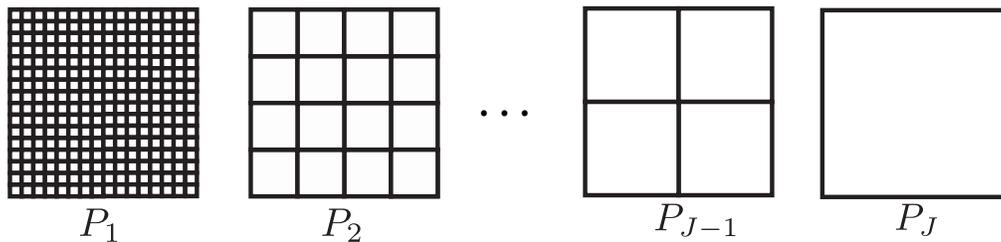
Related work

The proposed multiscale low rank matrix decomposition draws many inspirations from recent developments in rank minimization [24, 29, 31, 32, 25, 33, 34]. In particular, the multiscale low rank matrix decomposition is a generalization of the low rank + sparse decomposition proposed by Chandrasekaran et al. [28] and Candès et al. [35]. The multiscale low rank convex formulation also fits into the convex demixing framework proposed by McCoy et al. [36, 37, 38], who studied the problem of demixing components via convex optimization. The proposed

multiscale low rank decomposition can be viewed as a concrete and practical example of the convex demixing problem. However, their theoretical analysis assumes that each component is randomly oriented with respect to each other, and does not apply to the considered setting, where the direct summation of the components is observed. Bakshi et al. [39] proposed a multiscale principal component analysis by applying principal component analysis on wavelet transformed signals, but such method implicitly constrains the signal to lie on a predefined wavelet subspace. Various multi-resolution matrix factorization techniques [40, 41] were proposed to greedily peel off components of each scale by recursively applying matrix factorization. One disadvantage of these factorization methods is that it is not straightforward to incorporate them with other reconstruction problems as models. Similar multiscale modeling using demographic information was also used in collaborative filtering described in Vozalis and Margaritis [42].

3.2 Multiscale Low Rank Matrix Modeling

Multi-scale Matrix Partition



Multi-scale Low Rank Modeling

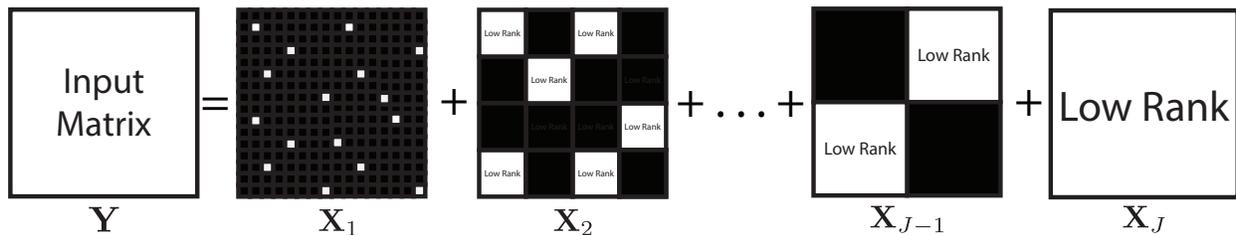


Figure 3.2: Illustration of a multiscale matrix partition and its associated multiscale low rank modeling. Since the zero matrix is a matrix with the least rank, the multiscale modeling naturally extends to sparse matrices as 1×1 low rank matrices.

In this section, I describe the proposed multiscale low rank matrix modeling in detail.

I assume that the data matrix of interest \mathbf{Y} can be partitioned into different scales. Specifically, I assume that I am given a multiscale partition $\{P_i\}_{i=1}^J$ of the indices of an

$M \times N$ matrix, where each block b in P_i is an order magnitude larger than the blocks in the previous scale P_{i-1} . Such multiscale partition can be naturally obtained in many applications. Figures 3.2 and 3.4 provide two examples of a multiscale partition, the first one with decimation along two dimensions and the second one with decimation along one dimension.

To easily transform between the data matrix and the block matrices, I also consider a block reshape operator $\mathcal{B}_b(\mathbf{X})$ that extracts a block b from the full matrix X and reshapes the block into an $m_i \times n_i$ matrix (Figure 3.3), where $m_i \times n_i$ is the i th scale block matrix size determined by the user.

Block reshape operator

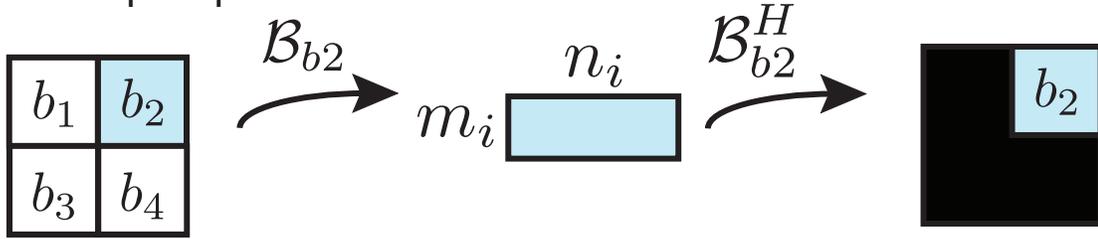


Figure 3.3: Illustration of the block reshape operator \mathcal{B}_b . \mathcal{B}_b extracts block b from the full matrix and reshapes it into an $m_i \times n_i$ matrix. Its adjoint operator \mathcal{B}_b^H takes an $m_i \times n_i$ matrix and embeds it into a full-size zero matrix.

Given an $M \times N$ input matrix \mathbf{Y} and its corresponding multiscale partition and block reshape operators, the multiscale low rank modeling models the $M \times N$ input matrix \mathbf{Y} as a sum of matrices $\sum_{i=1}^J \mathbf{X}_i$, in which each \mathbf{X}_i is block-wise low rank with respect to its partition P_i , that is,

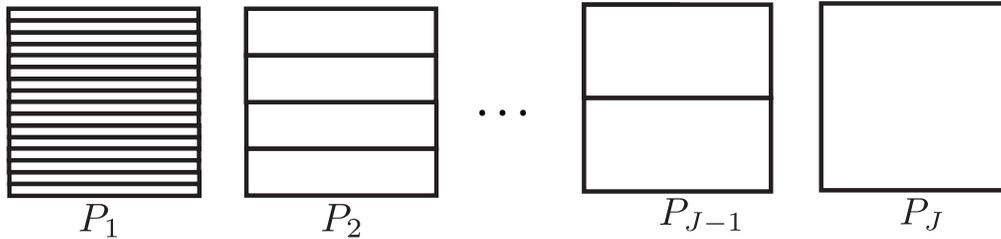
$$\mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i$$

$$\mathbf{X}_i = \sum_{b \in P_i} \mathcal{B}_b^H(\mathbf{U}_b \mathbf{S}_b \mathbf{V}_b^T),$$

where \mathbf{U}_b , \mathbf{S}_b , and \mathbf{V}_b are matrices with sizes $m_i \times \text{rank}_b$, $\text{rank}_b \times \text{rank}_b$ and $n_i \times \text{rank}_b$ respectively and form the rank_b reduced SVD of $\mathcal{B}_b(\mathbf{X}_i)$. Note that when the rank of the block matrix $\mathcal{B}_b(\mathbf{X}_i)$ is zero, I have $\{\mathbf{U}_b, \mathbf{S}_b, \mathbf{V}_b\}$ as empty matrices, which do not contribute to \mathbf{X}_i . Figure 3.2 and 3.4 provide illustrations of two kinds of modeling with their associated partitions.

By constraining each block matrices to be of low rank, the multiscale low rank modeling captures the notion that some nearby entries are more similar to each other than global entries in the data matrix. I note that the multiscale low rank modeling is a generalization of the low rank + sparse modeling proposed by Chandrasekaren et al. [28] and Candès et

Multi-scale Matrix Partition



Multi-scale Low Rank Modeling

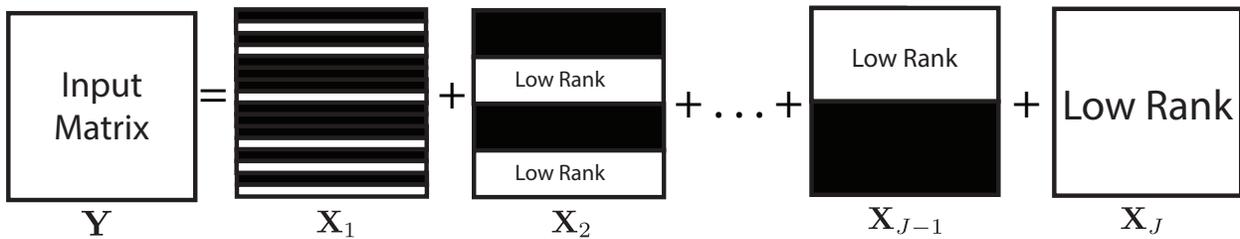


Figure 3.4: Illustration of another multiscale matrix partition and its associated multiscale low rank modeling. Here, only the vertical dimension of the matrix is decimated. Since a $1 \times N$ matrix is low rank if and only if it is zero, the multiscale modeling naturally extends to group sparse matrices.

al. [35]. In particular, the low rank + sparse modeling can be viewed as a 2-scale low rank modeling, in which the first scale has block size 1×1 and the second scale has block size $M \times N$. By adding additional scales between the sparse and globally low rank matrices, the multiscale low rank modeling can capture locally low rank components that would otherwise need many coefficients to represent for low rank + sparse.

Given a data matrix \mathbf{Y} that fits the multiscale low rank model, the goal is to decompose the data matrix \mathbf{Y} to its multiscale components $\{\mathbf{X}_i\}_{i=1}^J$. The ability to recover these multiscale components is beneficial for many applications and enables, for example, motion extraction at multiple scales in surveillance videos (Section 3.10). Since there are many more parameters to be estimated than the number of observations, it is necessary to impose conditions on \mathbf{X}_i . In particular, I will exploit the fact that each block matrix is low rank via a convex program, which will be described in detail in Section 3.3.

Multiscale low rank + noise

Before moving to the convex formulation, I note that the multiscale matrix modeling can easily account for data matrices that are corrupted by additive white Gaussian noise. Under the multiscale low rank modeling, the additive noise matrix can be thought as the largest scale signal component and is unstructured in any local scales. Specifically, if I observe

instead

$$\mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i + \mathbf{X}_Z$$

where \mathbf{X}_Z is an independent and identically distributed Gaussian noise matrix, then I can define a reshape operator \mathcal{B}_Z that reshapes the entire matrix into an $MN \times 1$ vector and the resulting matrix fits exactly to the multiscale low rank model with $J + 1$ scales. This incorporation of noise makes the model flexible in that it automatically provides a corresponding convex relaxation, a regularization parameter for the noise matrix and allows me to utilize the same iterative algorithm to solve for the noise matrix. Figure 3.5 provides an example of the noisy multiscale low rank matrix decomposition.

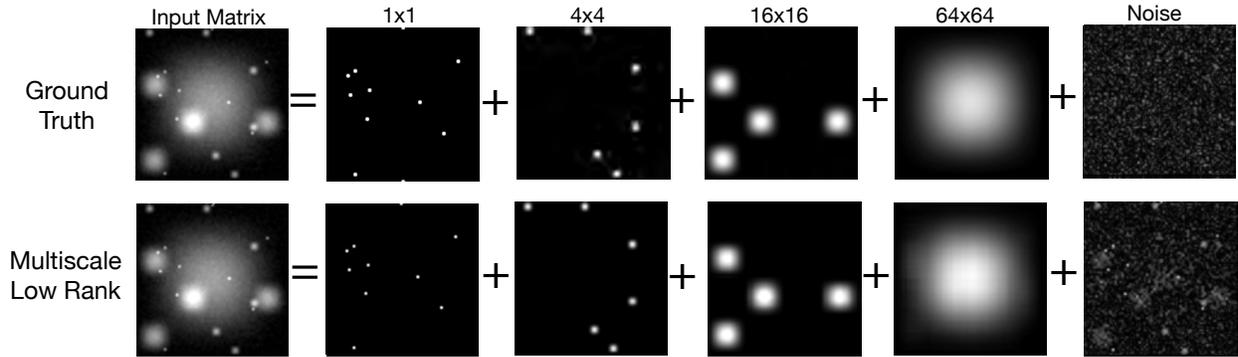


Figure 3.5: An example of the multiscale low rank decomposition in the presence of additive Gaussian noise by solving the convex program (3.1).

3.3 Problem Formulation and Convex Relaxation

Given a data matrix \mathbf{Y} that fits the multiscale low rank model, my goal is to recover the underlying multiscale components $\{\mathbf{X}_i\}_{i=1}^J$ using the fact that \mathbf{X}_i is block-wise low rank. Ideally, I would like to obtain a multiscale decomposition with the minimal block matrix rank and solve a problem similar to the following form:

$$\begin{aligned} & \underset{X_1, \dots, X_J}{\text{minimize}} && \sum_{i=1}^J \sum_{b \in P_i} \text{rank}(\mathcal{B}_b(\mathbf{X}_i)) \\ & \text{subject to} && \mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i \end{aligned}$$

However, each rank minimization for each block is combinatorial in nature. In addition, it is not obvious whether the direct summation of ranks is a correct formulation as a 1-sparse

matrix and a rank-1 matrix should intuitively not carry the same cost. Hence, the above non-convex problem is not a practical formulation to obtain the multiscale decomposition.

Recent development in convex relaxations suggests that rank minimization problems can often be relaxed to a convex program via nuclear norm relaxation [24, 33], while still recovering the optimal solution to the original problem. In particular, Chandrasekaran et al. [28] and Candès et al., [35] showed that a low rank + sparse decomposition can be relaxed to a convex program by minimizing a nuclear norm + ℓ_1 -norm objective as long as the signal constituents are incoherent with respect to each other. In addition, Candès et al., [35] showed that the regularization parameters for sparsity and low rank should be related by the square root of the matrix size. Hence, there is hope that, along the same line, I can perform the multiscale low rank decomposition exactly via a convex formulation.

Concretely, let me define $\|\cdot\|_{\text{nuc}}$ to be the nuclear norm, the sum of singular values, and $\|\cdot\|_{\text{msv}}$ be the maximum singular value norm. For each scale i , I consider the block-wise nuclear norm to be the convex surrogate for the block-wise ranks and define $\|\cdot\|_{(i)}$ the block-wise nuclear norm for the i th scale as

$$\|\cdot\|_{(i)} = \sum_{b \in P_i} \|\mathcal{B}_b(\cdot)\|_{\text{nuc}}.$$

Its associated dual norm $\|\cdot\|_{(i)}^*$ is then given by

$$\|\cdot\|_{(i)}^* = \max_{b \in P_i} \|\mathcal{B}_b(\cdot)\|_{\text{msv}},$$

which is the maximum of all block-wise maximum singular values.

I then consider the following convex relaxation:

$$\begin{aligned} & \underset{X_1, \dots, X_J}{\text{minimize}} && \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} \\ & \text{subject to} && \mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i, \end{aligned} \tag{3.1}$$

where $\{\lambda_i\}_{i=1}^J$ are the regularization parameters and their selection will be described in detail in Section 3.4.

The proposed convex formulation is a natural generalization of the low rank + sparse convex formulation [28, 35]. With the two-sided matrix partition (Fig. 3.2), the nuclear norm applied to the 1×1 blocks becomes the element-wise ℓ_1 -norm and the norm for the largest scale is the nuclear norm. With the one-sided matrix partition (Fig. 3.4), the nuclear norm applied to $1 \times N$ blocks becomes the group-sparse norm and can be seen as a generalization of the group sparse + low rank decomposition [31]. If I incorporate additive Gaussian noise in the model as described in Section 3.2, then I have a nuclear norm applied to an $MN \times 1$ vector, which is equivalent to the Frobenius norm.

One would hope that the theoretical conditions from low rank + sparse can be generalized rather seamlessly to the multiscale counterpart. Indeed, in Section 3.5, I show that the core theoretical guarantees in the work of Chandrasekaran et al. [28] on exact low rank + sparse decomposition can be generalized to the multiscale setting. In section 3.6, I show that the core theoretical guarantees in the work of Agarwal et al. [29] on noisy matrix decomposition can be generalized to the multiscale setting as well to provide approximate decomposition guarantees.

3.4 Guidance on Choosing Regularization Parameters

In this section, I provide practical guidance on selecting the regularization parameters $\{\lambda_i\}_{i=1}^J$. Selecting the regularization parameters $\{\lambda_i\}_{i=1}^J$ is crucial for the convex decomposition to succeed, both theoretically and practically. While theoretically, I can establish criteria on selecting the regularization parameters (see Section 3.5 and 3.6), such parameters are not straightforward to calculate in practice as it requires properties of the signal components $\{\mathbf{X}_i\}_{i=1}^J$ before the decomposition.

To select the regularization parameters $\{\lambda_i\}_{i=1}^J$ in practice, I follow the suggestions from Wright et al. [43] and Fogel et al. [44], and set each regularization parameter λ_i to be the Gaussian complexity of each norm $\|\cdot\|_{(i)}$, which is defined as the expectation of the dual norm of random Gaussian matrix:

$$\lambda_i \sim E[\|\mathbf{G}\|_{(i)}^*] \quad (3.2)$$

where \sim denotes equality up to some constant and \mathbf{G} is a unit-variance independent and identically distributed random Gaussian matrix.

The resulting expression for the Gaussian complexity is the maximum singular value of a random Gaussian matrix, which has been studied extensively Bandeira and Handel [45]. The recommended regularization parameter for scale i is given by

$$\lambda_i \sim \sqrt{m_i} + \sqrt{n_i} + \sqrt{\log(\min(m_i, n_i)|P_i|)} \quad (3.3)$$

For the sparse matrix scale with 1×1 block size, $\lambda_i \sim \sqrt{\log(MN)}$ and for the globally low rank scale with $M \times N$ block size, $\lambda_i \sim \sqrt{M} + \sqrt{N}$. Hence this regularization parameter selection is consistent with the ones recommended for low rank + sparse decomposition by Candès et al. [35], up to a log factor. In addition, for the noise matrix with $MN \times 1$ block size, $\lambda_i \sim \sqrt{MN}$, which has similar scaling as in square root LASSO [46]. In practice, I found that the suggested regularization parameter selection allows exact multiscale decomposition when the signal model is matched (for example Figure 3.1) and provides visually intuitive decomposition for real datasets.

For approximate multiscale low rank decomposition in the presence of additive noise, some form of theoretical guarantees for the regularization selection can be found in my analysis in Section 3.6. In particular, I show that if the regularization parameters λ_i is larger

than the Gaussian complexity of $\|\cdot\|_{(i)}^*$ in addition to some “spikiness” parameters, then the error between recovered decomposition and the ground truth $\{\mathbf{X}_i\}_{i=1}^J$ is bounded by the block-wise matrix rank.

3.5 Theoretical Analysis for Exact Decomposition

In this section, I provide a theoretical analysis of the proposed convex formulation and show that if $\{\mathbf{X}_i\}_{i=1}^J$ satisfies a deterministic incoherence condition, then the proposed convex formulation (3.1) recovers $\{\mathbf{X}_i\}_{i=1}^J$ from \mathbf{Y} exactly.

My analysis follows similar arguments taken by Chandrasekaran et al. [28] on low rank + sparse decomposition and generalizes them to the proposed multiscale low rank decomposition. Before showing the main result (Theorem 3.5.1), I first describe the subgradients of the objective function (Section 3.5) and define a coherence parameter in terms of the block-wise row and column spaces (Section 3.5).

Subdifferentials of the block-wise nuclear norms

To characterize the optimality of the convex problem, I first look at the subgradients of the objective function. I recall that for any matrix X with $\{\mathbf{U}, \mathbf{S}, \mathbf{V}\}$ as its reduced SVD representation, the subdifferential of $\|\cdot\|_{\text{nuc}}$ at X is given by [33, 47],

$$\partial\|\mathbf{X}\|_{\text{nuc}} = \{\mathbf{U}\mathbf{V}^H + \mathbf{W} : \mathbf{W} \text{ and } \mathbf{X} \text{ have orthogonal row/column spaces, } \|\mathbf{W}\|_{\text{msv}} \leq 1\}.$$

Now recall that I define the block-wise nuclear norm to be $\|\cdot\|_{(i)} = \sum_{b \in P_i} \|\mathcal{B}_b(\cdot)\|_{\text{nuc}}$. Then using the chain rule and the fact that $\mathcal{B}_b(\mathbf{X}_i) = \mathbf{U}_b \mathbf{S}_b \mathbf{V}_b^H$, I obtain an expression for the subdifferential of $\|\cdot\|_{(i)}$ at \mathbf{X}_i as follows:

$$\partial\|\mathbf{X}_i\|_{(i)} = \left\{ \begin{array}{l} \sum_{b \in P_i} \mathcal{B}_b^H(\mathbf{U}_b \mathbf{V}_b^H + \mathbf{W}_b) : \\ \mathbf{W}_b \text{ and } \mathcal{B}_b(\mathbf{X}_i) \text{ have orthogonal row/column spaces, } \|\mathbf{W}_b\|_{\text{msv}} \leq 1 \end{array} \right\}$$

To simplify the notation, I define $\mathbf{E}_i = \sum_{b \in P_i} \mathcal{B}_b^H(\mathbf{U}_b \mathbf{V}_b^H)$ and T_i to be a vector space that contains matrices with the same block-wise row spaces or column spaces as \mathbf{X}_i , that is,

$$T_i = \left\{ \sum_{b \in P_i} \mathcal{B}_b^H(\mathbf{U}_b \mathbf{X}_b^H + \mathbf{Y}_b \mathbf{V}_b^H) : \mathbf{X}_b \in \mathbb{C}^{n_i \times \text{rank}_b}, \mathbf{Y}_b \in \mathbb{C}^{m_i \times \text{rank}_b} \right\}$$

where $m_i \times n_i$ is the size of the block matrices for scale i and rank_b is the matrix rank for block b .

Then, the subdifferential of each $\|\cdot\|_{(i)}$ at \mathbf{X}_i can be compactly represented as,

$$\partial\|\mathbf{X}_i\|_{(i)} = \{\mathbf{E}_i + \mathbf{W}_i : \mathbf{W}_i \in T_i^\perp \text{ and } \|\mathbf{W}_i\|_{(i)}^* \leq 1\}.$$

I note that E_i can be thought of as the “sign” of the matrix \mathbf{X}_i , pointing toward the principal components, and, in the case of the sparse scale, is exactly the sign of the entries.

In the rest of the section, I will be interested in projecting a matrix X onto T_i , which can be performed with the following operation:

$$\mathcal{P}_{T_i}(\mathbf{X}) = \sum_{b \in P_i} \mathcal{B}_b^H (\mathbf{U}_b \mathbf{U}_b^H \mathcal{B}_b(\mathbf{X}) + \mathcal{B}_b(\mathbf{X}) \mathbf{V}_b \mathbf{V}_b^H - \mathbf{U}_b \mathbf{U}_b^H \mathcal{B}_b(\mathbf{X}) \mathbf{V}_b \mathbf{V}_b^H).$$

Similarly, to project a matrix X onto the orthogonal complement of T_i , I can apply the following operation:

$$\mathcal{P}_{T_i^\perp}(\mathbf{X}) = \sum_{b \in P_i} \mathcal{B}_b^H ((\mathbf{I} - \mathbf{U}_b \mathbf{U}_b^H) \mathcal{B}_b(\mathbf{X}) (\mathbf{I} - \mathbf{V}_b \mathbf{V}_b^H))$$

where \mathbf{I} is an appropriately sized identity matrix.

Incoherence

Following Chandrasekaren et al. [28], I consider a deterministic measure of incoherence through the block-wise column and row spaces of \mathbf{X}_i . Concretely, I define the coherence parameter for the j th scale signal component \mathbf{X}_j with respect to the i th scale to be the following:

$$\mu_{ij} = \max_{\mathbf{N}_j \in T_j, \|\mathbf{N}_j\|_{(j)}^* \leq 1} \|\mathbf{N}_j\|_{(i)}^*$$

Using μ_{ij} as a measure of incoherence, I can quantitatively say that the j th scale signal component is incoherent with respect to the i th scale if μ_{ij} is small. In the case of low rank + sparse, Chandrasekaren et al. [28] provides an excellent description of the concepts behind the coherence parameters. I refer the reader to their paper for more detail.

Main Result

Given the above definition of incoherence, the following theorem states the main result for exact multiscale low rank decomposition:

Theorem 3.5.1. *If I can choose regularization parameters $\{\lambda_i\}_{i=1}^J$ such that*

$$\sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i} < \frac{1}{2}, \quad \text{for } i = 1, \dots, J$$

then $\{\mathbf{X}_i\}_{i=1}^J$ is the unique optimizer of the proposed convex problem (3.1).

In particular when the number of scales $J = 2$, the condition on $\{\mu_{12}, \mu_{21}\}$ reduces to $\mu_{12}\mu_{21} < 1/4$ and the condition on $\{\lambda_1, \lambda_2\}$ reduces to $2\mu_{12} < \lambda_1/\lambda_2 < 1/(2\mu_{21})$, which is in similar form as Theorem 2 in Chandrasekaren et al. [28].

The proof for the above theorem is given in Section 3.12.

3.6 Theoretical Analysis for Approximate Decomposition

In this section, I provide a theoretical analysis for approximate multiscale low rank decomposition when the measurement is corrupted by additive noise as described in Section 3.2. The result follows arguments from Agarwal et al. [29] on noisy 2-scale matrix decomposition and extends it to the multiscale setting.

Instead of using the incoherence parameter μ_{ij} defined for the exact decomposition analysis in Section 3.5, I opt for a weaker characterization of incoherence between scales for approximate decomposition, studied in Agarwal et al. [29]. Concretely, I consider spikiness parameters α_{ij} between the j th signal component \mathbf{X}_j and i th scale norm $\|\cdot\|_{(i)}$ such that,

$$\alpha_{ij} = \|\mathbf{X}_j\|_{(i)}^*$$

for each $j \neq i$. Hence, if α_{ij} is small, I say \mathbf{X}_j is not spiky with respect to the i th norm.

For analysis purpose, I also impose the constraints $\|\mathbf{X}_j\|_{(i)}^* \leq \alpha_{ij}$ in the convex program. That is, I consider the solution from the following convex program:

$$\begin{aligned} & \underset{\mathbf{X}_1, \dots, \mathbf{X}_J, \mathbf{X}_Z}{\text{minimize}} && \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} + \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} \\ & \text{subject to} && \mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i + \mathbf{X}_Z \\ & && \|\mathbf{X}_j\|_{(i)}^* \leq \alpha_{ij} \quad \text{for } j \neq i \end{aligned} \tag{3.4}$$

I emphasize that the additional constraints $\|\mathbf{X}_j\|_{(i)}^* \leq \alpha_{ij}$ are imposed only for the purpose of theoretical analysis and are not imposed in the experimental results. In particular, for the simulation example in Figure 3.5, the minimizer of the convex program (3.1), using the recommended regularization parameters in Section 3.4, satisfied the constraints even when the constraints were not imposed.

Let me define $\{\Delta_i\}_{i=1}^J$ and Δ_Z to be the errors between the ground truth components $\{\mathbf{X}_i\}_{i=1}^J$ and \mathbf{X}_Z and the minimizers of convex program (3.4). Then, equivalently, I can denote $\{\mathbf{X}_i + \Delta_i\}_{i=1}^J$ and $\mathbf{X}_Z + \Delta_Z$ as the minimizers of the convex program (3.4). The following theorem states the main result for approximate decomposition.

Theorem 3.6.1. *If I choose $\{\lambda_i\}_{i=1}^J$ such that*

$$\lambda_i \geq 2\lambda_Z \frac{\|\mathbf{X}_Z\|_{(i)}^*}{\|\mathbf{X}_Z\|_{\text{fro}}} + \sum_{j \neq i} 2\alpha_{ij}$$

and λ_Z such that

$$\lambda_Z \geq \sqrt{64 \sum_{i=1}^J \lambda_i^2 \sum_{b \in P_i} \text{rank}_b},$$

then the error is bounded by

$$\sum_{i=1}^J \|\Delta_i\|_{fro} \lesssim \frac{\|\mathbf{X}_Z\|_{fro}}{\lambda_Z} \sum_{i=1}^J \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b}$$

where \lesssim denotes inequality up to a universal constant.

Hence, when the spikiness parameters are negligible and $\mathbf{X}_Z = \sigma \mathbf{G}$, where \mathbf{G} is an independent, identically distributed Gaussian noise matrix with unit variance and σ is the noise standard deviation, choosing $\lambda_Z \sim E[\|\mathbf{G}\|_{fro}] \sim \sqrt{MN}$ and $\lambda_i \sim E[\|\mathbf{G}\|_{(i)}^*] \sim \sqrt{m_i} + \sqrt{n_i} + \sqrt{\log(\min(m_i, n_i)|P_i|)}$ ensures the condition is satisfied with high probability. This motivates the recommended regularization selection in Section 3.4.

The proof for the above theorem is given in Section 3.13 and follows arguments from Agarwal et al. [29] on noisy matrix decomposition and Belloni et al. [46] on square root LASSO.

3.7 An Iterative Algorithm for Solving the Multiscale Low Rank Decomposition

In the following, I derive an iterative algorithm that solves for the multiscale low rank decomposition via the Alternating Direction of Multiple Multipliers (ADMM) [48]. While the proposed convex formulation (3.1) can be formulated into a semi-definite program, first-order iterative methods are commonly used when solving for large datasets for their computational efficiency and scalability. A conceptual illustration of the algorithm is shown in Figure 3.6.

To formally obtain update steps using ADMM, I first formulate the problem into the standard ADMM form with two separable objectives connected by an equality constraint,

$$\begin{aligned} & \underset{\mathbf{X}_i, \mathbf{Z}_i}{\text{minimize}} && \mathbf{I} \left\{ \mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i \right\} + \sum_{i=1}^J \lambda_i \|\mathbf{Z}_i\|_{(i)} \\ & \text{subject to} && \mathbf{X}_i = \mathbf{Z}_i \end{aligned} \quad (3.5)$$

where $\mathbf{I}\{\cdot\}$ is the indicator function.

To proceed, I then need to obtain the proximal operators [49] for the two objective functions $\mathbf{I}\{\mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i\}$ and $\sum_{i=1}^J \lambda_i \|\mathbf{Z}_i\|_{(i)}$. For the data consistency objective $\mathbf{I}\{\mathbf{Y} = \sum_{i=1}^J \mathbf{X}_i\}$, the proximal operator is simply the projection operator to the set. To obtain the proximal operator for the multiscale nuclear norm objective $\sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)}$, I first recall that the proximal operator for the nuclear norm $\|\mathbf{X}\|_{\text{nuc}}$ with parameter λ is given by the singular value soft-threshold operator [33],

$$\text{SVT}_\lambda(\mathbf{X}) = \mathbf{U} \max(\boldsymbol{\Sigma} - \lambda \mathbf{I}, \mathbf{0}) \mathbf{V}^H \quad (3.6)$$

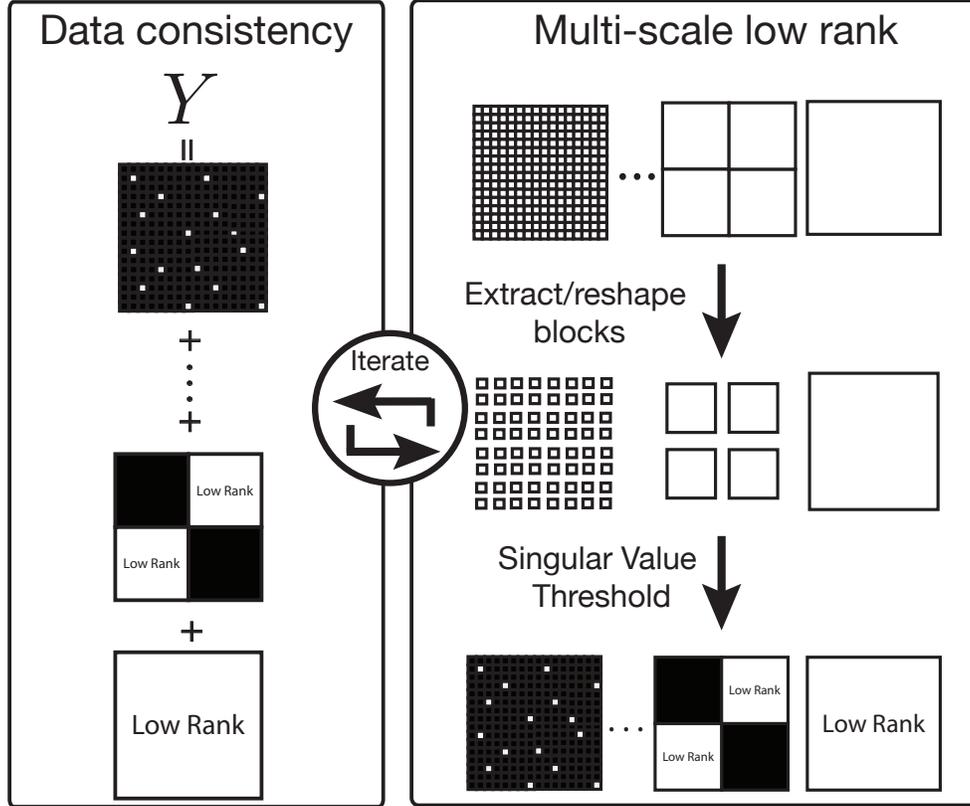


Figure 3.6: A conceptual illustration of how to obtain a multiscale low rank decomposition. First, I extract each block from the input matrix and perform a thresholding operation on its singular value to recover the significant components. Then, I subtract these significant components from the input matrix, thereby enabling the recovery of weaker, previously submerged components.

Since I defined the block-wise nuclear norm for each scale i as $\sum_{b \in P_i} \|\mathcal{B}_b(\cdot)\|_{\text{nuc}}$, the norm is separable with respect to each block and its proximal function with parameter λ_i is given by the block-wise singular value soft-threshold operator,

$$\text{BLOCKSVT}_{\lambda_i}(\mathbf{X}) = \sum_{b \in P_i} \mathcal{B}_b^H(\text{SVT}_{\lambda_i}(\mathcal{B}_b(\mathbf{X}))) \quad (3.7)$$

which simply extracts every blocks in the matrix, performs singular value thresholding and puts the blocks back to the matrix. I note that for 1×1 blocks, the block-wise singular value soft-threshold operator reduces to the element-wise soft-threshold operator and for $1 \times N$ blocks, the block-wise singular soft-threshold operator reduces to the joint soft-threshold operator.

Putting everything together and invoking the ADMM recipe [48], I have the following

algorithm to solve the convex multiscale low rank decomposition (3.1):

$$\begin{aligned}
\mathbf{X}_i &\leftarrow (\mathbf{Z}_i - \mathbf{U}_i) + \frac{1}{L} \left(\mathbf{Y} - \sum_{i=1}^J (\mathbf{Z}_i - \mathbf{U}_i) \right) \\
\mathbf{Z}_i &\leftarrow \text{BLOCKSVT}_{\lambda_i/\rho}(\mathbf{X}_i + \mathbf{U}_i) \\
\mathbf{U}_i &\leftarrow \mathbf{U}_i - (\mathbf{Z}_i - \mathbf{X}_i)
\end{aligned} \tag{3.8}$$

where ρ is the ADMM parameter that only affects the convergence rate of the algorithm.

The resulting ADMM update steps are similar in essence to the intuitive update steps in Figure 3.6, and alternates between data consistency and enforcing multiscale low rank. The major difference of ADMM is that it adds a dual update step with \mathbf{U}_i , which bridges the two objectives and ensures the convergence to the optimal solution. Under the guarantees of ADMM, in the limit of iterations, \mathbf{X}_i and \mathbf{Z}_i converge to the optimal solution of the convex program (3.1) and \mathbf{U}_i converges to a scaled version of the dual variable. In practice, I found that ~ 1000 iterations are sufficient without any visible change for imaging applications. Finally, I note that because the proximal operator for the multiscale nuclear norm is computationally simple, other proximal operator based algorithms [49] can also be used.

3.8 Computational Complexity

Given the iterative algorithm (3.8), one concern about the multiscale low rank decomposition might be that it is significantly more computationally intensive than other low rank methods as I have many more SVD's and variables to compute for. In this section, I show that because I decimate the matrices at each scale geometrically, the theoretical computational complexity of the multiscale low rank decomposition is similar to other low rank decomposition methods, such as the low rank + sparse decomposition.

For concreteness, let me consider the multiscale partition with two-sided decimation shown in Figure 3.2 and have block sizes $m_i = 2^{i-1}$ and $n_i = 2^{i-1}$. Similar to other low rank methods, the SVD's dominate the per iteration complexity for the multiscale low rank decomposition. For an $M \times N$ matrix, each SVD costs $\#\text{flops}(M \times N \text{ SVD}) = O(MN^2)$. The per iteration complexity for the multiscale low rank decomposition is dominated by the summation of all the SVD's performed for each scale, which is given by,

$$\begin{aligned}
&\#\text{flops}(M \times N \text{ SVD}) + 4 \#\text{flops}(M/2 \times N/2 \text{ SVD}) + \dots \\
&= O(MN^2) + O(MN^2)/2 + O(MN^2)/4 + \dots \\
&\leq 2O(MN^2) \approx \#\text{flops}(M \times N \text{ SVD})
\end{aligned} \tag{3.9}$$

Hence, the per-iteration computational complexity of the multiscale low rank with two-sided decimated partition is on the order of a $M \times N$ matrix SVD. In general, one can show

that the per-iteration complexity for arbitrary multiscale partition is at most $\log(N)$ times the full matrix SVD.

While theoretically, the computation cost for small block sizes should be less than bigger block sizes, I found that in practice the computation cost for computing the small SVD's can dominate the per-iteration computation. This is due to the overhead of copying small block matrices and calling library functions repeatedly to compute the SVD's.

Since I am interested in thresholding the singular values and in practice many of the small block matrices are zero as shown in Section 3.10, one trick of reducing the computation time is to quickly compute an upper bound on the maximum singular value for block matrices before the SVD's. Then if the upper bound for the maximum singular value is less than the threshold, I know the thresholded matrix will be zero and can avoid computing the SVD. Since for any matrix X , its maximum singular value is bounded by the square root of any matrix norm on $X^H X$ [50], there are many different upper bounds that I can use. In particular, I choose the maximum row norm and consider the following upper bound,

$$\sigma_{\max}(\mathbf{X}) \leq \sqrt{\max_i \sum_j |\mathbf{X}_{ik} \mathbf{X}_{jk}|} \quad (3.10)$$

Using this upper bound, I can identify many below-the-threshold matrices before computing the SVD's at all. In practice, I found that the above trick provides a modest speedup of $3 \sim 5\times$.

3.9 Heuristics for translation invariant decomposition

Similar to wavelet transforms, one drawback of the multiscale low rank decomposition is that it is not translation invariant, that is, shifting the input changes the resulting decomposition. In practice, this translation variant nature often creates blocking artifacts near the block boundaries, which can be visually jarring for image or video applications. One solution to remove these artifacts is to introduce overlapping partitions of the matrix so that the overall algorithm is translation invariant. However, this vastly increases both memory and computation, especially for large block sizes. In the following, I describe a cycle spinning approach to reduce the blocking artifacts with an only slight increase in per-iteration computation.

Cycle spinning [30] is commonly used in wavelet denoising to reduce the blocking artifacts due to the translation variant nature of the wavelet transform. To minimize artifacts, cycle spinning averages the denoised results from all possible shifted copies of the input, thereby making the entire process translation invariant. Concretely, let S be the set of all shifts possible in the target application, SHIFT_s denote the shifting operator by s , and DENOISE be the denoising operator of interest. Then the cycle spinned denoising of the input X is given by:

$$\frac{1}{|S|} \sum_{s \in S} \text{SHIFT}_{-s}(\text{DENOISE}(\text{SHIFT}_s(\mathbf{X}))) \quad (3.11)$$

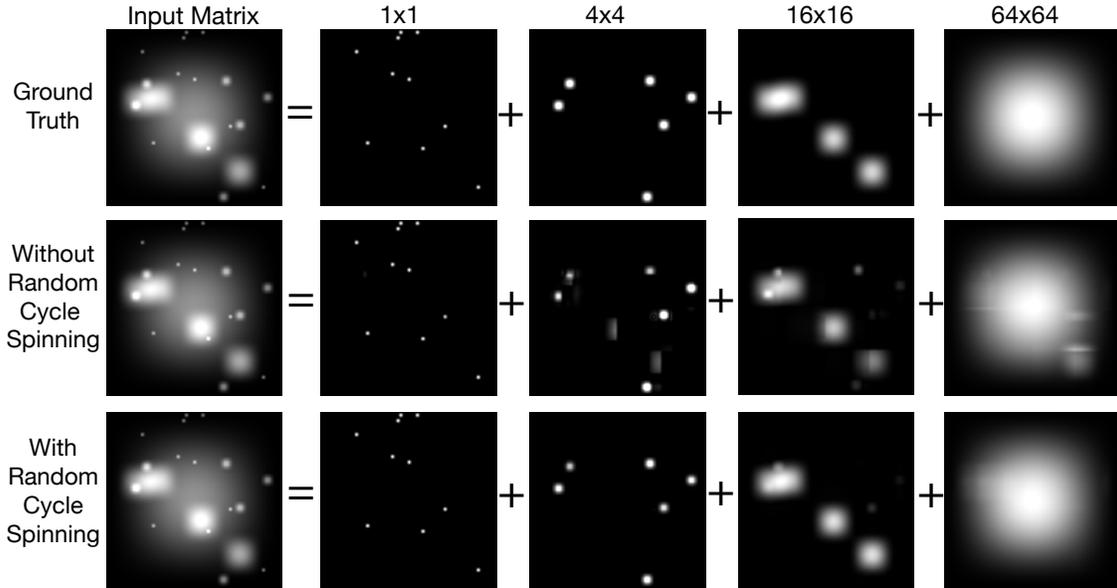


Figure 3.7: An example of the multiscale low rank decomposition with and without random cycle spinning. Each blob in the input matrix \mathbf{Y} is a rank-1 matrix constructed from an outer product of Hanning windows and is placed at random positions. Blocking artifacts can be seen in the decomposition without random cycle spinning while vastly diminished in the random cycle spinned decomposition.

In the context of multiscale low rank decomposition, I can make the iterative algorithm translation invariant by replacing the block-wise singular value thresholding operation in each iteration with its cycle spinning counterpart. In particular, for the ADMM update steps, I can replace the \mathbf{Z}_i step to:

$$\mathbf{Z}_i \leftarrow \frac{1}{|S|} \sum_{s \in S} \text{SHIFT}_{-s}(\text{BLOCKSVT}_{\lambda_i/\rho}(\text{SHIFT}_s(\mathbf{X}_i + \mathbf{U}_i))) \quad (3.12)$$

To further reduce computation, I perform random cycle spinning in each iteration as described in Figueiredo et al. [51], in which I randomly shift the input, perform block-wise singular value thresholding and then unshift back:

$$\mathbf{Z}_i \leftarrow \text{SHIFT}_{-s}(\text{BLOCKSVT}_{\lambda_i/\rho}(\text{SHIFT}_s(\mathbf{X}_i + \mathbf{U}_i))) \quad (3.13)$$

where s is randomly chosen from the set S .

Using random cycle spinning, blocking artifacts caused by thresholding are averaged over iterations. Figure 3.7 shows an example of the multiscale low rank decomposition with and without random cycle spinning applied on a simulated data that does not fall on the partitioned grid. The decomposition with random cycle spinning vastly reduces blocking artifacts that appeared in the one without random cycle spinning.

3.10 Applications

In the spirit of reproducible research, I provide a software package (in C and partially in MATLAB) to reproduce most of the results described in this chapter. The software package can be downloaded from:

https://github.com/frankong/multi_scale_low_rank.git

To test for practical performance, I applied the multiscale low rank decomposition on four different real datasets that are conventionally used in low rank modeling: illumination normalization for face images (Section 3.10), motion separation for surveillance videos (Section 3.10), multiscale modeling of dynamic contrast enhanced magnetic resonance imaging (Section 3.10) and collaborative filtering exploiting age information (Section 3.10). I compared the proposed multiscale low rank decomposition with low rank + sparse decomposition for the first three applications and with low rank matrix completion for the last application. Randomly cycle spinning was used for multiscale low rank decomposition for all experiments. Regularization parameters λ_i were chosen exactly as $\sqrt{m_i} + \sqrt{n_i} + \sqrt{\log(\min(m_i, n_i)|P_i|)}$ for multiscale low rank and $\max(m_i, n_i)$ for low rank + sparse decomposition. The simulations were implemented using the C programming language and run on a 20-core Intel Xeon workstation.

Multiscale Illumination Normalization for Face Recognition

Face recognition algorithms are sensitive to shadows or occlusions on faces. In order to obtain the best possible performance for these algorithms, it is desired to remove illumination variations and shadows on the face images. Low rank modeling is often used to model faces and is justified by approximating faces as convex Lambertian surfaces [22].

Low rank + sparse decomposition [35] was recently proposed to capture uneven illumination as sparse errors and was shown to remove small shadows while capturing the underlying faces as the low rank component. However, most shadows are not sparse and contain structure over different lighting conditions. Here, I propose modeling shadows and illumination changes in different face images as block-low rank as illumination variations are spatially correlated in multiple scales.

I considered face images from the Yale B face database [52]. Each face image was of size 192×168 with 64 different lighting conditions. The images were then reshaped into a $32,256 \times 64$ matrix and both multiscale low rank and low rank + sparse decomposition were applied on the data matrix. For low rank + sparse decomposition, I found that the best separation result was obtained when each face image was normalized to the maximum value. For multiscale low rank decomposition, the original unscaled image was used. Only the space dimension was decimated as I assumed there was no ordering in different illumination conditions. The multiscale matrix partition can be visualized as in Figure 3.4.

Figure 3.8 shows one of the comparison results. Multiscale low rank decomposition recovered almost shadow-free faces. In particular, the sparkles in the eyes were represented

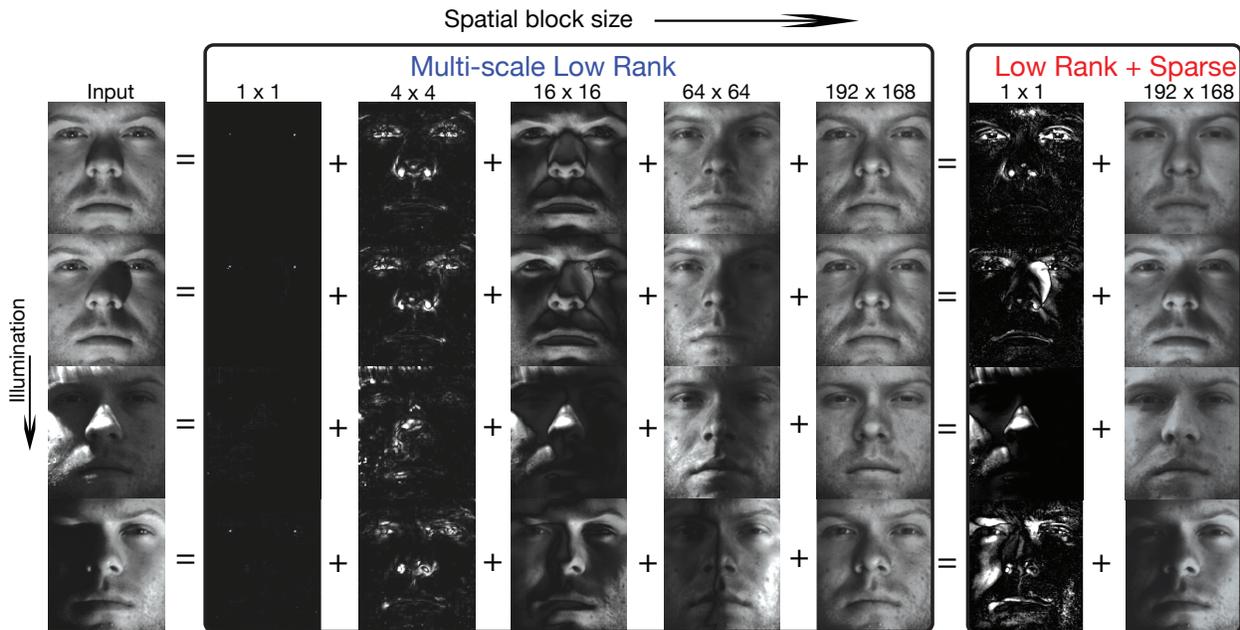


Figure 3.8: Multiscale low rank versus low rank + sparse on faces with uneven illumination. Multiscale low rank decomposition recovers almost shadow-free faces, whereas low rank + sparse decomposition can only remove some shadows.

in the 1×1 block size and the larger illumination changes were represented in bigger blocks, thus capturing most of the uneven illumination changes. In contrast, low rank + sparse decomposition could only recover from small illumination changes and still contained the larger shadows in the globally low rank component.

Multiscale Motion Separation for Surveillance Videos

In surveillance video processing, it is desired to extract foreground objects from the video. To be able to extract foreground objects, both the background and the foreground dynamics have to be modeled. Low rank modeling has been shown to be suitable for slowly varying videos, such as background illumination changes. In particular, if the video background only changes its brightness over time, then it can be represented as a rank-1 matrix.

Low rank + sparse decomposition [35] was proposed to foreground objects as sparse components and was shown to separate dynamics from background components. However, sparsity alone cannot capture motion compactly and often results in ghosting artifacts occurring around the foreground objects as shown in Figure 3.9. Since video dynamics are correlated locally at multiple scales in space and time, I propose using the multiscale low rank modeling with two sided decimation to capture different scales of video dynamics over space and time.

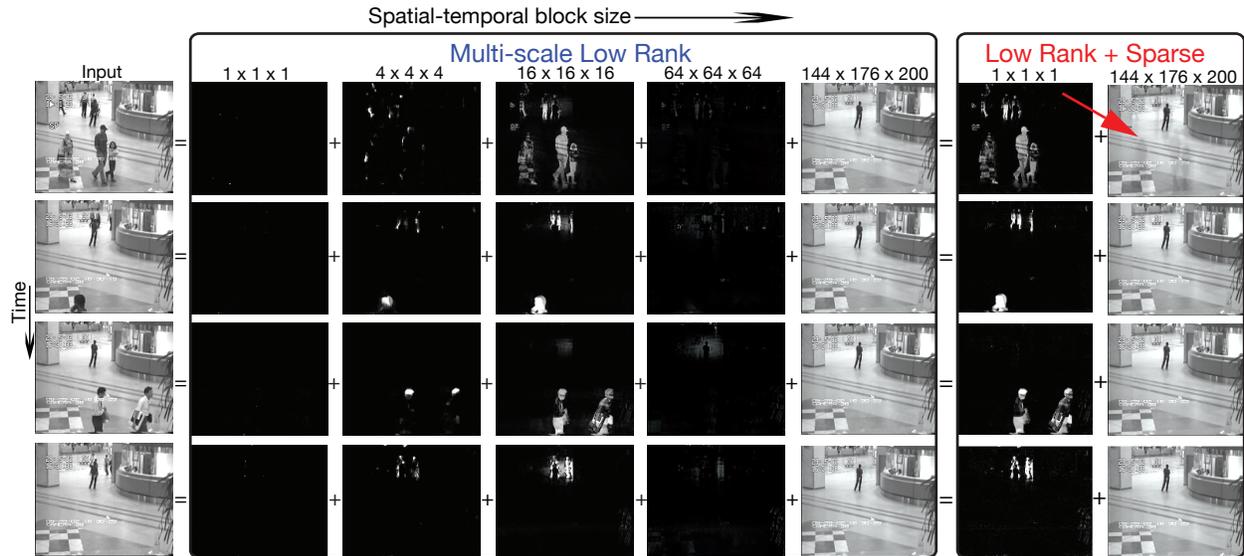


Figure 3.9: Multiscale low rank versus low rank + sparse decomposition on a surveillance video. For the multiscale low rank, body motion is mostly captured in the $16 \times 16 \times 16$ scale while fine-scale motion is captured in $4 \times 4 \times 4$ scale. The background video component is captured in the globally low rank component and is almost artifact-free. Low rank + sparse decomposition exhibits ghosting artifacts as pointed by the red arrow because they are neither globally low rank or sparse.

I considered a surveillance video from Li et al. [53]. Each video frame was of size 144×176 and the first 200 frames were used. The video frames were then reshaped into a $25,344 \times 200$ matrix and both multiscale low rank and low rank + sparse decomposition were applied on the data matrix.

Figure 3.9 shows one of the results. Multiscale low rank decomposition recovered a mostly artifact free background video in the globally low rank component whereas low rank + sparse decomposition exhibits ghosting artifact in certain segments of the video. For the multiscale low rank decomposition, body motion was mostly captured in the $16 \times 16 \times 16$ scale while the fine-scale motion was captured in $4 \times 4 \times 4$ scale.

Multiscale Low Rank Modeling for Dynamic Contrast Enhanced Magnetic Resonance Imaging

In dynamic contrast enhanced magnetic resonance imaging (DCE-MRI), a series of images over time is acquired after a T_1 contrast agent was injected into the patient. Different tissues then exhibit different contrast dynamics over time, thereby allowing radiologists to characterize and examine lesions. Compressed sensing Magnetic Resonance Imaging [10] is now a popular research approach used in three dimensional DCE-MRI to speed up the acquisition. Since the more compact I can represent the image series, the better the compressed

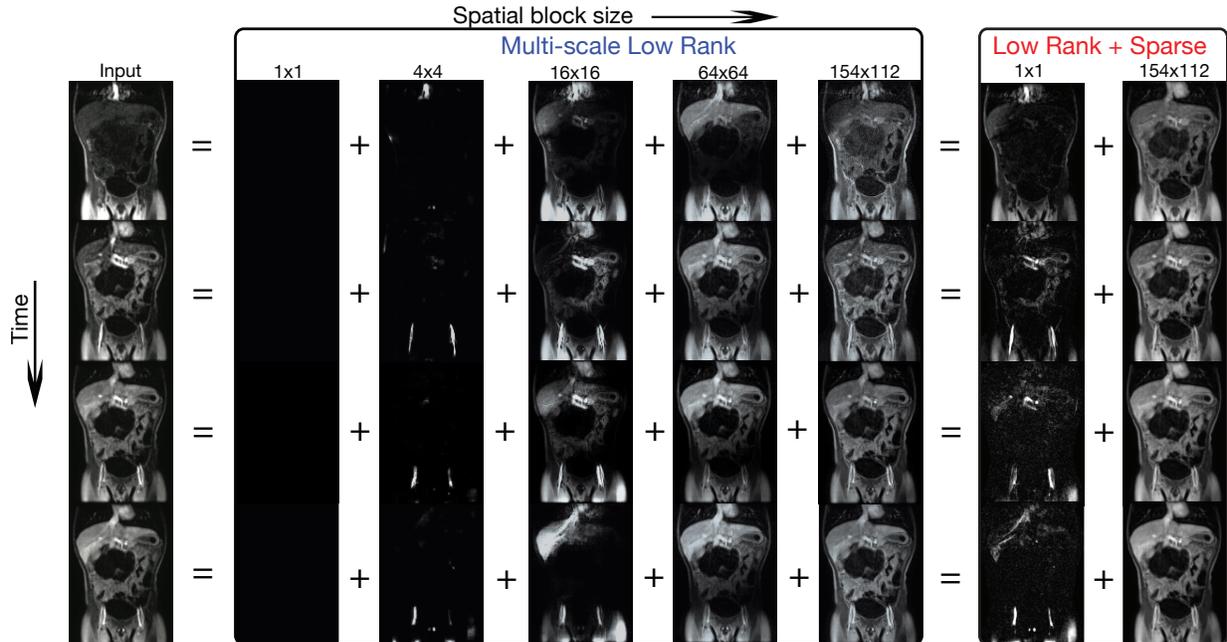


Figure 3.10: Multiscale low rank versus low rank + sparse decomposition on a dynamic contrast enhanced magnetic resonance image series. For the multiscale result, small contrast dynamics in vessels are captured in 4×4 blocks while contrast dynamics in the liver are captured in 16×16 blocks. The biggest block size captures the static tissues and interestingly the respiratory motion. In contrast, the low rank + sparse modeling could only provide a coarse separation of dynamics and static tissue, which result in neither truly sparse nor truly low rank components.

reconstruction result becomes, an accurate modeling of the dynamic image series is desired to improve the compressed sensing reconstruction results for DCE-MRI.

When a region contains only one type of tissue, then the block matrix constructed by stacking each frame as columns will have rank one. Hence, low rank modeling [21], and locally low rank modeling [54] have been popular models for DCE-MRI. Recently, low rank + sparse modeling [55] has also been proposed to model the static background and dynamics as low rank and sparse matrices respectively. However, dynamics in DCE-MRI are almost never sparse and often exhibit correlation across different scales. Hence, I propose using a multiscale low rank modeling to capture contrast dynamics over multiple scales.

I considered a fully sampled dynamic contrast enhanced image data. The data was acquired in a pediatric patient with 20 contrast phases, $1 \times 1.4 \times 2 \text{ mm}^3$ resolution, and 8 s temporal resolution. The acquisition was performed on a 3T GE MR750 scanner with a 32-channel cardiac array using an RF-spoiled gradient-echo sequence. I considered a 2D slice of size 154×112 were then reshaped into a 17248×20 matrix. Both multiscale low rank and low rank + sparse decomposition were applied to the data matrix.

Figure 3.10 shows one of the results. In the multiscale low rank decomposition result,

small contrast dynamics in vessels were captured in 4×4 blocks while contrast dynamics in the liver were captured in 16×16 blocks. The biggest block size captured the static tissues and interestingly the respiratory motion. Hence, different types of contrast dynamics were captured compactly in their suitable scales. In contrast, the low rank + sparse modeling could only provide a coarse separation of dynamics and static tissue, which resulted in neither truly sparse nor truly low rank components.

Multiscale Age Grouping for Collaborative Filtering

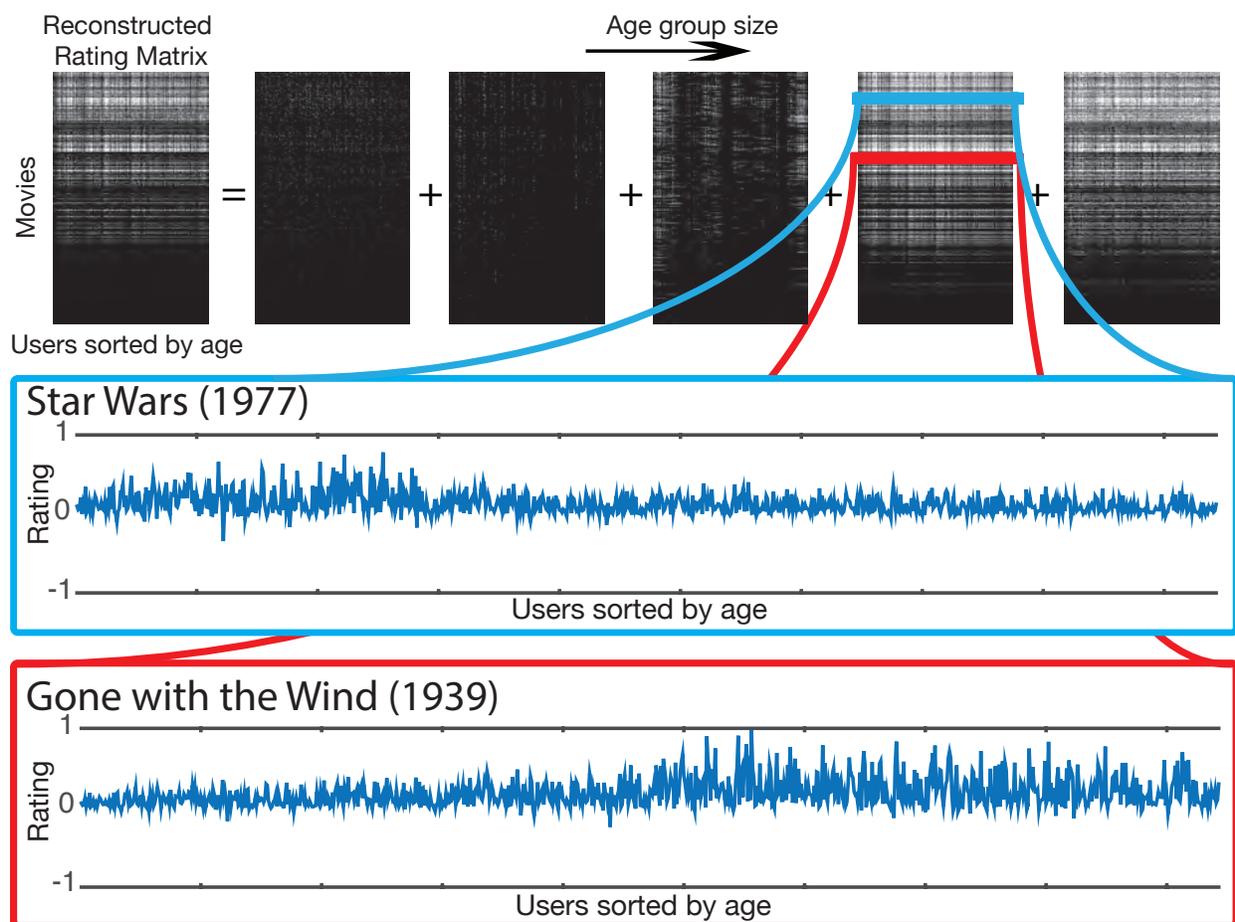


Figure 3.11: Multiscale low rank reconstructed matrix of the 100K MovieLens dataset. The extracted signal scale component captures the tendency that younger users rated Star Wars higher whereas the more senior users rated Gone with the Wind higher.

Collaborative filtering is the task of making predictions about the interests of a user using available information from all users. Since users often have a similar taste for the same

item, low rank modeling is commonly used to exploit the data similarity to complete the rating matrix [32, 25, 33]. On the other hand, low rank matrix completion does not exploit the fact that users with similar demographic backgrounds have a similar taste for similar items. In particular, users of similar age should have similar taste. Hence, I incorporated the proposed multiscale low rank modeling with matrix completion by partitioning users according to their age and compared it with the conventional low rank matrix completion. The proposed method belongs to the general class of collaborative filtering methods that utilize demographic information [42].

To incorporate multiscale low rank modeling into matrix completion, I change the data consistency constraint in problem (3.1) to $[\mathbf{Y}]_{jk} = [\sum_{i=1}^J \mathbf{X}_i]_{jk}$ for observed jk entries, and correspondingly, the update step for $\{\mathbf{X}_i\}_{i=1}^J$ in equation (3.8) is changed to $[\mathbf{X}_i]_{jk} \leftarrow [(\mathbf{Z}_i - \mathbf{U}_i) + \frac{1}{J}(\mathbf{Y} - \sum_{i=1}^J (\mathbf{Z}_i - \mathbf{U}_i))]_{jk}$ for observed jk entries and $[\mathbf{X}_i]_{jk} \leftarrow [\mathbf{Z}_i - \mathbf{U}_i]_{jk}$ for unobserved jk entries. I emphasize that my theoretical analysis does not cover matrix completion and the presented collaborative filtering application is mainly of empirical interest.

To compare the methods, I considered the 100K MovieLens dataset, in which 943 users rated 1682 movies. The resulting matrix was of size 1682×943 , where the first dimension represented movies and the second dimension represented users. The entire matrix had 93.7% missing entries. Test data was further generated by randomly undersampling the rating matrix by 5. The algorithms were then run on the test data and root mean squared errors were calculated over all available entries. To obtain a multiscale partition of the matrix, I sorted the users according to their age along the second dimension and partitioned them evenly into age groups.

Figure 3.11 shows a multiscale low rank reconstructed user rating matrix. Using multiple scales of block-wise low rank matrices, correlations in different age groups were captured. For example, one of the scales shown in Figure 3.11 captures the tendency that younger users rated Star Wars higher whereas the more senior users rated Gone with the Wind higher. The multiscale low rank reconstructed matrix achieved a root mean-squared-error of 0.9385 compared to a root mean-squared-error of 0.9552 for the low rank reconstructed matrix.

3.11 Conclusion

I have presented a multiscale low rank matrix decomposition method that combines both multiscale modeling and low rank matrix decomposition. Using a convex formulation, I can solve for the decomposition efficiently and exactly, provided that the multiscale signal components are incoherent. I provided a theoretical analysis of the convex relaxation for exact decomposition, which extends the analysis in Chandrasekaran et al. [28], and an analysis for approximate decomposition in the presence of additive noise, which extends the analysis in Agarwal et al. [29]. I also provided empirical results that the multiscale low rank decomposition performs well on real datasets.

I would also like to emphasize that the proposed decomposition empirically perform well even with the addition of noise, and hence in practice does not require manual tuning.

While some form of theoretical guarantees for the regularization parameters is provided in the approximate decomposition analysis, complete theoretical guarantees are not provided, especially for noiseless situations, and would be valuable for future work.

My experiments show that the multiscale low rank decomposition improves upon the low rank + sparse decomposition in a variety of applications. I believe that more improvement can be achieved if domain knowledge for each application is incorporated with the multiscale low rank decomposition. For example, for face shadow removal, prior knowledge of the illumination angle might be able to provide a better multiscale partition. For movie rating collaborative filtering, general demographic information and movie types can be used to construct multiscale partitions in addition to age information.

3.12 Proof of Theorem 3.5.1

In this section, I provide a proof of Theorem 3.5.1 and show that if $\{\mathbf{X}_i\}_{i=1}^J$ satisfies a deterministic incoherence condition, then the proposed convex formulation (3.1) recovers $\{\mathbf{X}_i\}_{i=1}^J$ from \mathbf{Y} exactly. The proof makes use of the dual certificate common in such proofs. I will begin by proving a technical lemma collecting three inequalities.

Lemma 3.12.1. *For $i = 1, \dots, J$, the following three inequalities hold,*

$$\|\mathcal{P}_{T_i}(\mathbf{X})\|_{(i)}^* \leq \|\mathbf{X}\|_{(i)}^* \text{ for any matrix } \mathbf{X} \quad (3.14)$$

$$\|\mathcal{P}_{T_i^\perp}(\mathbf{X})\|_{(i)}^* \leq \|\mathbf{X}\|_{(i)}^* \text{ for any matrix } \mathbf{X} \quad (3.15)$$

$$\|\mathbf{N}_j\|_{(i)}^* \leq \mu_{ij} \|\mathbf{N}_j\|_{(j)}^* \text{ for } j \neq i \text{ and } \mathbf{N}_j \in T_j \quad (3.16)$$

Proof. To show the first inequality (3.14), I recall that $\|\mathbf{X}\|_{(i)}^* = \max_{b \in P_i} \|\mathcal{B}_b(\mathbf{X})\|_{\text{msv}}$. Then, using the variational representation of the maximum singular value norm, I obtain,

$$\begin{aligned} \|\mathcal{P}_{T_i}(\mathbf{X})\|_{(i)}^* &= \max_{b \in P_i} \max_{\mathbf{u}, \mathbf{v}} \mathbf{u}^H \mathcal{B}_b(\mathcal{P}_{T_i}(\mathbf{X})) \mathbf{v} \\ &= \max_{b \in P_i} \max_{\substack{\mathbf{u} \in \text{col}(\mathcal{B}_b(\mathbf{X}_i)) \text{ or} \\ \mathbf{v} \in \text{row}(\mathcal{B}_b(\mathbf{X}_i))}} \mathbf{u}^H \mathcal{B}_b(\mathbf{X}) \mathbf{v} \\ &\leq \max_{b \in P_i} \max_{\mathbf{u}, \mathbf{v}} \mathbf{u}^H \mathcal{B}_b(\mathbf{X}) \mathbf{v} = \|\mathbf{X}\|_{(i)}^* \end{aligned}$$

where col and row denote the column and row spaces respectively.

Similarly, I obtain the second inequality (3.15):

$$\begin{aligned} \|\mathcal{P}_{T_i^\perp}(\mathbf{X})\|_{(i)}^* &= \max_{b \in P_i} \max_{\substack{\mathbf{u} \in \text{col}^\perp(\mathcal{B}_b(\mathbf{X}_i)) \text{ and} \\ \mathbf{v} \in \text{row}^\perp(\mathcal{B}_b(\mathbf{X}_i))}} \mathbf{u}^H \mathcal{B}_b(\mathbf{X}) \mathbf{v} \\ &\leq \max_{b \in P_i} \max_{\mathbf{u}, \mathbf{v}} \mathbf{u}^H \mathcal{B}_b(\mathbf{X}) \mathbf{v} = \|\mathbf{X}\|_{(i)}^* \end{aligned}$$

The third inequality (3.16) follows from the incoherence definition that $\mu_{ij} \geq \|\mathbf{N}_j\|_{(i)}^*/\|\mathbf{N}_j\|_{(j)}^*$ for any non-zero \mathbf{N}_j .

□

Next, I will show that if I can choose some parameters to “balance” the coherence between the scales, then the block-wise row/column spaces $\{T_i\}_{i=1}^J$ are independent, that is $\sum_{i=1}^J T_i$ is a direct sum. Consequently, each matrix N in the span of $\{T_i\}_{i=1}^J$ has a unique decomposition $\mathbf{N} = \sum_{i=1}^J \mathbf{N}_i$, where $\mathbf{N}_i \in T_i$.

Proposition 3.12.2. *If I can choose some positive parameters $\{\lambda_i\}_{i=1}^J$ such that*

$$\sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i} < 1, \quad \text{for } i = 1, \dots, J \quad (3.17)$$

then I have

$$T_i \cap \sum_{j \neq i} T_j = \{\mathbf{0}\}, \quad \text{for } i = 1, \dots, J \quad (3.18)$$

In particular when $J = 2$, the condition on $\{\mu_{12}, \mu_{21}\}$ reduces to $\mu_{12}\mu_{21} < 1$, which coincides with Proposition 1 in Chandrasekaran et al. [28]. I also note that given μ_{ij} , I can obtain $\{\lambda_i\}_{i=1}^J$ that satisfies the condition $\sum_{j \neq i} \mu_{ij} \lambda_j < \lambda_i$ by solving a linear program.

Proof. Suppose by contradiction that there exists $\{\lambda_i\}_{i=1}^J$ such that $\sum_{j \neq i} \mu_{ij} \lambda_j / \lambda_i < 1$, but $T_i \cap \sum_{j \neq i} T_j \neq \{\mathbf{0}\}$. Then there exists $\{\mathbf{N}_i \in T_i\}_{i=1}^J$ such that $\sum_{i=1}^J \lambda_i \mathbf{N}_i = \mathbf{0}$ and not all \mathbf{N}_i are zero. But this leads to a contradiction because for $i = 1, \dots, J$,

$$\begin{aligned} \|\mathbf{N}_i\|_{(i)}^* &= \left\| - \sum_{j \neq i} \frac{\lambda_j}{\lambda_i} \mathbf{N}_j \right\|_{(i)}^* \\ &\leq \sum_{j \neq i} \frac{\lambda_j}{\lambda_i} \mu_{ij} \|\mathbf{N}_j\|_{(j)}^* \\ &\leq \left(\sum_{j \neq i} \frac{\lambda_j}{\lambda_i} \mu_{ij} \right) \max_{j \neq i} \|\mathbf{N}_j\|_{(j)}^* \\ &< \max_{j \neq i} \|\mathbf{N}_j\|_{(j)}^*, \end{aligned}$$

where I have used equation (3.16) for the first inequality, Holder’s inequality for second inequality and $\sum_{j \neq i} \mu_{ij} \lambda_j / \lambda_i < 1$ for the last inequality. Hence, none of $\{\|\mathbf{N}_i\|_{(i)}^*\}_{i=1}^J$ is the largest of the set, which is a contradiction.

□

The next theorem shows an optimality condition of the convex program (3.1) in terms of its dual solution.

Theorem 3.12.3 (Lemma 4.2 [43]). $\{\mathbf{X}_i\}_{i=1}^J$ is the unique minimizer of the convex program (3.1) if there exists a matrix \mathbf{Q} such that for $i = 1, \dots, J$,

1. $\mathcal{P}_{T_i}(\mathbf{Q}) = \lambda_i \mathbf{E}_i$
2. $\|\mathcal{P}_{T_i^\perp}(\mathbf{Q})\|_{(i)}^* < \lambda_i$

Proof. Consider any non-zero perturbation $\{\Delta_i\}_{i=1}^J$ to $\{\mathbf{X}_i\}_{i=1}^J$ such that $\{\mathbf{X}_i + \Delta_i\}_{i=1}^J$ stays in the feasible set, that is $\sum_{i=1}^J \Delta_i = \mathbf{0}$. I will show that $\sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)} > \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)}$.

I first decompose Δ_i into orthogonal parts with respect to T_i , that is, $\Delta_i = \mathcal{P}_{T_i}(\Delta_i) + \mathcal{P}_{T_i^\perp}(\Delta_i)$. I also consider a specific subgradient $\mathbf{G} = [\mathbf{G}_1 \cdots \mathbf{G}_J]^\top$ of $\sum_{i=1}^J \lambda_i \|\cdot\|_{(i)}$ at $\{\mathbf{X}_i\}_{i=1}^J$ such that $\|\mathcal{P}_{T_i^\perp}(\mathbf{G}_i)\|_{(i)}^* \leq \lambda$, and $\langle \mathcal{P}_{T_i^\perp}(\Delta_i), \mathcal{P}_{T_i^\perp}(\mathbf{G}_i) \rangle = \lambda_i \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)}$. Then, from the definition of subgradient and the fact that $\sum_{i=1}^J \Delta_i = \mathbf{0}$, I have,

$$\begin{aligned} \sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)} &\geq \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} + \langle \Delta_i, \mathbf{G}_i \rangle \\ &= \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} + \langle \Delta_i, \mathbf{G}_i \rangle - \langle \Delta_i, \mathbf{Q} \rangle. \end{aligned}$$

Applying the orthogonal decomposition with respect to T_i and using $\mathcal{P}_{T_i}(\mathbf{G}_i) = \mathcal{P}_{T_i}(\mathbf{Q}) = \lambda_i \mathbf{E}_i$, I have

$$\begin{aligned} \sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)} &\geq \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} + \langle \mathcal{P}_{T_i^\perp}(\Delta_i), \mathcal{P}_{T_i^\perp}(\mathbf{G}_i) \rangle \\ &\quad - \langle \mathcal{P}_{T_i^\perp}(\Delta_i), \mathcal{P}_{T_i^\perp}(\mathbf{Q}) \rangle. \end{aligned}$$

Using Holder's inequality and the assumption for the subgradient \mathbf{G}_i , I obtain

$$\begin{aligned} \sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)} &\geq \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} + \lambda_i \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \\ &\quad - \|\mathcal{P}_{T_i^\perp}(\mathbf{Q})\|_{(i)}^* \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \\ &> \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)}. \end{aligned}$$

□

With Proposition 3.12.2 and Theorem 3.12.3, I am ready to prove Theorem 3.5.1.

Proof of Theorem 3.5.1. Since $\sum_{j \neq i} \mu_{ij} \lambda_j / \lambda_i < 1/2$, by Proposition 3.12.2, $T_i \cap \sum_{j \neq i} T_j = \{\mathbf{0}\}$ for all i . Thus, there is a unique matrix \mathbf{Q} in $\sum_{i=1}^J T_i$ such that $\mathcal{P}_{T_i}(\mathbf{Q}) = \lambda_i \mathbf{E}_i$. In addition, \mathbf{Q} can be uniquely expressed as a sum of elements in T_i . That is, $\mathbf{Q} = \sum_{i=1}^J \mathbf{Q}_i$ with $\mathbf{Q}_i \in T_i$. I now have a matrix \mathbf{Q} that satisfies the first optimality condition. In the following, I will show that it also satisfies the second optimality condition $\|\mathcal{P}_{T_i^\perp} \mathbf{Q}\|_{(i)}^* < \lambda_i$.

If the vector spaces $\{T_i\}_{i=1}^J$ are orthogonal, then \mathbf{Q}_i is exactly $\lambda_i \mathbf{E}_i$. Because they are not necessarily orthogonal, I express \mathbf{Q}_i as $\lambda_i \mathbf{E}_i$ plus a correction term $\lambda_i \boldsymbol{\epsilon}_i$. That is, I express $\mathbf{Q}_i = \lambda_i(\mathbf{E}_i + \boldsymbol{\epsilon}_i)$. Putting \mathbf{Q}_i 's back to \mathbf{Q} , I have

$$\mathbf{Q} = \sum_{i=1}^J \lambda_i (\mathbf{E}_i + \boldsymbol{\epsilon}_i).$$

Combining the above equation with the first optimality condition (3.12.3), $\mathcal{P}_{T_i}(\mathbf{Q}) = \lambda_i \mathbf{E}_i$, I have $\sum_{j=1}^J \lambda_j \mathcal{P}_{T_i}(\mathbf{E}_j + \boldsymbol{\epsilon}_j) = \lambda_i \mathbf{E}_i$. Since $\mathcal{P}_{T_i}(\mathbf{E}_i + \boldsymbol{\epsilon}_i) = \mathbf{E}_i + \boldsymbol{\epsilon}_i$, rearranging the equation, I obtain the following recursive expression for $\boldsymbol{\epsilon}_i$:

$$\boldsymbol{\epsilon}_i = -\mathcal{P}_{T_i} \left(\sum_{j \neq i} \frac{\lambda_j}{\lambda_i} (\mathbf{E}_j + \boldsymbol{\epsilon}_j) \right). \quad (3.19)$$

I now obtain a bound on $\|\mathcal{P}_{T_i^\perp}(\mathbf{Q})\|_{(i)}^*$ in terms of $\boldsymbol{\epsilon}_i$.

$$\begin{aligned} \|\mathcal{P}_{T_i^\perp}(\mathbf{Q})\|_{(i)}^* &= \|\mathcal{P}_{T_i^\perp} \left(\sum_{j \neq i} \lambda_j (\mathbf{E}_j + \boldsymbol{\epsilon}_j) \right)\|_{(i)}^* \\ &\leq \left\| \sum_{j \neq i} \lambda_j (\mathbf{E}_j + \boldsymbol{\epsilon}_j) \right\|_{(i)}^* \\ &\leq \sum_{j \neq i} \mu_{ij} \lambda_j (1 + \|\boldsymbol{\epsilon}_j\|_{(j)}^*) \\ &\leq \left(\sum_{j \neq i} \mu_{ij} \lambda_j \right) \max_{j \neq i} (1 + \|\boldsymbol{\epsilon}_j\|_{(j)}^*) \end{aligned} \quad (3.20)$$

where I obtain the first inequality from equation (3.14), second inequality from equation (3.15) and the last inequality from Holder's inequality.

Similarly, I obtain a recursive expression for $1 + \|\boldsymbol{\epsilon}_i\|_{(i)}^*$ using equation (3.19)

$$\begin{aligned}
1 + \|\boldsymbol{\epsilon}_i\|_{(i)}^* &= 1 + \|\mathcal{P}_{T_i}(\sum_{j \neq i} \frac{\lambda_j}{\lambda_i} (\mathbf{E}_j + \boldsymbol{\epsilon}_j))\|_{(i)}^* \\
&\leq 1 + \|\sum_{j \neq i} \frac{\lambda_j}{\lambda_i} (\mathbf{E}_j + \boldsymbol{\epsilon}_j)\|_{(i)}^* \\
&\leq 1 + \sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i} (1 + \|\boldsymbol{\epsilon}_j\|_{(j)}^*) \\
&\leq 1 + (\sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i}) \max_{j \neq i} (1 + \|\boldsymbol{\epsilon}_j\|_{(j)}^*)
\end{aligned}$$

where I obtain first inequality from equation (3.14), second inequality from equation (3.16) and the last inequality from Holder's inequality.

Taking the maximum over i on both sides and rearranging, I have

$$\max_i (1 + \|\boldsymbol{\epsilon}_i\|_{(i)}^*) \leq \frac{1}{1 - \max_i \sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i}}$$

Putting the bound back to equation (3.20), I obtain

$$\begin{aligned}
\|\mathcal{P}_{T_i^\perp}(\mathbf{Q})\|_{(i)}^* &\leq \lambda_i \frac{\sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i}}{1 - \max_i \sum_{j \neq i} \mu_{ij} \frac{\lambda_j}{\lambda_i}} \\
&< \lambda_i
\end{aligned} \tag{3.21}$$

where I used $\sum_{j \neq i} \mu_{ij} \lambda_j / \lambda_i < 1/2$ in the last inequality.

Thus, I have constructed a dual certificate \mathbf{Q} that satisfies the optimality conditions (3.12.3) and $\{\mathbf{X}_i\}_{i=1}^J$ is the unique optimizer of the convex problem (3.1). □

3.13 Proof of Theorem 3.6

In this section, I provide a proof of Theorem 3.6, showing that as long as I can choose the regularization parameters accordingly, I obtain a solution from the convex program (3.4) that is close to the ground truth $\{\mathbf{X}_i\}_{i=1}^J$.

I will begin by proving a technical lemma collecting three inequalities. Throughout the section, I will assume \mathbf{X}_Z is non-zero for simplicity, so that the subgradient of $\|\mathbf{X}_Z\|_{\text{fro}}$ is exactly $\mathbf{X}_Z / \|\mathbf{X}_Z\|_{\text{fro}}$.

Lemma 3.13.1. *For $i = 1, \dots, J$, the following three inequalities hold,*

$$\|\mathbf{X}_i\|_{(i)} - \|\mathbf{X}_i + \boldsymbol{\Delta}_i\|_{(i)} \leq \|\mathcal{P}_{T_i}(\boldsymbol{\Delta}_i)\|_{(i)} - \|\mathcal{P}_{T_i^\perp}(\boldsymbol{\Delta}_i)\|_{(i)} \tag{3.22}$$

$$\sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \leq 3 \sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} \quad (3.23)$$

$$\|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} \leq \sqrt{2 \sum_{b \in P_i} \text{rank}_b} \|\Delta_i\|_{\text{fro}} \quad (3.24)$$

Proof. I will prove the inequalities in order.

Let me choose a subgradient $\mathbf{G}_i = \mathbf{E}_i + \mathbf{W}_i$ of $\|\mathbf{X}_i\|_{(i)}$ at \mathbf{X}_i such that $\langle \mathbf{W}_i, \mathcal{P}_{T_i^\perp}(\Delta_i) \rangle = \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)}$. Then, from the definition of the subgradient, I have,

$$\begin{aligned} \|\mathbf{X}_i + \Delta_i\|_{(i)} &\geq \|\mathbf{X}_i\|_{(i)} + \langle \mathbf{G}_i, \Delta_i \rangle \\ &= \|\mathbf{X}_i\|_{(i)} + \langle \mathbf{E}_i, \mathcal{P}_{T_i}(\Delta_i) \rangle + \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \\ &\geq \|\mathbf{X}_i\|_{(i)} - \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} + \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \end{aligned}$$

where I used Holder's inequality in the end. Re-arranging, I obtain the first inequality.

For the second inequality, I note that since $\sum_{i=1}^J \mathbf{X}_i + \Delta_i + \mathbf{X}_Z + \Delta_Z = \mathbf{Y}$, I have $\Delta_Z = -\sum_{i=1}^J \Delta_i$. From the definition of subgradient, I obtain,

$$\begin{aligned} \lambda_Z \|\mathbf{X}_Z + \Delta_Z\|_{\text{fro}} &\geq \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} + \lambda_Z \left\langle \frac{\mathbf{X}_Z}{\|\mathbf{X}_Z\|_{\text{fro}}}, \Delta_Z \right\rangle \\ &= \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} - \sum_{i=1}^J \lambda_Z \left\langle \frac{\mathbf{X}_Z}{\|\mathbf{X}_Z\|_{\text{fro}}}, \Delta_i \right\rangle \\ &\geq \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} - \sum_{i=1}^J \lambda_Z \frac{\|\mathbf{X}_Z\|_{(i)}^*}{\|\mathbf{X}_Z\|_{\text{fro}}} \|\Delta_i\|_{(i)} \\ &\geq \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} - \sum_{i=1}^J \frac{\lambda_i}{2} \|\Delta_i\|_{(i)} \\ &\geq \lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} - \sum_{i=1}^J \frac{\lambda_i}{2} \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} - \frac{\lambda_i}{2} \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \end{aligned}$$

where I obtain the second inequality from Holder's inequality, the third inequality from the condition of λ_i and the last inequality from the triangle inequality.

Since $\{\mathbf{X}_i + \Delta_i\}_{i=1}^J$ and $\mathbf{X}_Z + \Delta_Z$ achieves the minimum objective function, I have

$$\lambda_Z \|\mathbf{X}_Z\|_{\text{fro}} + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} \geq \lambda_Z \|\mathbf{X}_Z + \Delta_Z\|_{\text{fro}} + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)}.$$

Substituting back, I obtain,

$$\begin{aligned} \lambda_{\mathcal{Z}} \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} &\geq \lambda_{\mathcal{Z}} \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} - \sum_{i=1}^J \frac{\lambda_i}{2} \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} - \frac{\lambda_i}{2} \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \\ &\quad + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} - \lambda_i \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} + \lambda_i \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \end{aligned}$$

Cancelling and re-arranging, I obtain the desired inequality (3.23) ,

$$\sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i^\perp}(\Delta_i)\|_{(i)} \leq 3 \sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)}$$

For the third inequality, recall that for any rank- r matrix \mathbf{X} , its nuclear norm $\|\mathbf{X}\|_{\text{nuc}}$ is upper bounded by $\sqrt{r} \|\mathbf{X}\|_{\text{fro}}$. Moreover, the projection of any matrix \mathbf{Y} to the column and row space T of a rank r matrix is at most rank- $2r$, that is $\text{rank}(\mathcal{P}_T(\mathbf{Y})) \leq 2r$. Hence, I obtain,

$$\begin{aligned} \|\mathcal{P}_{T_i}(\Delta_i)\|_{(i)} &= \sum_{b \in P_i} \|\mathcal{B}_b(\mathcal{P}_{T_i}(\Delta_i))\|_{\text{nuc}} \\ &\leq \sum_{b \in P_i} \sqrt{2 \text{rank}_b} \|\mathcal{B}_b(\Delta_i)\|_{\text{fro}} \\ &\leq \sqrt{\sum_{b \in P_i} 2 \text{rank}_b} \|\Delta_i\|_{\text{fro}} \end{aligned}$$

where the last inequality follows from Cauchy-Schwartz inequality and the fact that $\sum_{b \in P_i} \|\mathcal{B}_b(\Delta_i)\|_{\text{fro}}^2 = \|\Delta_i\|_{\text{fro}}^2$. \square

With these three inequalities, I now proceed to prove Theorem 3.6.

Proof of Theorem 3.6. From the optimality of $\{\mathbf{X}_i + \Delta_i\}_{i=1}^J$, I have the following inequality,

$$\begin{aligned} \lambda_{\mathcal{Z}} \|\mathbf{X}_{\mathcal{Z}} + \Delta_{\mathcal{Z}}\|_{\text{fro}} + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i + \Delta_i\|_{(i)} \\ \leq \lambda_{\mathcal{Z}} \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} + \sum_{i=1}^J \lambda_i \|\mathbf{X}_i\|_{(i)} \end{aligned}$$

Re-arranging, I obtain,

$$\begin{aligned} \|\mathbf{X}_{\mathcal{Z}} + \Delta_{\mathcal{Z}}\|_{\text{fro}} - \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} \\ \leq \frac{1}{\lambda_{\mathcal{Z}}} \sum_{i=1}^J \lambda_i (\|\mathbf{X}_i\|_{(i)} - \|\mathbf{X}_i + \Delta_i\|_{(i)}) \end{aligned}$$

For convenience, let me define $\mathbf{\Lambda}_T = \sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i}(\mathbf{\Delta}_i)\|_{(i)}$ and $\mathbf{\Lambda}_{T^\perp} = \sum_{i=1}^J \lambda_i \|\mathcal{P}_{T_i^\perp}(\mathbf{\Delta}_i)\|_{(i)}$. Then, from Lemma 3.13.1 equation (3.22), I obtain,

$$\|\mathbf{X}_{\mathcal{Z}} + \mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}} - \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} \leq \frac{1}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp}) \quad (3.25)$$

I would like to keep only $\mathbf{\Delta}_{\mathcal{Z}}$ on the left hand side and cancel $\mathbf{X}_{\mathcal{Z}}$. To do this, I multiply both sides of equation (3.25) with $\|\mathbf{X}_{\mathcal{Z}} + \mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}} + \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}$. Then, using $(a+b)(a-b) = a^2 - b^2$, I expand the left hand side as:

$$\begin{aligned} \text{L.H.S.} &= \|\mathbf{X}_{\mathcal{Z}} + \mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 - \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}^2 \\ &= \|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 + 2\langle \mathbf{X}_{\mathcal{Z}}, \mathbf{\Delta}_{\mathcal{Z}} \rangle \end{aligned}$$

Recall that $\mathbf{\Delta}_{\mathcal{Z}} = -\sum_{i=1}^J \mathbf{\Delta}_i$, I obtain the following lower bound for the left hand side:

$$\begin{aligned} \text{L.H.S.} &= \|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 - 2\langle \mathbf{X}_{\mathcal{Z}}, \sum_{i=1}^J \mathbf{\Delta}_i \rangle \\ &\geq \|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 - 2 \sum_{i=1}^J \|\mathbf{X}_{\mathcal{Z}}\|_{(i)}^* \|\mathbf{\Delta}_i\|_{(i)} \\ &\geq \|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 - \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \sum_{i=1}^J \lambda_i \|\mathbf{\Delta}_i\|_{(i)} \\ &\geq \|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 - \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T + \mathbf{\Lambda}_{T^\perp}) \end{aligned}$$

where I used Holder's inequality for the first inequality, the condition for λ_i for the second inequality, and the triangle inequality for the last inequality.

I now turn to upper bound the right hand side. I know $\|\mathbf{X}_{\mathcal{Z}} + \mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}} \leq \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} + (\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp})/\lambda_{\mathcal{Z}}$ from equation (3.25). Hence, I obtain,

$$\begin{aligned} \text{R.H.S.} &= (\|\mathbf{X}_{\mathcal{Z}} + \mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}} + \|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}) \frac{1}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp}) \\ &\leq (2\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}} + \frac{1}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp})) \frac{1}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp}) \end{aligned}$$

Using Lemma 3.13.1 equation (3.23), I have,

$$\begin{aligned} \text{R.H.S.} &\leq 2 \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp}) + \frac{1}{\lambda_{\mathcal{Z}}^2}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp})^2 \\ &\leq 2 \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}}(\mathbf{\Lambda}_T - \mathbf{\Lambda}_{T^\perp}) + 16 \frac{1}{\lambda_{\mathcal{Z}}^2} \mathbf{\Lambda}_T^2 \end{aligned} \quad (3.26)$$

Combining and simplifying the lower bound and the upper bound, I obtain,

$$\|\mathbf{\Delta}_{\mathcal{Z}}\|_{\text{fro}}^2 \leq 3 \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \mathbf{\Lambda}_T + 16 \frac{1}{\lambda_{\mathcal{Z}}^2} \mathbf{\Lambda}_T^2 \quad (3.27)$$

I now lower bound $\|\Delta_{\mathcal{Z}}\|_{\text{fro}}^2 = \|\sum_{i=1}^J \Delta_i\|_{\text{fro}}^2$ by individual terms:

$$\begin{aligned} \left\| \sum_{i=1}^J \Delta_i \right\|_{\text{fro}}^2 &= \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 + \left\langle \Delta_i, \sum_{j \neq i} \Delta_j \right\rangle \\ &\geq \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 - \|\Delta_i\|_{(i)} \sum_{j \neq i} \|\Delta_j\|_{(i)}^* \end{aligned}$$

where I used Holder's inequality for the last inequality.

Now, using the assumption that both $\|\mathbf{X}_j\|_{(i)}^*$ and $\|\mathbf{X}_j + \Delta_j\|_{(i)}^*$ are bounded by α_{ij} . I have,

$$\begin{aligned} \left\| \sum_{i=1}^J \Delta_i \right\|_{\text{fro}}^2 &\geq \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 - \|\Delta_i\|_{(i)} \sum_{j \neq i} 2\alpha_{ij} \\ &\geq \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 - \lambda_i \|\Delta_i\|_{(i)} \\ &\geq \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 - \Lambda_T - \Lambda_{T^\perp} \\ &\geq \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 - 4\Lambda_T \end{aligned}$$

where I used the triangle inequality for the second inequality and Lemma 3.13.1 for the last inequality.

Substituting the lower bound back to equation (3.27), I have

$$\sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 \leq \left(3 \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} + 4\right) \Lambda_T + 16 \frac{1}{\lambda_{\mathcal{Z}}^2} \Lambda_T^2 \quad (3.28)$$

I now turn to upper bound the equation. From Lemma 3.13.1, I know that $\Lambda_T \leq \sum_{i=1}^J \lambda_i \sqrt{2 \sum_{b \in P_i} \text{rank}_b} \|\Delta_i\|_{\text{fro}}$. Hence, I have,

$$\begin{aligned} 16\Lambda_T^2 &\leq 16 \left(\sum_{i=1}^J \lambda_i \sqrt{2 \sum_{b \in P_i} \text{rank}_b} \|\Delta_i\|_{\text{fro}} \right)^2 \\ &\leq 16 \left(\sum_{i=1}^J \lambda_i^2 2 \sum_{b \in P_i} \text{rank}_b \right) \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 \\ &\leq \frac{1}{2} \lambda_{\mathcal{Z}}^2 \sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 \end{aligned}$$

where I used Cauchy-Schwartz's inequality for the second inequality and the condition for $\lambda_{\mathcal{Z}}$ for the third inequality. Hence, substituting back to equation (3.28), rearranging and ignoring constants, I have,

$$\sum_{i=1}^J \|\Delta_i\|_{\text{fro}}^2 \lesssim \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \sum_{i=1}^J \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b} \|\Delta_i\|_{\text{fro}}$$

Completing the squares with respect to $\|\Delta_i\|_{\text{fro}}$ gives us,

$$\begin{aligned} & \sum_{i=1}^J \left(\|\Delta_i\|_{\text{fro}} - \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b} \right)^2 \\ & \lesssim \sum_{i=1}^J \left(\frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b} \right)^2 \end{aligned}$$

Using the triangle inequality to lower bound the left-hand side, I obtain

$$\begin{aligned} & \sum_{i=1}^J \|\Delta_i\|_{\text{fro}} - \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b} \\ & \lesssim \sqrt{\sum_{i=1}^J \left(\frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b} \right)^2} \end{aligned}$$

Using the fact that ℓ_1 -norm is larger than the ℓ_2 -norm, and re-arranging give the desired result,

$$\sum_{i=1}^J \|\Delta_i\|_{\text{fro}} \lesssim \frac{\|\mathbf{X}_{\mathcal{Z}}\|_{\text{fro}}}{\lambda_{\mathcal{Z}}} \sum_{i=1}^J \lambda_i \sqrt{\sum_{b \in P_i} \text{rank}_b}$$

□

Chapter 4

High Resolution Volumetric Dynamic MRI from Non-Gated Acquisitions

4.1 Introduction

Three-dimensional (3D) dynamic MRI has become a core technology in a wide variety of applications, including dynamic contrast enhanced (DCE) imaging [56, 54], pulmonary imaging [57, 5], and flow imaging [58, 59]. A high spatiotemporal resolution realization can provide comprehensive information in a single scan, greatly simplifying clinical workflow and reducing overall scan time. By providing volumetric information at a high spatiotemporal resolution, 3D dynamic MRI can also provide additional information about the underlying anatomy and physiology, compared to 2D scans.

On the other hand, reconstructing high resolution 3D dynamic MRI is inherently underdetermined and demanding of computation and memory. To mitigate the reconstruction difficulty, most existing methods rely on view sharing and/or gating. These techniques have demonstrated their effectiveness in capturing smooth or periodic dynamics, but inherently fail to represent the full dynamics, including transient contrast injection, irregular breathing, and bulk motion. As described in Chapter 2, view sharing techniques [2, 3, 4] aggregate k-space data in a sliding window, but have the limitation of smoothing the underlying dynamics. Another approach is to impose periodicity assumptions of the underlying dynamics using gating or data binning [6, 60, 5, 7]. Data sorting can be accomplished by leveraging external navigator signals or navigators derived from the MR data itself. The drawback is that when the underlying periodic assumption does not hold, the reconstruction breaks down and can exhibit severe artifacts.

In this chapter, I focus on reconstructing high resolution 3D dynamic MRI from continuous non-gated acquisitions. Such reconstruction can go beyond the restrictive periodicity assumption of gating/binning and potentially achieve higher temporal resolution than view sharing. The main challenge here is that the reconstruction problem is vastly underdetermined, and computationally and memory demanding. In particular, the problem I consider

requires recovering hundreds of gigabytes of an image from a few gigabytes of measurements.

I propose three innovations to overcome these challenges: 1) a compressed representation using the multi-scale low rank matrix model (MSLR) described in Chapter 3 to regularize the problem 2) an objective function formulation that directly solves for the compressed representation to reduce memory usage, and 3) an algorithm using stochastic optimization to reduce computation and memory burden.

MSLR was previously studied in Chapter 3 and generalizes low rank (LR), locally low rank (LLR) [61], and low rank + sparse (L+S) [55] matrix models. The proposed representation can capture all scales of dynamics, not just local, global, or sparse. Hence, it can obtain a more accurate and compact signal representation than other LR methods. In Section 4.2, I will combine the compact MSLR representation with the MRI acquisition model.

A consequence of using this compressed representation is that the underlying dynamic sequence of images, which requires hundreds of gigabytes of storage, can be represented in mere few gigabytes. I propose an objective function that directly solves for the compressed representation. Such an approach makes it feasible to implement the reconstruction on local workstations. In the context of LR modeling, this is often referred to as the Burer-Monteiro heuristic, which will be explained in Section 4.3.

Finally, to further reduce computation, I develop a reconstruction algorithm incorporating stochastic optimization. This reduces the number of Fourier transforms from hundreds per iteration to a single one per iteration. The idea of using stochastic optimization to accelerate reconstruction is not new in medical imaging. Examples include the ordered subset algorithms for positron emission tomography reconstruction [62, 63], and the algebraic reconstruction technique for computed tomography [64]. A recent work [65] also proposed to use stochastic optimization to accelerate MRI reconstruction. Incorporating stochastic optimization for the proposed method reduces the reconstruction time from days to hours, and will be described in more detail in Section 4.4.

I will show that the proposed method, reconstructed at near millimeter spatial resolution and subsecond temporal resolution, can visualize certain transient contrast dynamics that are lost in low frame-rate reconstructions. In particular, to compare the proposed method quantitatively with existing techniques, I evaluate it on simulated dynamic datasets. Then using in vivo scans, I demonstrate its feasibility in DCE imaging acquired with the 3D cones trajectory [66] and lung imaging acquired with 3D ultra short time echo (UTE) radial trajectory [57].

4.2 Forward Model with MSLR

In this section, I will formalize the image acquisition model, and incorporate the MSLR representation directly into the reconstruction such that it can compactly represent the underlying volumetric image sequence.

I begin by considering the multi-channel MRI acquisition forward model and divide the overall scan time into T frames. An approximation that I make here is that for fine enough

temporal resolution, the underlying image for each frame is static. In particular, given the dynamic images $\{\mathbf{x}_t \in \mathbb{C}^N\}_{t=1}^T$, the following k-space measurements $\{\mathbf{y}_t\}_{t=1}^T$ are acquired,

$$\mathbf{y}_t = \mathbf{F}_t \mathbf{S} \mathbf{x}_t + \mathbf{w}_t \text{ for } t = 1, \dots, T \quad (4.1)$$

where $\mathbf{F}_t \in \mathbb{C}^{MC \times NC}$ is a Fourier sampling operator, $\mathbf{S} \in \mathbb{C}^{NC \times N}$ is a sensitivity map operator, and $\mathbf{w}_t \in \mathbb{C}^{MC}$ represents the white Gaussian noise of the acquisition.

Since the number of measurements is much smaller than the total dynamic image size, it is necessary to impose additional constraints in the reconstruction. As described in Chapter 2, compressed sensing MRI [10] has shown that with an incoherent sampling acquisition, the underlying signal can be recovered with an efficient algorithm if it can be represented compactly using a sparsifying transform. In this chapter, I will use MSLR for compactly representing 3D dynamic images.

Multiscale Low Rank Matrix Model

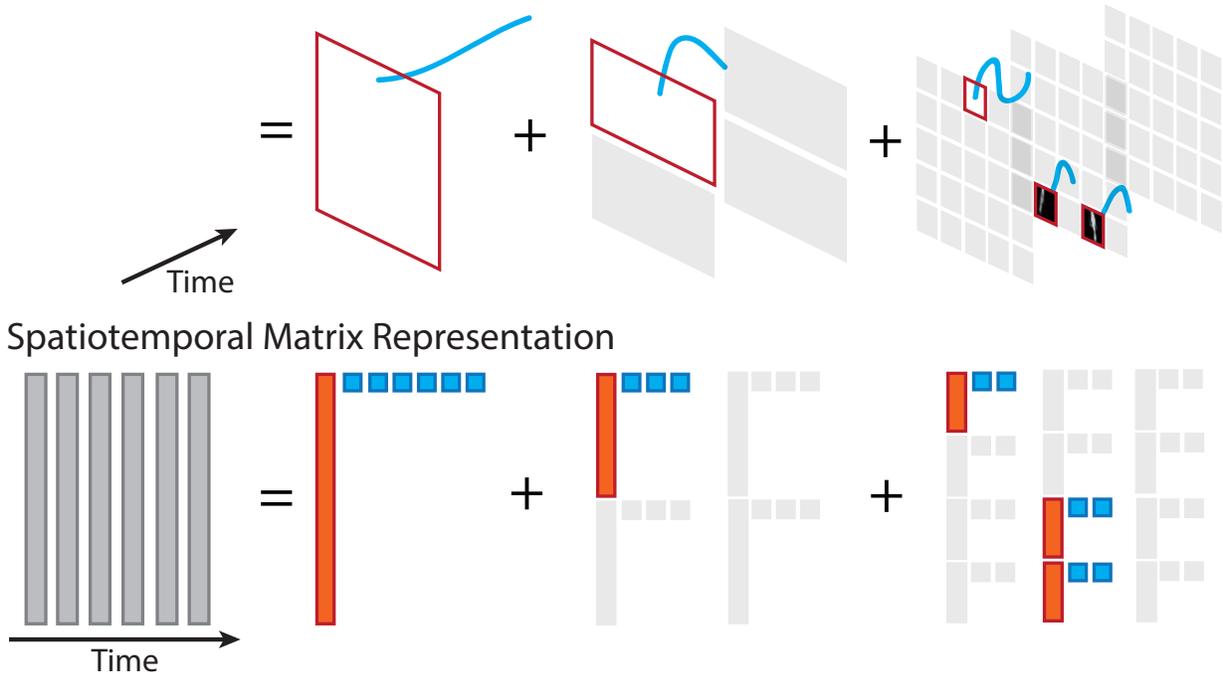


Figure 4.1: Illustration of the multi-scale low rank matrix model.

LR modeling [21, 67] is effective at representing static tissues, global contrast changes or smooth dynamics, and has shown to be effective in many dynamic imaging applications [68, 69, 70, 71]. A rarely used advantage is that explicit LR factorization can drastically reduce memory usage by representing dynamic images with a few basis vectors. This memory saving property was also explored in the work of Christodoulou et al. [7], and is crucial in my application as detailed in Section 4.3.

On the other hand, LR representation does not capture the spatiotemporal locality of the underlying components. Even if most dynamics are concentrated locally in space, such as contrast dynamics in the blood vessels, LR still requires many basis vectors, each of which has the size of the full image. To mitigate this, LLR was proposed by Trazasko et al. [61] to better capture spatial locality, and has been shown to represent dynamics more accurately in DCE imaging [54]. Another direction to enhance LR is the L+S representation [55]. L+S separately represents static background dynamics as a LR matrix and fast transient dynamics as a sparse matrix.

Here I adopt the MSLR representation that generalizes the above mentioned LR models, to capture dynamics at multiple scales. In MSLR, the spatiotemporal matrix is represented as a sum of block-wise low rank matrices with increasing scales of block sizes, as illustrated in Figure 4.1.

Concretely, for scale $i = 1, \dots, J$, I consider a multiscale partition $\{P_i\}_{i=1}^J$, and let $\{\{\mathbf{L}_b \in \mathbb{C}^{N_i \times \text{rank}_b}\}_{b \in P_i}\}_{i=1}^J$ be the block-wise spatial bases, and $\{\{\mathbf{R}_b \in \mathbb{C}^{T_i \times \text{rank}_b}\}_{b \in P_i}\}_{i=1}^J$ be the block-wise temporal bases. The volumetric image sequence is then represented as

$$\mathbf{x}_t = \sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \mathbf{L}_b [\mathbf{R}_b]_t^H, \quad (4.2)$$

where the operator $\mathbf{B}_b^H \in \mathbb{C}^{N \times N_i}$ embeds the b th block matrix in the i th scale into the full image, and $[\mathbf{R}_b]_t$ extracts the block temporal bases corresponding to time t . Here, I note that \mathbf{L}_b and \mathbf{R}_b are not orthonormal matrices because they incorporate the singular values in them.

The overall forward model is then given by,

$$\mathbf{y}_t = \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \mathbf{L}_b [\mathbf{R}_b]_t^H \right) + \mathbf{w}_t. \quad (4.3)$$

4.3 Objective function using the Burer-Monteiro Heuristic

Given the forward model, one way to impose low rank constraints is through convex relaxation minimizing the nuclear norm [24, 33, 25], or equivalently the sum of singular values. In particular, let $\|\cdot\|_*$ denote the nuclear norm, one possible formulation is to consider the following nuclear norm regularized problem,

$$\underset{\{\mathbf{X}_b\}_b}{\text{minimize}} \sum_{t=1}^T \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H [\mathbf{X}_b]_t \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \lambda_i \|\mathbf{X}_b\|_* \quad (4.4)$$

where λ_i is the regularization parameter for scale i . Nuclear norm formulations enjoy rich theoretical guarantees, and in particular can recover the underlying low rank matrix when the

sensing model is incoherent [24, 33, 25]. In addition, MSLR using nuclear norm minimization was previously studied in Chapter 3, and has been shown that the decomposition can be recovered when $\{\lambda_i\}_i$ are chosen proportional to:

$$\lambda_i \propto \sqrt{N_i} + \sqrt{T_i} + \sqrt{\min(N_i, T_i) \log |P_i|}. \quad (4.5)$$

On the other hand, a significant downside of the convex formulation is that it uses tremendously more memory since the explicit matrices are stored in all scales even if they are extremely low rank. In particular, for image size $320 \times 320 \times 320$ and 500 frames, merely storing the image using complex single precision floats requires 125 GBs. Applying iterative algorithms would require a few times more memory as workspace, which can approach terabytes. Local workstations simply cannot handle such memory demand.

Rather than minimizing the convex problem (4.4), I consider the Burer-Monteiro heuristic [72, 33] to directly solve for the compressed representation using the following objective function:

$$\underset{\{\mathbf{L}_b, \mathbf{R}_b\}_b}{\text{minimize}} \sum_{t=1}^T \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \mathbf{L}_b [\mathbf{R}_b]_t^H \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \lambda_i (\|\mathbf{L}_b\|_F^2 + \|\mathbf{R}_b\|_F^2) \quad (4.6)$$

The primary benefit of this formulation is that it uses much less memory, and, for a typical 3D volum, can even fit the variables in GPUs. Existing works [72, 33] have shown that the global minimum of this non-convex problem is equivalent to the global minimum of the convex problem using nuclear norm, as long as the matrix rank prescribed is larger than the underlying rank. More surprisingly, recent theoretical results [73, 74] have shown that under the usual compressed sensing incoherence condition (such as restricted isometry property), alternating minimization can reach the global minimum even when solving the non-convex formulation of the low rank factorization.

4.4 Stochastic Alternating Minimization

Since the objective function (4.6) is reduced to a linear least squares problem when either \mathbf{L} or \mathbf{R} is fixed, a natural way of optimizing it is through alternating minimization:

$$\begin{aligned} \{\hat{\mathbf{R}}_b\}_b &= \underset{\{\mathbf{R}_b\}_b}{\text{argmin}} \sum_{t=1}^T \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \hat{\mathbf{L}}_b [\mathbf{R}_b]_t^H \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \lambda_i \|\mathbf{R}_b\|_F^2 \\ \{\hat{\mathbf{L}}_b\}_b &= \underset{\{\mathbf{L}_b\}_b}{\text{argmin}} \sum_{t=1}^T \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \mathbf{L}_b [\hat{\mathbf{R}}_b]_t^H \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \lambda_i \|\mathbf{L}_b\|_F^2 \end{aligned} \quad (4.7)$$

However, the above alternating minimization iteration still requires performing many non-uniform fast Fourier transforms per iteration, one for each time frame. To address this,

I propose to apply stochastic optimization techniques to accelerate the reconstruction, which updates the bases using a random subset of the measurements.

Concretely, in each iteration, I randomly pick a time index t , and perform alternating minimization stochastically as follows:

$$\{\hat{\mathbf{R}}_b\}_b = \operatorname{argmin}_{\{\mathbf{R}_b\}_b} \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \hat{\mathbf{L}}_b [\mathbf{R}_b]_t^H \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \lambda_i \|\mathbf{R}_b\|_F^2 \quad (4.8)$$

$$\{\hat{\mathbf{L}}_b\}_b = \operatorname{argmin}_{\{\mathbf{L}_b\}_b} \left\| \mathbf{F}_t \mathbf{S} \left(\sum_{i=1}^J \sum_{b \in P_i} \mathbf{B}_b^H \mathbf{L}_b [\hat{\mathbf{R}}_b]_t^H \right) - \mathbf{y}_t \right\|_2^2 + \sum_{i=1}^J \sum_{b \in P_i} \left(\frac{\lambda_i}{T} \|\mathbf{L}_b\|_F^2 + \frac{1}{\alpha} \|\mathbf{L}_b - \hat{\mathbf{L}}_b\|_F^2 \right) \quad (4.9)$$

where α is a step-size parameter to control how much to update $\hat{\mathbf{L}}_b$. The $\hat{\mathbf{L}}_b$ update step is generally referred to as an implicit update or stochastic proximal update [75]. It is known to converge to an approximate stationary point in expectation, and more robust to the step-size choice α than stochastic gradient methods [76]. Each of the sub-problems in (4.8) and (4.9) can be approximated using the conjugate gradient algorithm. Figure 4.2 shows an illustration of the proposed algorithm.

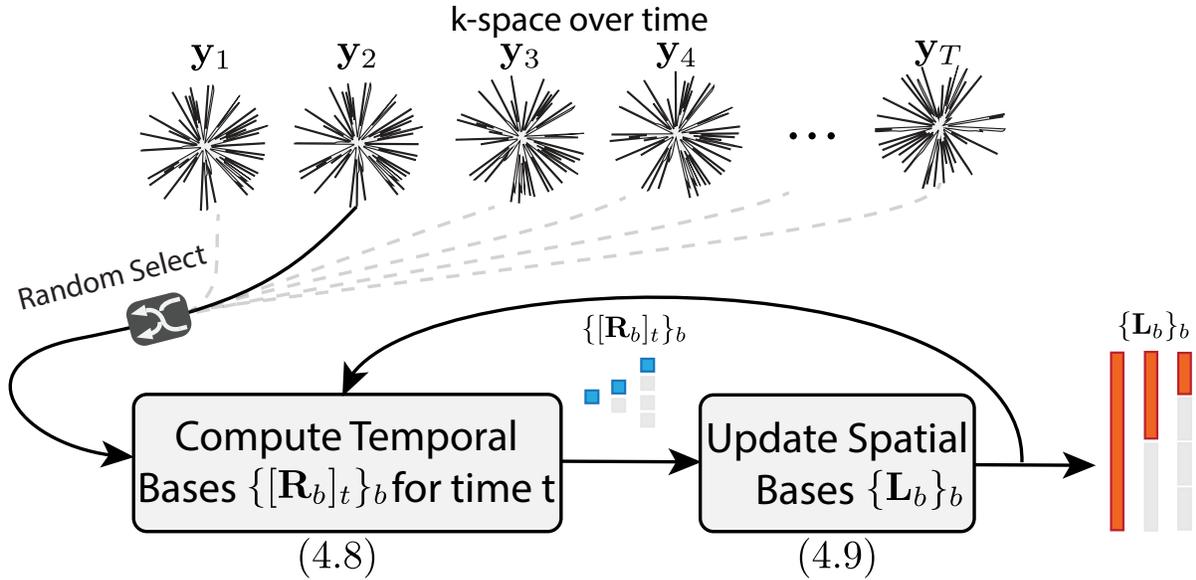


Figure 4.2: Illustration of the algorithm. Alternating minimization is performed stochastically using a random subset of the measurements.

4.5 Methods

The proposed reconstruction was implemented in Python using the package SigPy [77] on a workstation with two Intel Xeon Gold CPUs and four Titan Xp GPUs. All operations, except loading the data and splitting into frames, were performed using a GPU. Here, I note that the proof-of-concept implementation only used one GPU for each reconstruction instance, and can be sped up using multiple GPUs.

The number of scales in MSLR was determined by the GPU memory constraint. In particular, I used three scales for the experiments. The first scale used a block size in which the spatial bases $\{\hat{\mathbf{L}}_b\}_b$ cover the entire spatial dimension and the temporal bases $\{\hat{\mathbf{R}}_b\}_b$ cover all frames. The second halved the block size in each dimension, including time, by two, and the last further halved it each dimension by two. To prevent blocking artifacts, the second and the third scales had overlapping blocks, with each block overlapping by half the block size in all spatial dimensions. In addition, the overlapped parts are weighted by the Hann window to provide a smooth transition between blocks. Each voxel in the end was represented by eight basis vectors in each scale, including the overlapped bases.

The spatial bases $\{\hat{\mathbf{L}}_b\}_b$ were initialized as white Gaussian noise, and the temporal bases $\{\hat{\mathbf{R}}_b\}_b$ were initialized as zeros. Each sub-problem in the alternating minimization was approximated using the conjugated gradient algorithm with 10 iterations, and initialized with the previous iterate solution.

I evaluated the proposed method on simulated dynamic datasets to compare the proposed method quantitatively with existing reconstruction methods. I also demonstrated the feasibility of the proposed method in DCE datasets acquired with the 3D cones trajectory [66] and lung datasets acquired with the 3D UTE radial trajectory [57].

Simulation

I applied the proposed method on a simulated data to compare it with three other low rank reconstructions: LR, LLR, and L+S. In particular, MR-XCAT [78], a realistic numerical 2D dynamic phantom simulating DCE and respiratory motion, was used for the simulation. A perfusion phantom was generated with 50 frames, 224×192 matrix size, 8 coils, and $TR=2ms$, and sampled in k-space using a golden-angle radial trajectory. Each frame contains 30 spokes. Each spoke has 224 readout points. Complex white Gaussian noise was added in k-space with standard deviation equal to 0.001 of the maximum k-space magnitude. Peak-signal-to-noise-ratio (PSNR) was used as the evaluation metric. The regularization parameters for all methods were chosen between 0.001 to 0.1 with step 0.01 to maximize the PSNR. The locally low rank implementation used a spatial block size of 32×32 and a temporal block size of 50, which covered all frames.

Dynamic Contrast Enhanced Imaging Acquired with the 3D Cones Trajectory

I applied the proposed reconstruction on two DCE datasets from pediatric patients to qualitatively evaluate its performance. The first dataset has regular respiratory motion and little bulk motion. The second dataset has repeated large bulk motion throughout the scan. I selected this dataset in order to stress test the reconstruction limits. The patient scans used a field-of-view (FOV) that is smaller than the patient body to reduce the gridding reconstruction complexity. For iterative reconstruction, the actual FOV was required to prevent artifacts from model mismatch. The reconstruction matrix size was estimated by setting a threshold of 0.01 of the maximum amplitude of a single-frame low resolution gridding reconstruction with a resolution of about 4 mm^3 . To evaluate how different a high frame-rate reconstruction is from a standard low frame-rate one, I also reconstructed the datasets with one tenth the frame-rate.

In particular, the first DCE dataset was acquired on a GE 3T scanner using a spoiled gradient echo (SPGR) sequence with a 3D golden-angle cones trajectory, 16-channel coil array, an overall scan time of 4 minutes 40 seconds, $TE=0.1 \text{ ms}$, $TR=5.8 \text{ ms}$, flip angle=14 degrees, and bandwidth=125 kHz. The number of readout points is 624, and the number of interleaves is 48129. The spatial resolution was reconstructed at $1 \times 1 \times 2.8 \text{ mm}^3$, and the matrix size was reconstructed at $392 \times 318 \times 165$. The high frame-rate result was reconstructed with 580 ms temporal resolution, and the low frame-rate result was reconstructed with 5.8 s temporal resolution.

The second DCE dataset was acquired on a GE 3T scanner, using a SPGR sequence with a 3D golden-angle cones trajectory, 12-channel coil array, an overall scan time of 5 minutes 11 seconds, $TE=0.1 \text{ ms}$, $TR=9.6 \text{ ms}$, flip angle 15 degrees, bandwidth=125 kHz. The number of readout points was 711, and the number of interleaves was 41861. The spatial resolution was reconstructed at $1 \times 1 \times 1.8 \text{ mm}^3$, and the matrix size was reconstructed at $370 \times 163 \times 126$. The high frame-rate result was reconstructed with 622ms temporal resolution, and the low frame-rate result was reconstructed with 6.2 s temporal resolution.

Pulmonary Imaging Acquired with the 3D UTE Radial Trajectory

I applied the proposed reconstruction on two lung datasets from adult patients to evaluate its performance qualitatively. Similar to the DCE experiments, the first dataset has variable rate respiratory motion and little bulk motion. The second dataset has sudden motion such as coughing throughout the scan and is meant to stress test the reconstruction limits. Similar to the DCE datasets, the patient scans have a small prescribed FOV. The reconstruction matrix size was estimated the same as the DCE experiments.

In particular, the first lung dataset was acquired on a GE 3T Discovery MR750 clinical scanner (GE Healthcare, Waukesha, WI) using an optimized UTE sequence [57] with a 3D bit-reversed radial trajectory, 8-channel coil array, an overall scan time of 4 minutes 18 seconds, $TE=80 \mu\text{s}$, $TR=3.4 \text{ ms}$, flip angle 4 degrees, and sampling bandwidth=250 kHz.

The number of readout points was 654, and the number of interleaves was 75768. The spatial resolution was reconstructed at $1.25 \times 1.25 \times 1.25 \text{ mm}^3$, and the matrix size was reconstructed at $353 \times 197 \times 332$. The temporal resolution was reconstructed at 515 ms.

The second lung dataset was acquired on a GE 3T Discovery MR750 clinical scanner (GE Healthcare, Waukesha, WI) using an optimized UTE sequence [57] with a 3D bit-reversed radial trajectory, 8-channel coil array, an overall scan time of 4 minutes 18 seconds, TE=80 μs , TR=3.4 ms, flip angle 4 degrees, and sampling bandwidth=250 kHz. The number of readout points was 654, and the number of interleaves was 75768. The spatial resolution was reconstructed at $1.25 \times 1.25 \times 1.25 \text{ mm}^3$, and the matrix size was reconstructed at $327 \times 183 \times 396$. The temporal resolution was reconstructed at 515 ms.

4.6 Results

In the spirit of reproducible research, I provide a software package in Python to reproduce the results described in this chapter. The software package can be downloaded from:

https://github.com/mikgroup/low_rank_recon.git

Simulation

Figures 4.3 and 4.4 show the reconstructions of the MRXCAT phantom using LR, LLR, L+S and the proposed MSLR representations (full-length video at: <https://doi.org/10.6084/m9.figshare.7466579>). The proposed MSLR showed the least amount of streaking artifacts, compared to other methods. These artifacts are more apparent during contrast enhancement as pointed out by the red arrows in Figure 4.3.

All methods display certain temporal artifacts, but overall LLR and MSLR show more accurate dynamics. In particular, in Figure 4.4, the red arrows point to ghosting artifacts from other frames in all methods except the proposed MSLR. The yellow arrows point to contrast mismatch compared to the ground truth in all methods except LLR. These artifacts are common in LR methods when the underlying dynamics cannot be represented with the few number of bases. Both LR and L+S also display blurring artifacts in the temporal profiles. PSNRs for LR, LLR, L+S, and MSLR were respectively 35.99 dB, 34.80 dB, 36.04 dB, and 36.76 dB, which match with the visual assessment.

Dynamic Contrast Enhanced Imaging Acquired with the 3D Cones Trajectory

Figure 4.5 shows the high frame-rate and low frame-rate reconstructions of the first DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453178>). In the high frame-rate reconstruction, contrast injection dynamics faster than respiration are nicely shown, such as those in the heart and in the blood vessels. From the video, slight bulk motion,

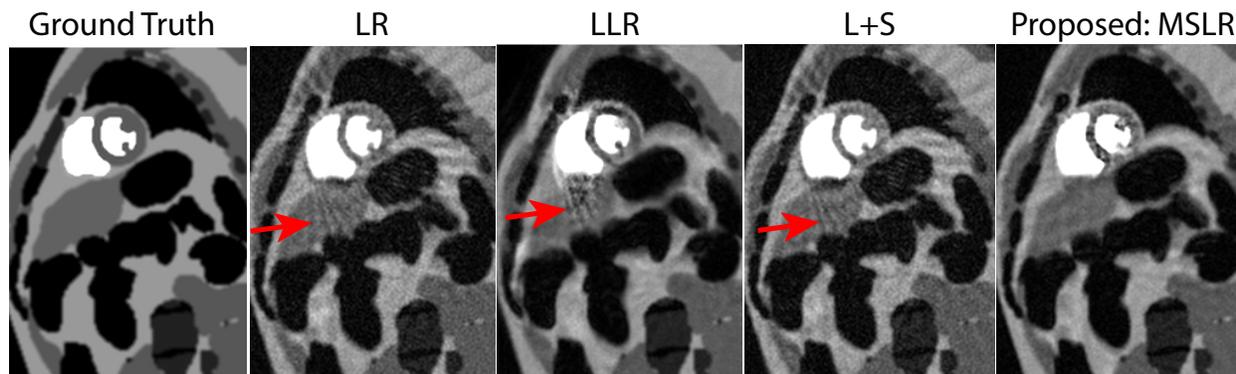


Figure 4.3: Reconstructions of the MRXCAT phantom using LR, LLR, L+S and the proposed MSLR representations (full-length video at: <https://doi.org/10.6084/m9.figshare.7466579>). Frames during contrast injection are shown. The proposed method with MSLR shows the least amount of streaking artifacts, compared to other methods as pointed out by the red arrows. Overall, it also achieves the highest PSNR.

pointed out by the white arrows, was observed when the contrast was injected. Respiratory motion was also observed throughout the scan. For the low frame-rate reconstruction, the contrast injection dynamics are all combined into one frame. Because of the averaging effect over time, the contrast amplitude is also much less than the one in the high frame-rate reconstruction. More artifacts can also be seen in the images, even in the post-contrast image, which might be due to the respiratory motion.

Figure 4.6 shows the corresponding MSLR decomposition of the high frame-rate reconstruction (full-length video at: <https://doi.org/10.6084/m9.figshare.7453232>). The global scale mainly captures the respiratory motion, whereas the finer scales mainly capture localized dynamics such as contrast injection dynamics in the heart and blood vessels. This matches the modelling assumption that the local block-wise bases can capture localized dynamics more compactly than the global block-wise bases.

Figure 4.7 shows the high frame-rate and low frame-rate reconstructions of the second DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453175>). In the high frame-rate reconstruction, the contrast injection dynamic from the left arm blood vessel can be visualized nicely. From the video, the contrast injection also caused an extreme bulk motion, creating severe artifacts afterwards. As soon as the body returned to the original position, the image quality improved, which is also seen in the post-contrast image. In the low frame-rate reconstruction, all frames with contrast injection dynamics were corrupted by the bulk motion. More artifacts can also be seen in the images, which might be due to the respiratory motion.

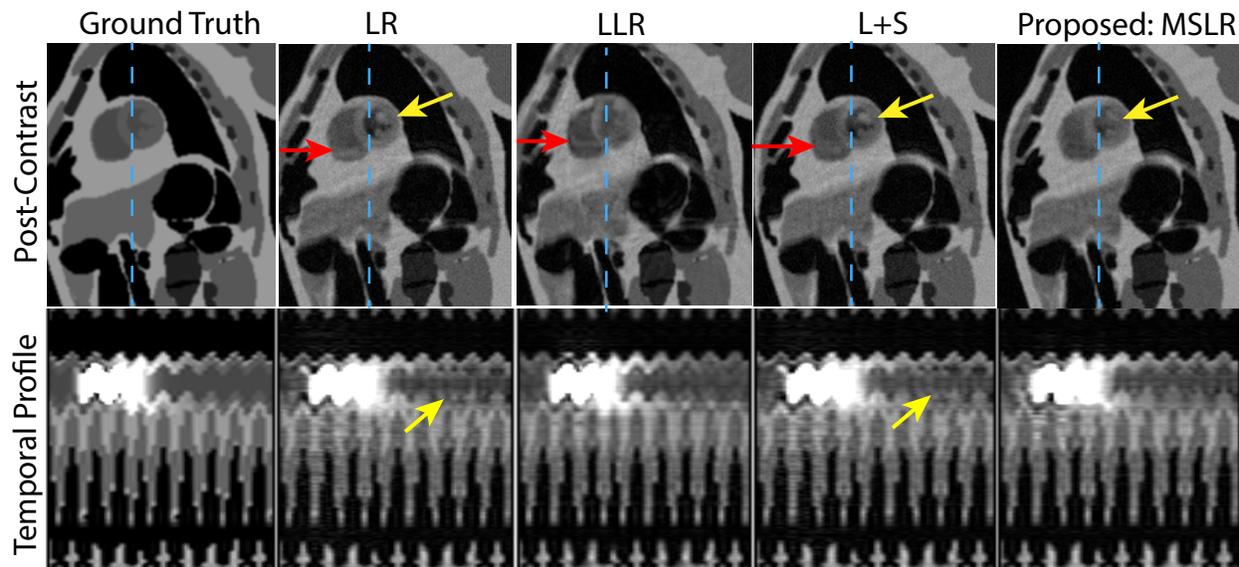


Figure 4.4: Reconstructions of the MRXCAT phantom using LR, LLR, L+S and the proposed MSLR representations (full-length video at: <https://doi.org/10.6084/m9.figshare.7466579>). Frames after contrast injection are shown, along with the temporal profiles indicated by the blue dashed lines. All methods display certain temporal artifacts, but overall LLR and MSLR show more accurate dynamics. The red arrows point to ghosting artifacts from other frames in all methods except the proposed MSLR. The yellow arrows point to contrast mismatch compared to the ground truth in all methods except LLR. Both LR and L+S also display blurring artifacts in the temporal profiles.

Pulmonary Imaging Acquired with the 3D UTE Radial Trajectory

Figure 4.8 shows a respiratory cycle excerpt of the first lung dataset (full length video at: <https://doi.org/10.6084/m9.figshare.7453229>). From the video, irregular breathing with variable rates can be observed. Despite this, the pulmonary structures, such as blood vessels, can be visualized clearly each cycle as seen in the figure. The result matches the modeling assumption that global and recurrent dynamics can be well represented with LR models. At the same time, the proposed reconstruction does not require an explicit estimate of the respiratory signal that is needed in gating reconstruction, and hence is robust to errors associated with it.

Figure 4.9 shows a respiratory cycle excerpt of the second lung dataset (full length video at: <https://doi.org/10.6084/m9.figshare.7453226>). From the video, coughing can be observed throughout the scan, resulting in poor image quality whenever it occurs. However, when the subject returns to a more regular breathing pattern, image quality also improves significantly. This shows that MSLR is able to localize the artifacts in time. Thus, the extracted images from a single respiratory cycle shown in the figure are able to show certain

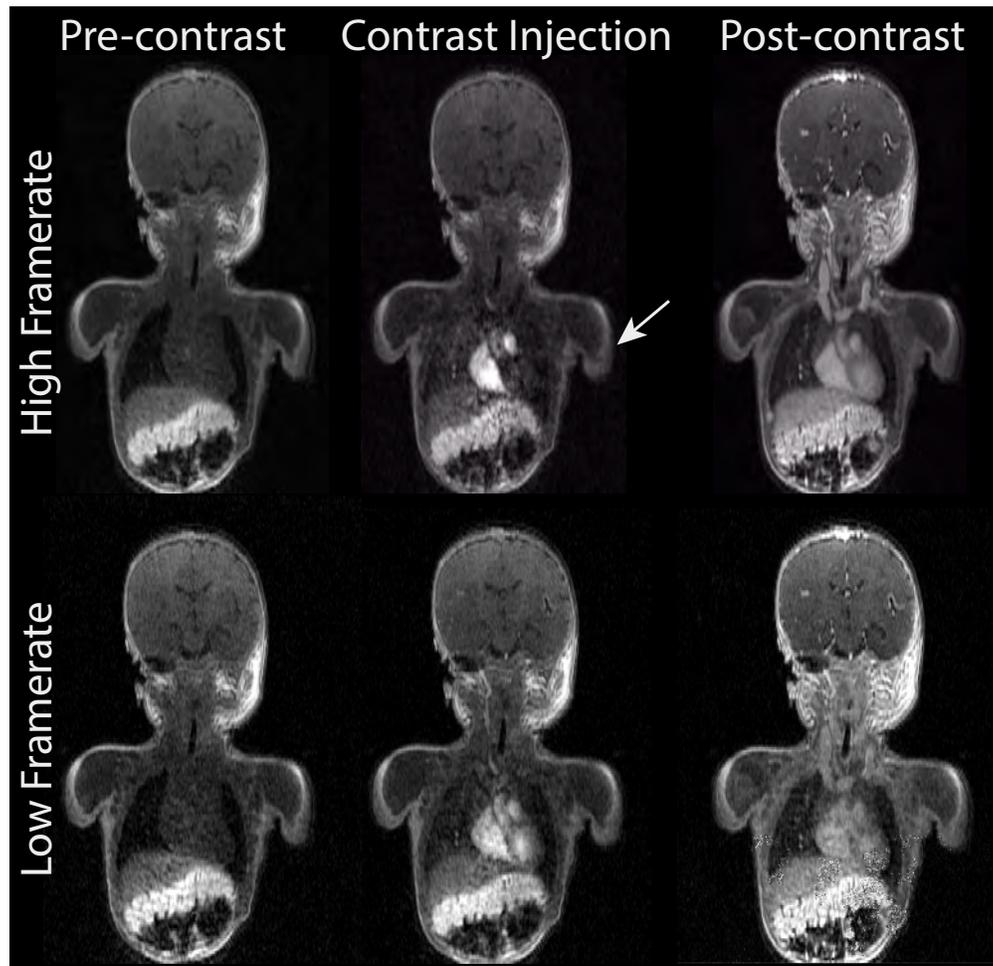


Figure 4.5: High frame-rate and low frame-rate reconstructions of the first DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453178>). In the high frame-rate reconstruction, contrast injection dynamics faster than respiration are nicely shown, such as those in the heart and in the blood vessels. From the video, slight bulk motion, pointed out by the white arrows, was observed when the contrast was injected. Respirator motion was also observed throughout the scan. In the low frame-rate reconstruction, the contrast injection dynamics were all combined into one frame. Because of the averaging effect over time, the contrast amplitude was also much less than the one in the high frame-rate reconstruction. More artifacts can also be seen in the images, even in the post-contrast image, which might be due to the respiratory motion.

detailed pulmonary structures, such as blood vessels.

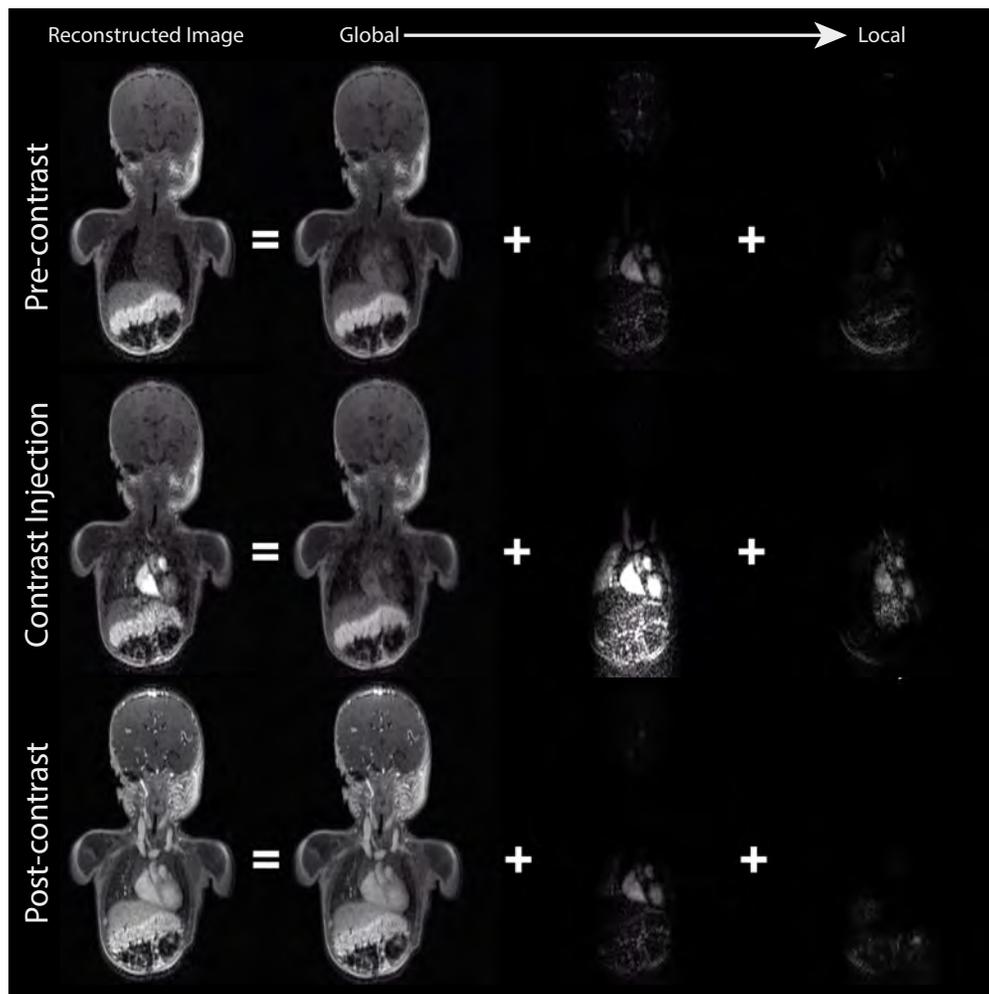


Figure 4.6: The MSLR decomposition of the DCE reconstruction shown in Figure 4.5 (full-length video at: <https://doi.org/10.6084/m9.figshare.7453232>). The global scale mainly captures the respiratory motion, whereas finer scales mainly capture localized dynamics such as contrast injection dynamics in the heart and blood vessels. This matches the expectation that the local block-wise bases can capture localized dynamics more compactly than the global block-wise bases.

4.7 Discussion

I have developed a method to reconstruct high spatiotemporal resolution 3D dynamic images from non-gated continuous acquisitions. The reconstruction problem considered is vastly underdetermined, and computationally and memory demanding. Using MSLR, the proposed method can greatly compress dynamic images on the order of 100 gigabytes to only a few gigabytes, as shown in Figure 4.6. Using the Burer-Monteiro heuristic, the reconstruction can directly optimize for the compressed representation, thereby allowing it to fit on GPU

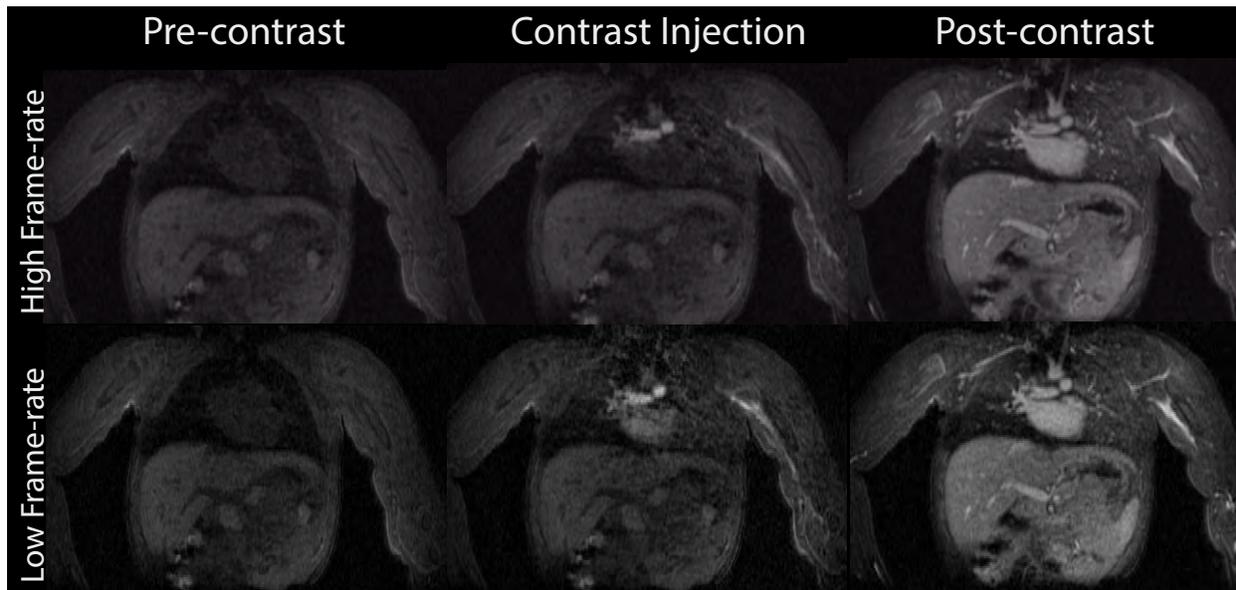


Figure 4.7: High frame-rate and low frame-rate reconstructions of the second DCE dataset (full-length video at: <https://doi.org/10.6084/m9.figshare.7453175>). In the high frame-rate reconstruction, the contrast injection dynamic from the left arm blood vessel can be visualized nicely. From the video, the contrast injection also caused a global shift of the subject body, creating severe artifacts afterwards. As soon as the body returned to the original position, the image quality improved, which is also seen in the post-contrast image in the figure. In the low frame-rate reconstruction, all frames with contrast injection dynamics were corrupted by the bulk motion. More artifacts can also be seen in the images, which might be due to the respiratory motion.

memory. Finally incorporating stochastic optimization, the runtime for the resulting method was further reduced.

In the simulation experiment comparing against LR, LLR and L+S, the proposed MSLR representation results in the highest PSNR, the least streaking artifacts, and in general more accurate dynamics. LLR also provides accurate dynamics, but suffers from low reconstruction PSNR, and higher level of streaking artifacts. This is because LLR does not leverage the global nature of the background tissues. L+S performs slightly better than LR in terms of PSNR, but both of them have more temporal blurring. Since MSLR incorporates LR, LLR, and L+S, it is able to combine the strengths of each method.

The experimental results show that localized dynamics faster than respiration can be visualized with the proposed reconstruction method. For example, Figures 4.5 and 4.7 and the associated videos show contrast dynamics in the heart and blood vessels that are faster than respiration in the high frame-rate reconstructions, whereas most of them are lost in the low frame-rate reconstructions. The MSLR decomposition in Figure 4.6 shows that these localized dynamics are mostly captured using the finer scales. Global and recurrent dynamics can also be visualized well with the proposed reconstruction. Figure 4.8 and the associated

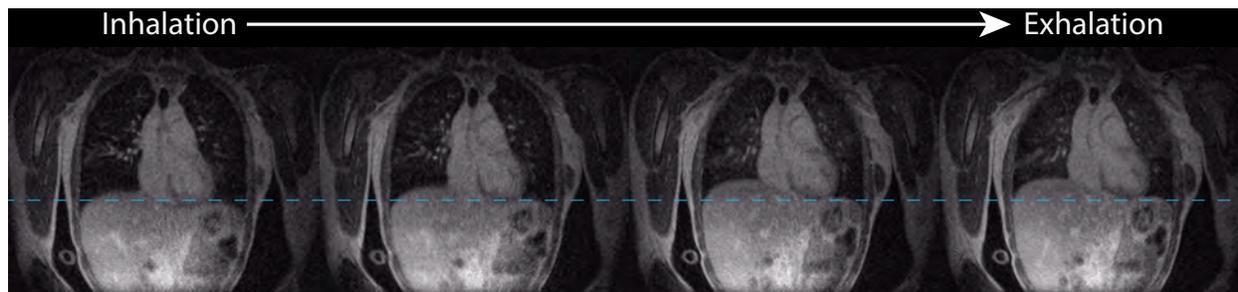


Figure 4.8: A respiratory cycle excerpt of the first lung dataset (full length video at: <https://doi.org/10.6084/m9.figshare.7453229>). From the video, irregular breathing with variable rates can be observed. Despite this, the pulmonary structures, such as blood vessels, can be visualized clearly when a cycle is extracted, as seen in the figure. The result matches the modeling assumption that global and recurrent dynamics can be well represented with LR models. At the same time, the proposed reconstruction does not require an explicit estimate of the respiratory signal that is needed in gating reconstruction, and hence is robust to errors associated with it.

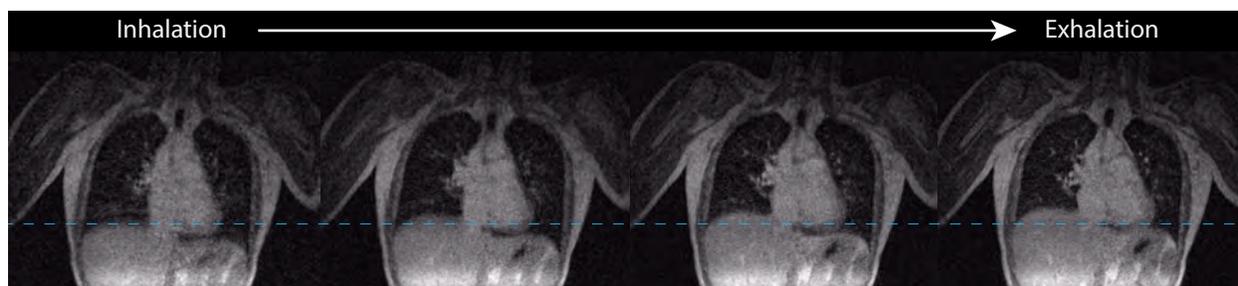


Figure 4.9: A respiratory cycle excerpt of the second lung dataset (full length video at: <https://doi.org/10.6084/m9.figshare.7453226>). From the video, coughing can be observed throughout the scan, resulting in poor image quality whenever it occurs. However, when the subject returns to a more regular breathing pattern, image quality also improves significantly. This shows that MSLR, though not able to resolve sudden motion like coughing, is able to localize the artifacts in time. Thus, the extracted images from a single respiratory cycle shown in the figure are able to show certain detailed pulmonary structures, such as blood vessels.

video shows detailed pulmonary structures despite of the irregular breathing rates. These types of motion are represented by the global LR component in the MSLR decomposition as demonstrated in Figure 4.6. Although gating methods might be able to handle variable rate breathing, the proposed reconstruction does not require an explicit estimate of the respiratory signal. Thus, it can be robust to errors associated with respiratory signal estimation.

However, bulk motion still poses a challenge for MSLR and results in artifacts in the reconstructions. In particular, the video for Figure 4.7 shows severe artifacts when the subject's body shifted entirely, and the video for Figure 4.9 displays noise-like artifacts

when the subject coughed. This is because MSLR inherently does not incorporate motion explicitly, and bulk motion is LR neither globally or locally. On the other hand, in Figure 4.7 and in the corresponding video, as soon as the body returns to the regular position, the image quality improves. Since MSLR exploits the time locality of the dynamics, even though the bulk motion cannot be fully resolved, it only affects nearby frames in the reconstruction. In comparison, with gating reconstruction, bulk motion can corrupt the entire respiratory phase, and even affect all phases with joint temporal reconstructions.

One might ask what is the finest temporal resolution one can get using the proposed method. In the experiments, 500 frames were used to provide a temporal resolution of about 0.5 seconds in order to visualize respiratory motion. I have tried 1000 frames, but the reconstruction quality remains similar, which indicates the temporal resolution for the proposed reconstruction has not been stress-tested yet. Fundamentally, the number of measurements has to be greater than the number of variables in the compressed representation. This sets a theoretical limit, which prevents reconstructing, for example, only one frame per one TR. However, such lower bound is also quite loose. Practically, like other compressed sensing reconstruction methods, the signal-to-noise ratio is the dominant factor that limits the temporal resolution. Of course, higher temporal resolution can be achieved if a lower spatial resolution can be tolerated.

4.8 Conclusion

I presented a method to reconstruct high spatiotemporal resolution 3D dynamic MRI data from continuous non-gated acquisitions. Transient dynamics, such as localized contrast dynamics, that are lost in low frame-rate reconstructions can be seen in my results. The proposed method also shows detailed pulmonary structures for respiratory motion at variable rates. Finally, the proposed reconstruction is robust to bulk motion by localizing the artifacts in time.

Chapter 5

Accelerating Convergence for Non-Cartesian MRI Reconstructions

5.1 Introduction

Non-Cartesian trajectories can offer many benefits based on the properties of each trajectory. Spiral [79, 80] and cones trajectories [66], for example, can be designed to traverse k-space very efficiently. Such rapid k-space coverage is suitable for fast imaging applications, including coronary imaging [80], and arterial spin labeled perfusion imaging [81]. Also, many non-Cartesian trajectories, such as radial [82] and projection reconstruction [83], naturally sample low-frequency regions densely. This can provide auto-calibration regions for parallel imaging (PI), and robustness to motion for dynamic applications. Finally, variable density sampling [84] property is more adapted to signal energy than uniform sampling. This can result in less coherent undersampling artifacts in the wavelet transform domain. Hence, variable density non-Cartesian trajectories are often used with compressed sensing (CS) [10].

On the other hand, reconstructions from non-Cartesian trajectories, especially with PI, are more complex and time-consuming than from Cartesian trajectories. The long reconstruction time is one reason that has limited the clinical adoption of non-Cartesian trajectories. In particular, because samples from non-Cartesian trajectories do not fall on a uniform grid, an inverse fast Fourier transform (FFT) cannot be used directly. The inverse discrete Fourier transform is often approximated using the gridding reconstruction, which involves a multiplication with a density compensation factor [85, 80, 86, 87] followed by the adjoint non-uniform fast Fourier transform (NUFFT). Other approximations include the BURS method [88]. However, these single-step techniques cannot be extended to leverage the additional multi-coil array encoding for arbitrary trajectories. Instead, iterative reconstructions, such as CG-SENSE [9], have to be used in general, which can often take many iterations to converge. In comparison, the Cartesian SENSE method [8] has an analytic solution that can be efficiently solved in a single step.

One way to make non-Cartesian imaging more efficient is reducing the number of iter-

ations in iterative reconstructions. In general, the slow convergence of iterative methods is due to the ill-conditioning of the reconstruction problem. For non-Cartesian imaging, such ill-conditioning comes from the variable density sampling distribution in k-space. This often shows up in images as blurring artifacts when the reconstruction has not yet converged. Slow convergence is even more significant for 3D acquisitions and CS reconstructions. For instance, Figure 5.1 shows the iteration progression for a ℓ_1 -wavelet regularized reconstruction of a 3D ultra-short echo-time (UTE) radial acquisition using the Fast Iterative Soft-Thresholding Algorithm (FISTA) [11], and primal dual hybrid gradient method (PDHG) [89] (also known as the Chambolle-Pock method). Even after 100 iterations, the reconstructed image still displays significant blurring due to slow convergence.

To compensate for slow convergence in non-Cartesian iterative reconstruction, a heuristic which uses density compensation during iterations is often used. Density compensation [85, 80, 86, 87] was originally developed for gridding reconstruction, and was mostly designed for Nyquist-sampled trajectories. The use of density compensation in iterative PI reconstruction was first introduced by Pruessmann et al. [9]. While their work showed that in practice density compensation can speed up convergence, reconstruction error is also increased. This is because the data consistency for densely sampled regions is weighted down in the objective function (more detail in Section 5.2).

An alternative to density compensation is preconditioning. Preconditioning has the advantage of preserving the original objective function and hence does not affect the reconstruction accuracy. However, a drawback of existing methods is that they increase the per-iteration computation. In particular, the use of preconditioning in MRI reconstruction was first described by Sutton et al. [90] for single-channel non-Cartesian imaging in the presence of field inhomogeneities. It was further explored by Ramani et al. [91] for PI-CS reconstructions. Their method leveraged a circulant preconditioner developed by Yagle [92] for Toeplitz systems. Weller et al. [93] considered the non-Cartesian ℓ_1 -SPIRiT [94] method, using an ℓ_2 -optimal circulant preconditioner developed by Chan [95]. Muckley et al. [96] considered FISTA [11] iterations and designed a circulant preconditioner that majorizes the sensing matrix motivated by the convergence criterion of FISTA. Koolstra et al. [97] considered the split-Bregman method for Cartesian PI-CS reconstructions and presented a circulant preconditioner that incorporates multi-channel sensitivity maps in the construction of their proposed preconditioner. All of the above preconditioners have circulant structures, and require at least two additional FFTs per iteration. Moreover, they all require inner loops in their algorithms for non-Cartesian reconstructions, which further lengthen the reconstruction time. Recently, a work of Trzasko et al. [98] showed that through an algebraic manipulation, a diagonal preconditioner can be applied in k-space for the least squares sub-problem of the alternating direction method of multiplier (ADMM) [48] method. This enables a different mechanism for preconditioning. In particular, they show that it is possible to use efficient operations in k-space for preconditioning. However, the formulation still required inner loops to solve for the sub-problem. Moreover, an off-the-shelf density compensation factor, which was not designed for preconditioning, was used as the preconditioner.

In this chapter, I present a method for speeding up convergence that combines the com-

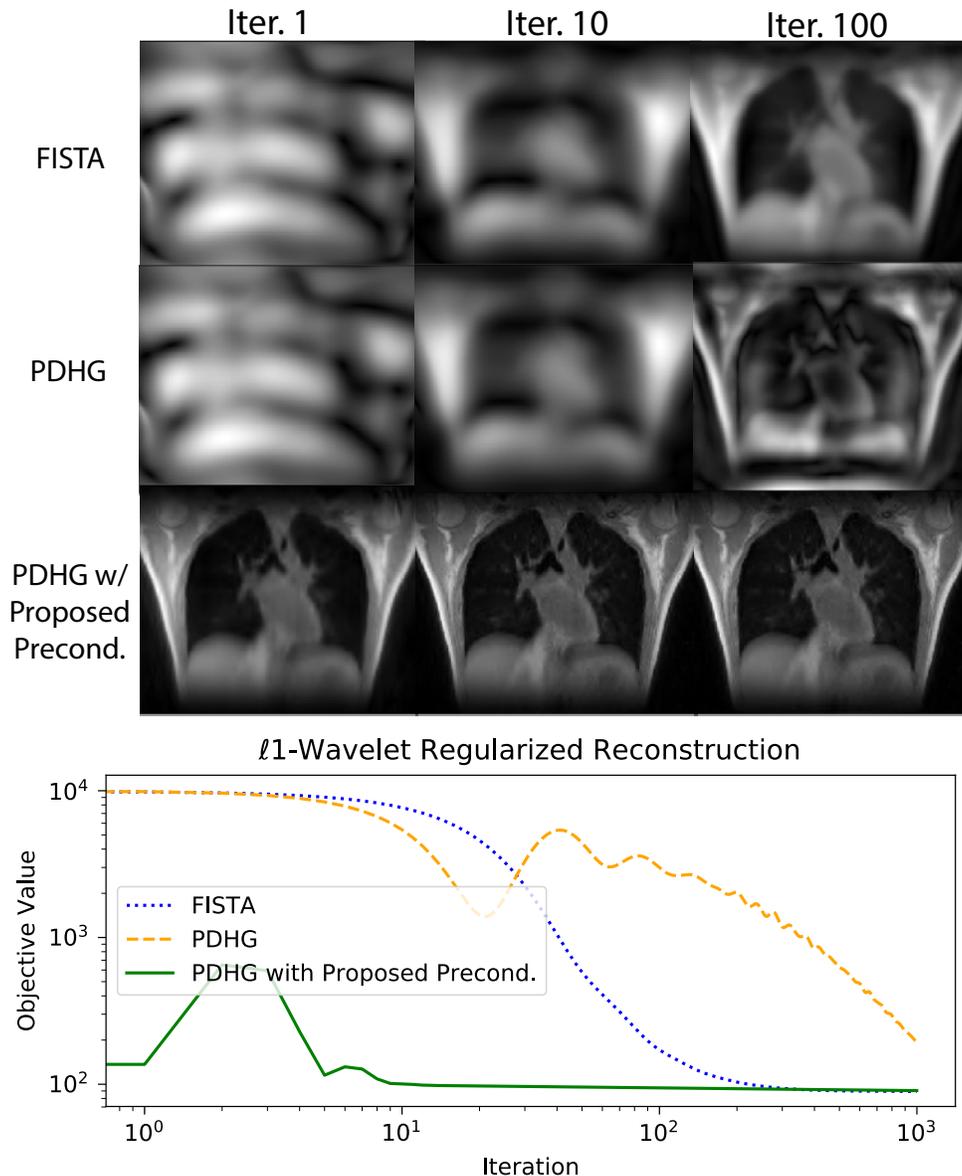


Figure 5.1: Iteration progression for ℓ_1 wavelet regularized reconstruction of a 3D UTE lung dataset. Both FISTA and PDHG exhibit extreme blurring even after 100 iterations. In contrast, PDHG with the proposed preconditioner converges in about ten iterations, both visually and quantitatively in terms of minimizing the objective value.

putational efficiency of density compensation, and the objective preserving property of preconditioning. Similar to the work of Trzakso et al. [98], I consider using efficient operations in k-space for preconditioning. My contribution is to recognize that a diagonal preconditioner can be applied in k-space more generally by viewing the objective function in the dual

formulation. In particular, the algebraic manipulation considered by Trzasko et al. [98] is found as a special case. Using PDHG [89], the resulting method with preconditioning does not have inner loops, so it has a similar computational complexity as the vanilla proximal gradient method. Moreover, instead of using off-the-shelf density compensation factors, I derive an ℓ_2 -optimized diagonal preconditioner for the multi-channel MRI forward model. I demonstrate through experiments that the proposed diagonal preconditioner speeds up iterative reconstruction for non-Cartesian imaging, with ℓ_2 -, ℓ_1 -wavelet, and total variation regularizations.

5.2 Problem Setup

I begin by formalizing the problem setup and illustrating the advantages and disadvantages of iterative reconstruction with density compensation and with existing preconditioning formulation.

Throughout this chapter, I consider the following discrete multi-channel MRI forward model, in which I am given an N -size image $\mathbf{x} \in \mathbb{C}^N$, a C -channel sensitivity maps $\mathbf{s} \in \mathbb{C}^{NC}$, a white Gaussian noise vector $\mathbf{w} \in \mathbb{C}^{MC}$, and k-space measurements $\mathbf{y} \in \mathbb{C}^{MC}$ with $\{f_i\}_{i=1}^M$ k-space sampling points such that

$$\mathbf{y}_{ic} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \mathbf{s}_{cn} \mathbf{x}_n e^{-i2\pi f_i n/N} + \mathbf{w}_{ic} \quad (5.1)$$

for $i \in \{1, \dots, M\}$, and $c \in \{1, \dots, C\}$. For simplicity, the mathematical notations will focus on one-dimensional signals. The above model can be succinctly represented as a linear model:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \quad (5.2)$$

Given the acquired k-space measurements \mathbf{y} , I consider the following regularized least squares problem to reconstruct the image:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + g(\mathbf{x}) \quad (5.3)$$

where $g(\mathbf{x})$ is the regularization function.

Since the size of the image \mathbf{x} is on the order of tens of thousands of pixels or more, the above reconstruction problem is in practice only solved approximately using first-order gradient methods. In the following of this section, I will focus on the proximal gradient method as an example to illustrate the advantages and disadvantages of using density compensation and preconditioners to accelerate convergence. I note that the concepts and issues shown here apply to other first-order methods, such as FISTA [11], and ADMM [48].

Concretely, the proximal gradient method when applied to objective function (5.3) gives the following update for the k th iteration:

$$\mathbf{x}^{k+1} = \text{prox}_{\alpha g}(\mathbf{x}^k - \alpha \mathbf{A}^H(\mathbf{A}\mathbf{x}^k - \mathbf{y})) \quad (5.4)$$

where $\text{prox}_{\alpha g}(\mathbf{z}) = \text{argmin}_{\mathbf{x}} \frac{1}{2\alpha} \|\mathbf{x} - \mathbf{z}\|_2^2 + g(\mathbf{x})$.

The convergence rate depends only on $\mathbf{A}^H \mathbf{A}$. More concretely, when \mathbf{A} is not singular, then the step-size α can be chosen so that the convergence rate is inversely proportional to the condition number of $\mathbf{A}^H \mathbf{A}$. When \mathbf{A} is singular, then the step-size can be chosen so that the convergence rate is inversely proportional to the maximum eigenvalue of $\mathbf{A}^H \mathbf{A}$. For variable density sampling, the condition number or the maximum eigenvalue of $\mathbf{A}^H \mathbf{A}$ is much higher than for uniform density sampling and hence results in slow convergence.

Density Compensation

One effective heuristic to accelerate convergence for non-Cartesian imaging is incorporating density compensation factors during iterations. Given a diagonal matrix $\mathbf{D} \in \mathbb{C}^{MC \times MC}$ with density compensation factor as diagonals, the heuristic modifies the proximal gradient method as follows:

$$\mathbf{x}^{k+1} = \text{prox}_{\alpha g}(\mathbf{x}^k - \alpha \mathbf{A}^H \mathbf{D}(\mathbf{A}\mathbf{x}^k - \mathbf{y})) \quad (5.5)$$

Computationally, incorporating density compensation in each iteration costs an additional $O(MC)$ multiplications, adding very little overhead to the overall iteration. However, the main drawback is that such k-space weighting is known to increase reconstruction errors, as implicitly it is solving for a weighted objective function:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{D}^{1/2}(\mathbf{A}\mathbf{x} - \mathbf{y})\|_2^2 + g(\mathbf{x}) \quad (5.6)$$

Note that data consistency is weighed down in densely sampled regions, so measurements are essentially thrown away for convergence, resulting in increased reconstruction error, and noise coloring.

Image-domain Preconditioning

An alternative is to use preconditioning, which only affects the convergence, but not the objective function. Since the objective function is not changed, there is no error penalty for using preconditioners. However, most existing preconditioning methods are applied on the variable directly, and hence in the image domain. Given a preconditioner $\mathbf{P} \in \mathbb{C}^{N \times N}$, the preconditioned proximal gradient method applies:

$$\mathbf{x}^{k+1} = \text{prox}_{\alpha g, \mathbf{P}}(\mathbf{x}^k - \alpha \mathbf{P} \mathbf{A}^H(\mathbf{A}\mathbf{x}^k - \mathbf{y})) \quad (5.7)$$

The preconditioner \mathbf{P} should be designed to approximate the (pseudo) inverse of $\mathbf{A}^H \mathbf{A}$ such that the condition number or maximum eigenvalue of $\mathbf{P} \mathbf{A}^H \mathbf{A}$ is much lower than that of $\mathbf{A}^H \mathbf{A}$. However, in order to compensate ill conditioning from variable density in k-space, existing preconditioners have to go to the Fourier domain, and use circulant operators, which cost two additional Fast Fourier transforms (FFT) per iteration. That is, existing preconditioners are of the form,

$$\mathbf{P} = \mathbf{F} \text{diag}(\mathbf{p}) \mathbf{F}^H \quad (5.8)$$

where $\mathbf{p} \in \mathbb{C}^N$ is a Fourier weighting vector, and $\mathbf{F} \in \mathbb{C}^{N \times N}$ is the unitary discrete Fourier transform operator.

A more subtle issue is that the proximal operator has to be modified to incorporate the preconditioner, which often requires inner iterations to solve even when the proximal operator is simple. In particular, the proximal operator becomes:

$$\text{prox}_{\alpha g, \mathbf{P}}(\mathbf{z}) = \underset{\mathbf{x}}{\text{argmin}} \frac{1}{2\alpha} \|\mathbf{P}^{-1/2} \mathbf{x} - \mathbf{z}\|_2^2 + g(\mathbf{x}) \quad (5.9)$$

which amounts to another optimization problem that in general is not simple to solve (for example when g is the ℓ_1 -norm).

In summary, although existing preconditioners have shown that they can accelerated convergence, their shortcoming lies in the per-iteration increase in complexity.

5.3 k-space Preconditioning

Ideally, I would like to develop a preconditioning method that can achieve the computational efficiency of density compensation without changing the objective function. Here I show that this is achievable by looking at the convex dual problem.

In particular, since the reconstruction problem (5.3) is unconstrained, it must satisfy strong duality. Its corresponding dual problem (see Appendix 5.8 for a derivation using the augmented Lagrangian) is given by:

$$\max_{\mathbf{u}} - \left(\frac{1}{2} \|\mathbf{u}\|_2^2 - \langle \mathbf{u}, \mathbf{y} \rangle + g^*(-\mathbf{A}^H \mathbf{u}) \right)$$

where $\mathbf{u} \in \mathbb{C}^m$ is the dual variable. Note that the dual variable resides in k-space, which now enables performing preconditioning in k-space.

Of course, solving for the dual problem does not automatically solve the primal problem. Instead, the primal and dual variables \mathbf{x} and \mathbf{u} are connected with the following relationship:

$$\begin{aligned} -\mathbf{A}^H \mathbf{u} &\in \partial g(\mathbf{x}) \\ \mathbf{A} \mathbf{x} &= \mathbf{u} + \mathbf{y} \end{aligned} \quad (5.10)$$

In general, the above relationship does not find a primal solution automatically using the dual solution, and one requires primal-dual methods that can solve for the primal and dual problems at the same time. However, it turns out that ℓ_2 -regularized reconstruction is a special case that can efficiently recover the primal variable from the dual.

ℓ_2 -regularized Reconstruction

Let me consider $g(\mathbf{x}) = \frac{\lambda}{2}\|\mathbf{x}\|_2^2$, then the dual problem is given by,

$$\max_{\mathbf{u}} - \left(\frac{1}{2}\|\mathbf{u}\|_2^2 - \langle \mathbf{u}, \mathbf{y} \rangle + \frac{1}{2\lambda}\|\mathbf{A}^H\mathbf{u}\|_2^2 \right)$$

which has the optimality condition:

$$(\mathbf{A}\mathbf{A}^H + \lambda\mathbf{I})\mathbf{u} = \lambda\mathbf{y}$$

Hence, I can precondition k-space by preconditioning the dual variable by solving:

$$\mathbf{P}(\mathbf{A}\mathbf{A}^H + \lambda\mathbf{I})\mathbf{u} = \lambda\mathbf{P}\mathbf{y}$$

Since $\partial g(\mathbf{x}) = \{\lambda\mathbf{x}\}$, from the primal dual relationship (5.10) I can recover the primal variable by performing,

$$\mathbf{x} = \frac{1}{\lambda}\mathbf{A}^H\mathbf{u}$$

The above method is precisely what Trzasko et al. [98] proposed for the ℓ_2 -regularized sub-problem within ADMM. Here I rederive it through convex duality. While still requiring inner loops for general regularization functions other than ℓ_2 -regularization, this formulation allows me to precondition using density compensation like operations.

General case: Primal-Dual Hybrid Gradient Method

For the general case, to derive a method for solving for the primal and dual problem simultaneously without inner loops, I opt for the PDHG [89] method. I note that other primal-dual reconstruction methods, such as those described in the work of Komodakis et al. [99], can also be used.

Following [89] and [100], for each iteration k , the preconditioned version of PDHG for simple proximal operators is given by,

$$\begin{aligned} \mathbf{u}^{k+1} &= (\mathbf{I} + \sigma^k\mathbf{P})^{-1}(\mathbf{u}^k + \mathbf{P}\sigma^k(\mathbf{A}\bar{\mathbf{x}}^k - \mathbf{y})) \\ \mathbf{x}^{k+1} &= (\mathbf{I} + \tau^k\partial g)^{-1}(\mathbf{x}^k - \tau^k\mathbf{A}^H\mathbf{u}^{k+1}) \\ \bar{\mathbf{x}}^{k+1} &= \mathbf{x}^{k+1} + \theta^k(\mathbf{x}^{k+1} - \mathbf{x}^k) \end{aligned}$$

where $\bar{\mathbf{x}}^k$ and θ^k are the extrapolated primal variable and extrapolation parameter to provide acceleration. τ^k and σ^k are the primal and dual step-size respectively such that

$$\sigma^k\tau^k\lambda_{\max}(\mathbf{P}\mathbf{A}\mathbf{A}^H) < 1$$

Since the data consistency function is smooth, acceleration can be obtained by choosing step-sizes appropriately, following [89].

For total variation regularization, which has the form:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{y}\|_2^2 + g(\mathbf{Gx})$$

The PDHG can be modified to perform:

$$\begin{aligned} \mathbf{u}^{k+1} &= (\mathbf{I} + \sigma^k \mathbf{P})^{-1} (\mathbf{u}^k + \mathbf{P} \sigma^k (\mathbf{A} \bar{\mathbf{x}}^k - \mathbf{y})) \\ \mathbf{v}^{k+1} &= (\mathbf{I} + \sigma^k \partial g)^{-1} (\mathbf{v}^k + \sigma^k \mathbf{G} \bar{\mathbf{x}}^k) \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - \tau^k (\mathbf{A}^H \mathbf{u}^{k+1} + \mathbf{G}^H \mathbf{v}^{k+1}) \\ \bar{\mathbf{x}}^{k+1} &= \mathbf{x}^{k+1} + \theta^k (\mathbf{x}^{k+1} - \mathbf{x}^k) \end{aligned}$$

where τ^k and σ^k are the primal and dual step-size respectively such that

$$\sigma^k \tau^k \lambda_{\mathbf{Ax}} (\mathbf{PAA}^H + \mathbf{GG}^H) < 1$$

5.4 L2 optimized diagonal k-space preconditioner

Now that I know how to precondition in k-space, it becomes clear from the convergence criterion that the preconditioner should be designed to precondition the matrix \mathbf{AA}^H . In this chapter, I consider a diagonal preconditioner to approximate the inverse of the normal operator \mathbf{AA}^H in the least squares sense. The diagonal structure is desired because I want to be able to apply the preconditioner efficiently in k-space, similarly to density compensation. The least squares design, on the other hand, is used here for efficiently computing the preconditioner.

Concretely, I consider a Fourier preconditioner $\mathbf{P} = \text{diag}(\mathbf{v})$, $\mathbf{v} \in \mathbb{C}^{MC}$ such that,

$$\mathbf{v} = \text{argmin}_{\mathbf{v}} \|\text{diag}(\mathbf{v}) \mathbf{AA}^H - \mathbf{I}\|_F^2$$

Let $\mathbf{a}_i \in \mathbb{C}^N$ denote the i th row vector of \mathbf{A} . As shown in Appendix 5.9, the general expression for the inverse of the diagonal preconditioner is given by:

$$\mathbf{v}_i^{-1} = \frac{\sum_{j=1}^M |\mathbf{a}_i^H \mathbf{a}_j|^2}{\|\mathbf{a}_i\|_2^2}$$

Note that this diagonal preconditioner and its inverse are always defined, under the reasonable assumption that none of the row vectors \mathbf{a}_i are zeros.

To further look into the preconditioner, I first consider the single-channel case. In this case, $\mathbf{a}_{in} = \frac{1}{\sqrt{N}} e^{-i2\pi f_i n/N}$, and $\|\mathbf{a}_i\|_2^2 = 1$. Then the diagonal preconditioner at k-space

position i , is given by:

$$\begin{aligned} \mathbf{v}_i^{-1} &= \frac{1}{N} \sum_{j=1}^M \left| \sum_{n=0}^{N-1} e^{-i2\pi(f_i-f_j)n/N} \right|^2 \\ &= \frac{1}{N} \sum_{j=1}^M \left| \frac{\sin(\pi(f_i-f_j))}{\sin(\pi(f_i-f_j)/N)} \right|^2 \end{aligned}$$

For Cartesian trajectories, the frequency spacing $f_i - f_j$ are all integers, and hence $\mathbf{v} = 1$, which matches our expectation that single channel Cartesian reconstruction does not require preconditioning. For non-Cartesian trajectories, the diagonal preconditioner can be interpreted as calculating density from the sinc squared kernel $\frac{1}{N} \left| \frac{\sin(\pi f)}{\sin(\pi f/N)} \right|^2$.

Moving on to multi-channel, for k-space position i and coil c , the row vector is given by $\mathbf{a}_{icn} = \frac{1}{\sqrt{N}} \mathbf{s}_{cn} e^{-i2\pi f_i n/N}$. Hence, I obtain,

$$\mathbf{v}_{ic}^{-1} = \frac{1}{\|\mathbf{s}_c\|_2^2 N} \sum_{j=1}^M \sum_{c'=1}^C \left| \sum_{n=0}^{N-1} \mathbf{s}_{cn} \mathbf{s}_{c'n}^* e^{-i2\pi(f_i-f_j)n/N} \right|^2$$

Incorporating coil sensitivity maps allows me to precondition the problem more effectively than without them, as I address the forward model directly. On the other hand, I also note that one downside is that the proposed preconditioner has to be recalculated whenever the coil sensitivity maps change. For many clinical applications, the coil sensitivity maps are calculated from the pre-scan or estimated from the first scan and used multiple times for a sequence of scans. In this case, the overhead of computing the preconditioner becomes small. This is the case I consider here. For applications in which this overhead matters, the single-channel preconditioner may be used instead, which can be pre-computed. However, I will not explore this in this work.

Since the preconditioner has to be computed whenever the coil sensitivity maps change, its computation time matters. A direct summation implementation takes $O(M^2 N C^2)$ computation. In the following, I show that using Fourier transform properties, I can reduce the computational complexity to $O(C^2 N \log N + CM)$, which makes it comparable to common calibration methods, such as ESPIRiT [101]. Figure 5.2 provides a high-level diagram of the overall process.

Efficient computation of the proposed preconditioner

First, I note that I can express the squared terms with cross-correlations, which can be computed in $O(C^2 N \log N)$ using FFTs. Let me define,

$$\mathbf{r}_{cc'}[k] = \sum_{\substack{n, n': \\ n-n'=k}} (\mathbf{s}_{cn} \mathbf{s}_{c'n}^*)^* (\mathbf{s}_{cn'} \mathbf{s}_{c'n'}^*)$$

Then

$$\left| \sum_{n=0}^{N-1} \mathbf{s}_{cn} \mathbf{s}_{c'n}^* e^{-i2\pi(f_i - f_j)n/N} \right|^2 = \sum_{k=-N+1}^{N-1} \mathbf{r}_{cc'}[k] e^{-i2\pi(f_i - f_j)k/N}$$

Next, I note that the preconditioner can be expressed in terms of convolution with the point spread function, which can be computed approximately using NUFFT with $O(N \log N + M)$ computational complexity. Let me define

$$\mathbf{h}[k] = \frac{1}{\sqrt{N}} \sum_{j=1}^M e^{i2\pi f_j k/N}$$

Then

$$\begin{aligned} \mathbf{v}_{ic}^{-1} &= \frac{1}{\|\mathbf{s}_c\|_2^2 N} \sum_{j=1}^M \sum_{k=-N+1}^{N-1} \sum_{c'=1}^C \mathbf{r}_{cc'} k e^{-i2\pi(f_i - f_j)k/N} \\ &= \frac{1}{\|\mathbf{s}_c\|_2^2 \sqrt{N}} \sum_{k=-N+1}^{N-1} \sum_{c'=1}^C \mathbf{r}_{cc'} k \left(\frac{1}{\sqrt{N}} \sum_{j=1}^M e^{i2\pi f_j k/N} \right) e^{-i2\pi f_i k/N} \\ &= \frac{1}{\|\mathbf{s}_c\|_2^2 \sqrt{N}} \sum_{k=-N+1}^{N-1} \sum_{c'=1}^C \mathbf{r}_{cc'} k \mathbf{h}[k] e^{-i2\pi f_i k/N} \end{aligned}$$

The final step involves C NUFFTs on the pointwise multiplication of \mathbf{r} , and \mathbf{h} . Hence the overall computational complexity is $O(C^2 N \log N + CM)$.

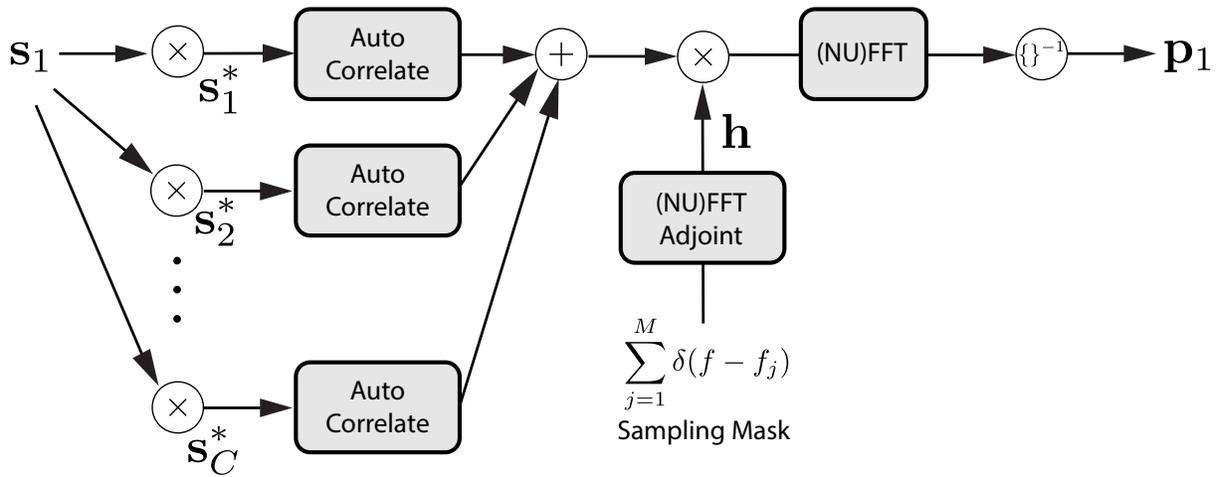


Figure 5.2: Diagram of computing the proposed diagonal k-space preconditioner for the first channel.

5.5 Experiments

In the spirit of reproducible research, I provide a software package in Python to reproduce the results described in this chapter. The software package can be downloaded from:

https://github.com/mikgroup/kSPACE_precond.git

I evaluated the proposed method with three regularization functions: ℓ_2 -norm, ℓ_1 -wavelet, and total variation. For each regularization, I evaluated on three 2D non-Cartesian datasets: a liver dataset acquired with stack-of-stars trajectory, a brain dataset acquired with ramp-sampled UTE radial trajectory, and a cardiac dataset with variable density spiral trajectory. I also applied on one 3D UTE dataset to illustrate the additional benefit of using preconditioners on 3D datasets. These datasets are described in more detail in Section 5.5.

For ℓ_2 -regularized reconstruction, conjugate gradient (CG) with and without preconditioner from Koostra et al., and PDHG with and without the proposed preconditioner were applied and compared with $\lambda = 0.01$. I note that Koostra et al. originally proposed their method for Cartesian imaging, and I extended it to the non-Cartesian case by going through the same derivation.

For ℓ_1 -wavelet regularized reconstruction, FISTA and PDHG with and without the proposed preconditioner were applied and compared with $\lambda = 0.001$.

For total variation regularized reconstruction, PDHG with and without the proposed preconditioner were applied and compared $\lambda = 0.001$.

All methods were implemented in Python using the software packages NumPy [102], and CuPy [103], on a workstation with four Nvidia Titan Xp GPUs. All operations, except the wavelet transform, were run on a single GPU. All methods were run for 1000 iterations, and the objective values were computed for each iteration. Also for ℓ_2 regularized reconstruction, per iteration computation time for all methods was recorded and averaged over 1000 iterations. The computation time for constructing the Koostra et al.'s preconditioner and the proposed preconditioner was also recorded.

Dataset Details

The liver dataset was acquired with a stack-of-stars trajectory using a 3D T1-FFE sequence (TR/TE 4.35 ms 1.20ms, resolution $1 \times 1 \times 1.5 \text{ mm}^3$, field-of-view $40 \times 40 \times 12.5 \text{ cm}^3$). The sequence was implemented on a 3T MR system (Philips Healthcare) equipped with a 16-channel torso coil. The center slice was extracted after taking an inverse FFT along the slice direction for the experiments.

The cardiac dataset was acquired with a variable density spiral trajectory on a 1.5 T GE scanner (GE Healthcare, Waukesha, WI) with an 8-channel cardiac coil and the HeartVista RTHawk platform (HeartVista, Los Altos, CA). The trajectory consists of 3 interleaves. It has a matrix size of 360×360 and TR of 25.8 ms.

The brain dataset was acquired with a centered-out radial trajectory on a 7.0 T GE clinical scanner (GE Healthcare, Waukesha, WI) with 8-channel head coil. The following

prescribed parameters were used: flip angle of 5 degree, field-of-view $20 \times 20 \text{ cm}^2$, in-plane resolution $1 \times 1 \text{ mm}^2$, and $\text{TE/TR} = 3.4 \text{ ms}/2 \text{ seconds}$.

The 3D UTE dataset was acquired with a bit reversed ordered radial trajectory. [5]. The following prescribed parameters were used: FOV of $32 \times 32 \times 32 \text{ cm}^3$, flip angle of 4 degrees, 1.25 mm isotropic resolution, sampling bandwidth of 62.5 kHz, and readout duration of 1 ms. 75,800 spokes were acquired.

Results

Figure 5.3 shows the iteration progression for the ℓ_2 regularized reconstruction of the liver dataset, comparing CG with and without Koolstra et al.’s preconditioning, and PDHG with and without the proposed preconditioning. Both visually and quantitatively in terms of objective value, methods with preconditioning converge faster than the non-preconditioned counterparts, in less than ten iterations. Although in this case, the proposed method converges faster than CG with Koolstra et al.’s preconditioner, there are other cases shown in Figures 5.4 and 5.5, showing the opposite way. In general, the proposed preconditioner perform similarly to Koolstra et al.’s preconditioner, while consistently improving the convergence of non-preconditioned methods.

Table 5.1 shows the per-iteration computation time for ℓ_2 -regularized reconstruction experiments. CG with Koolstra et al.’s preconditioner is consistently the slowest, as it requires two additional FFT’s per iteration. The computation time for PDHG with the proposed preconditioner is comparable to CG and PDHG. Table 5.2 shows the computation time for constructing the preconditioners. The construction of the proposed preconditioner is about twice as slow as constructing Koostra et al. ’s preconditioner, as theirs grows linearly proportional to the number of coils. However, I emphasize that I am considering applications in which the coil sensitivity maps are calculated from the pre-scan or estimated from the first scan and used multiple times for a sequence of scans.

Table 5.1: Per-iteration computation time

	Liver	Cardiac	Brain
CG	0.0792 s	0.0223 s	0.0308 s
CG with Koostra et al.’s precondition.	0.0917 s	0.0314 s	0.0368 s
PDHG	0.0814 s	0.0210 s	0.0311 s
PDHG with proposed precondition.	0.0810 s	0.0228 s	0.0313 s

Table 5.2: Computation time for constructing preconditioners

	Liver	Cardiac	Brain
Koostra et al.’s precondition.	0.0974 s	0.0502 s	0.0147 s
Proposed precondition.	0.231 s	0.117 s	0.0334 s

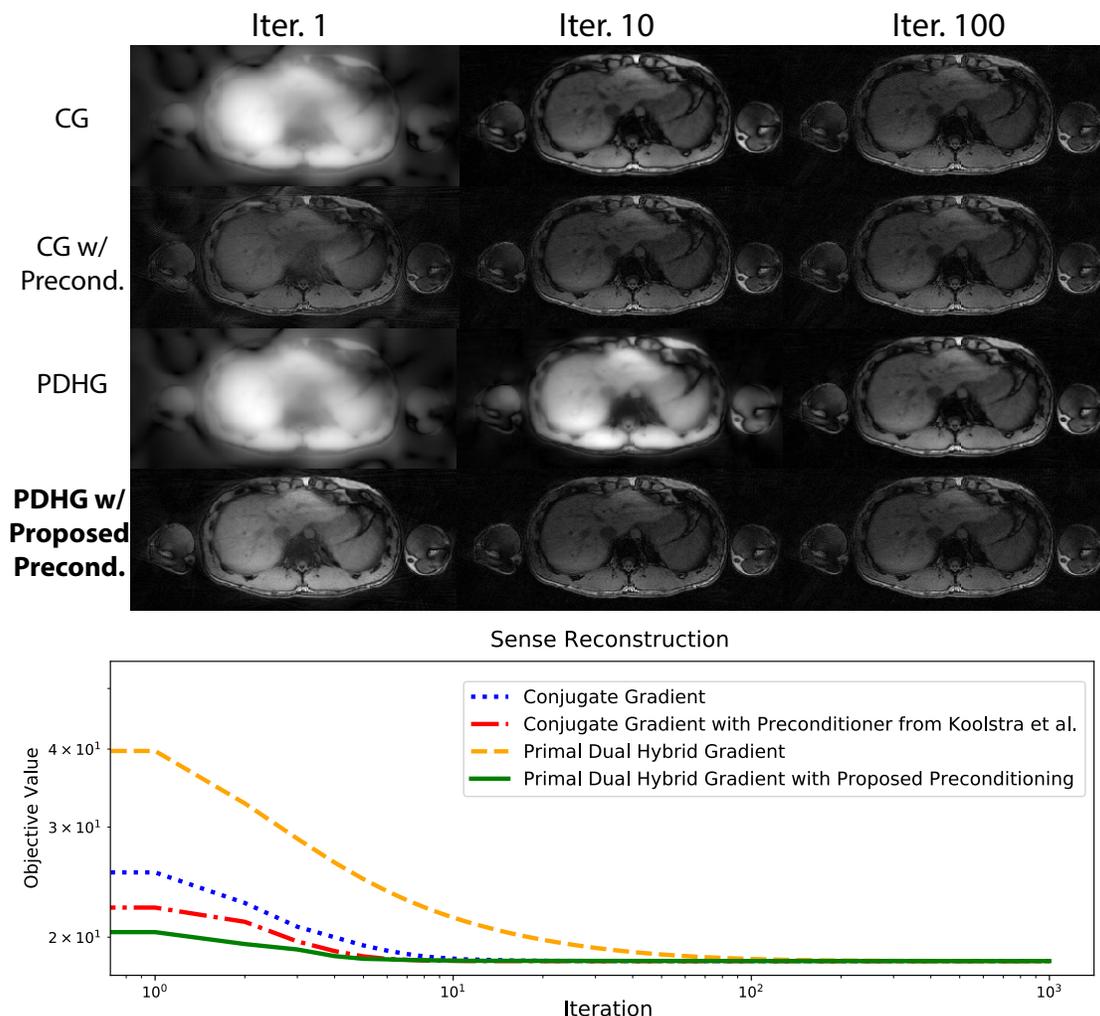


Figure 5.3: Iteration progression for ℓ_2 regularized reconstruction of the liver dataset, comparing CG with and without Koolstra et al.’s preconditioning, and PDHG with and without the proposed preconditioning. Both visually and quantitatively in terms of objective value, methods with preconditioning converge faster than the non-preconditioned counterparts. Both preconditioned methods converge in less than ten iterations.

Figure 5.6 shows the iteration progression for ℓ_1 -wavelet regularized reconstruction of the cardiac dataset, comparing FISTA and PDHG with and without the proposed preconditioning. Again, both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about ten iterations. Other experiments shown in Figures 5.7 and 5.8 support this as well.

Figure 5.9 shows the iteration progression for total variation regularized reconstruction of the brain dataset, comparing PDHG with and without the proposed preconditioning. Both

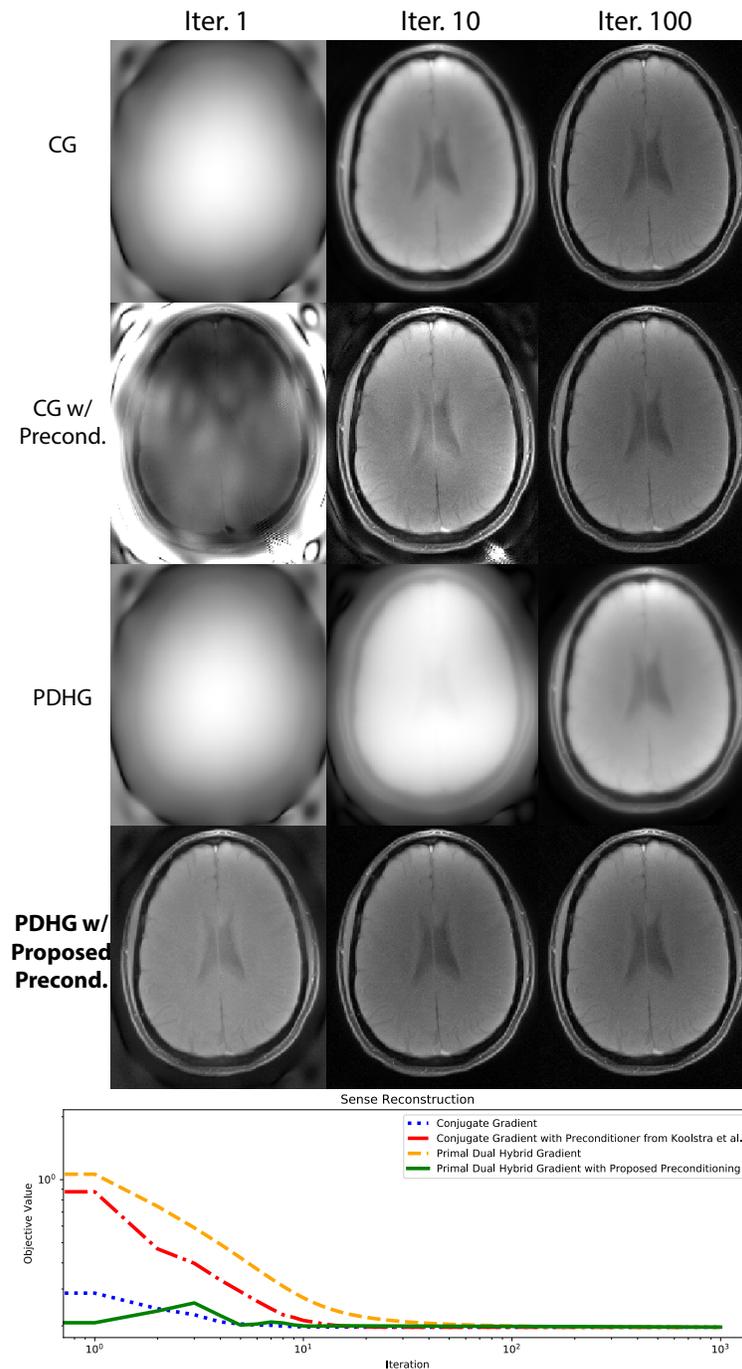


Figure 5.4: Iteration progression for ℓ_2 -regularized reconstruction of the brain dataset, comparing CG with and without Koolstra et al.s preconditioning, and PDHG with and without the proposed preconditioning

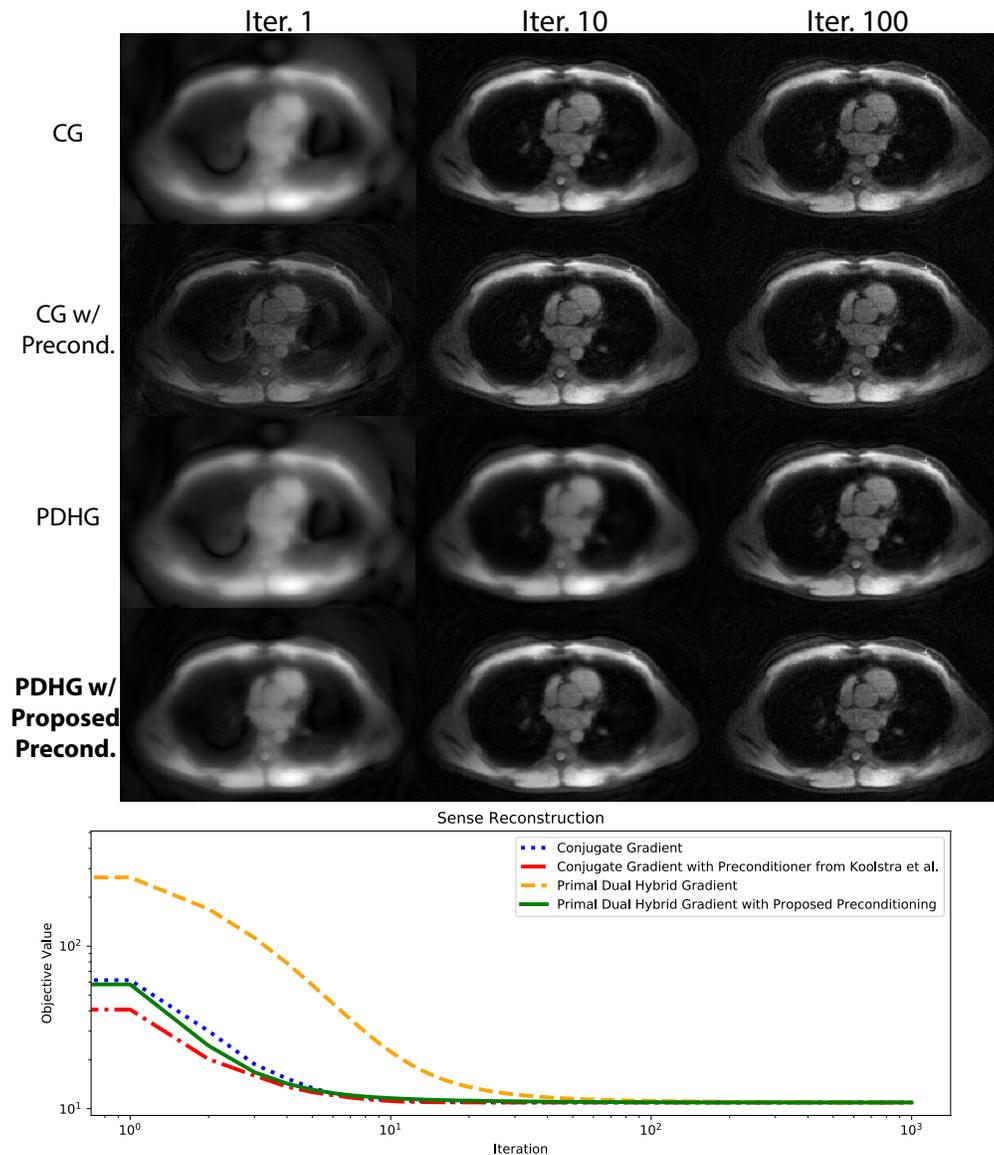


Figure 5.5: Iteration progression for ℓ_2 -regularized reconstruction of the cardiac dataset, comparing CG with and without Koolstra et al.s preconditioning, and PDHG with and without the proposed preconditioning

methods converged slower compared to the two other reconstruction experiments. Both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about 30 iterations. Other experiments shown in Figures 5.10 and 5.11 support this as well.

Finally, the iteration progression for the 3D UTE dataset was shown earlier in Figure 5.1.

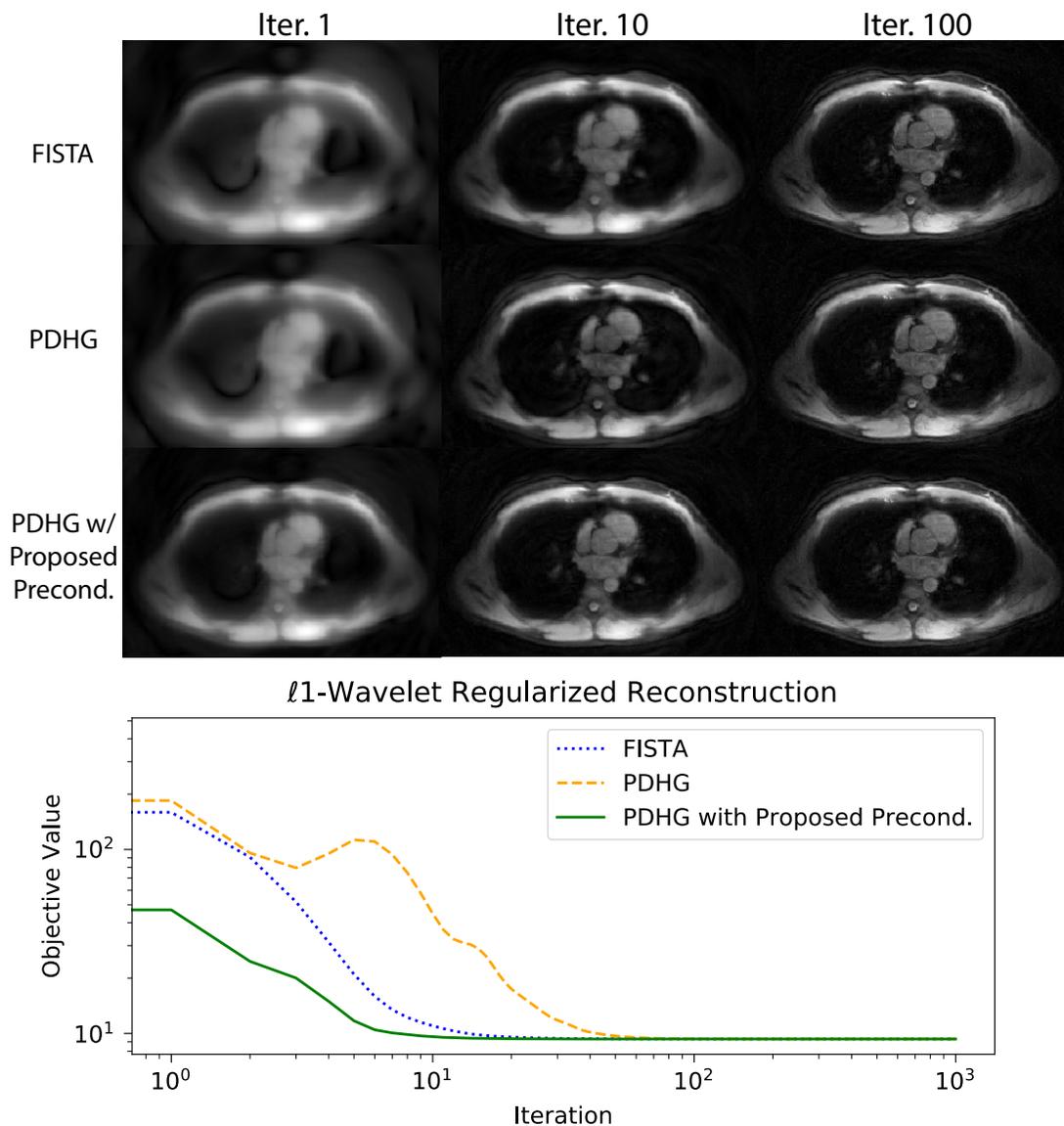


Figure 5.6: Iteration progression for ℓ_1 wavelet regularized reconstruction of the cardiac dataset, comparing FISTA and PDHG with and without the proposed preconditioning. Again, both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about ten iterations.

Both FISTA and PDHG exhibit extreme blurring even after 100 iterations. In contrast, PDHG with the proposed preconditioner converges in about ten iterations, both visually and quantitatively in terms of minimizing the objective value. This shows that the proposed method can offer an order magnitude speedup in 3D than in 2D.

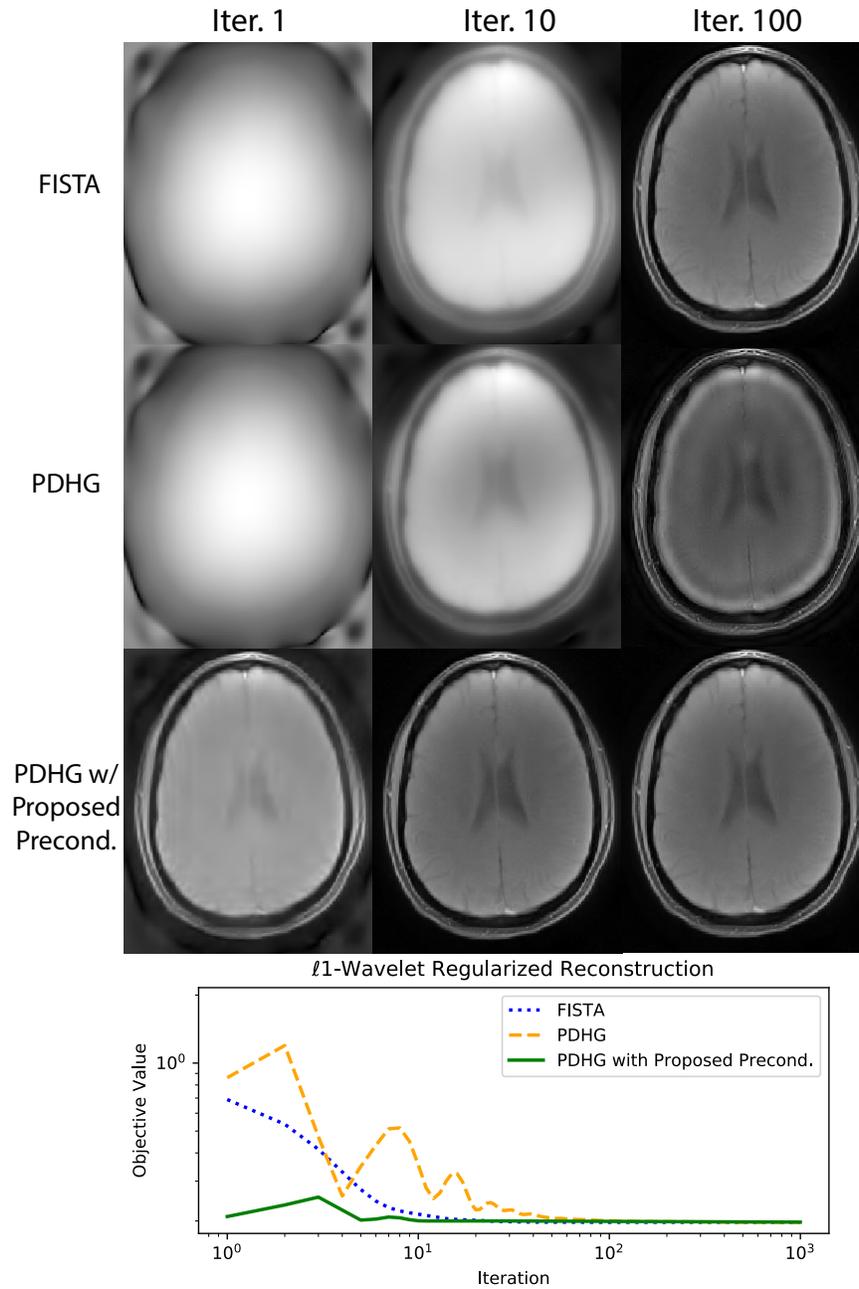


Figure 5.7: Iteration progression for ℓ_1 wavelet regularized reconstruction of the brain dataset, comparing FISTA and PDHG with and without the proposed preconditioning.

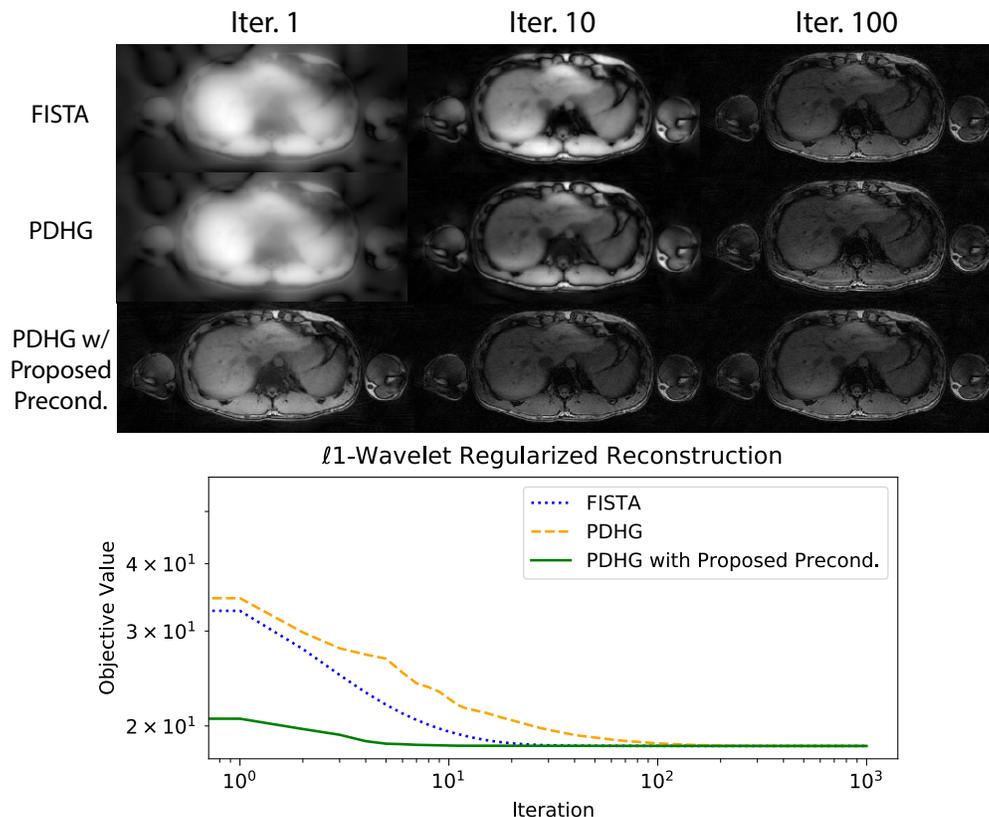


Figure 5.8: Iteration progression for ℓ_1 wavelet regularized reconstruction of the liver dataset, comparing FISTA and PDHG with and without the proposed preconditioning.

5.6 Discussion

In this chapter, I presented a preconditioning method through the convex dual formulation. This enables the use of efficient k-space operations as preconditioners and does not modify the objective function. Through experiments, I have demonstrated that in practice the proposed technique accelerates convergence of non-Cartesian reconstruction.

In particular, I compared the performance of the proposed preconditioning to that of Koostra et al. for ℓ_2 -regularized reconstructions. In terms of convergence, the proposed preconditioning performs similarly to Koolstra's et al.'s method: In some datasets, such as the one shown in Figure 5.3, the proposed method converged faster. For others, Koostra et al.'s converged faster. Both preconditioning techniques improved the convergence for CG and PDHG. The main advantage of the proposed preconditioning lies in the per-iteration computation time shown in Table 5.1. The proposed method is much faster than Koostra et al.'s preconditioning, and performs similarly as CG in terms of per-iteration time. This is expected as the circulant preconditioning requires two additional FFT's per iteration, whereas

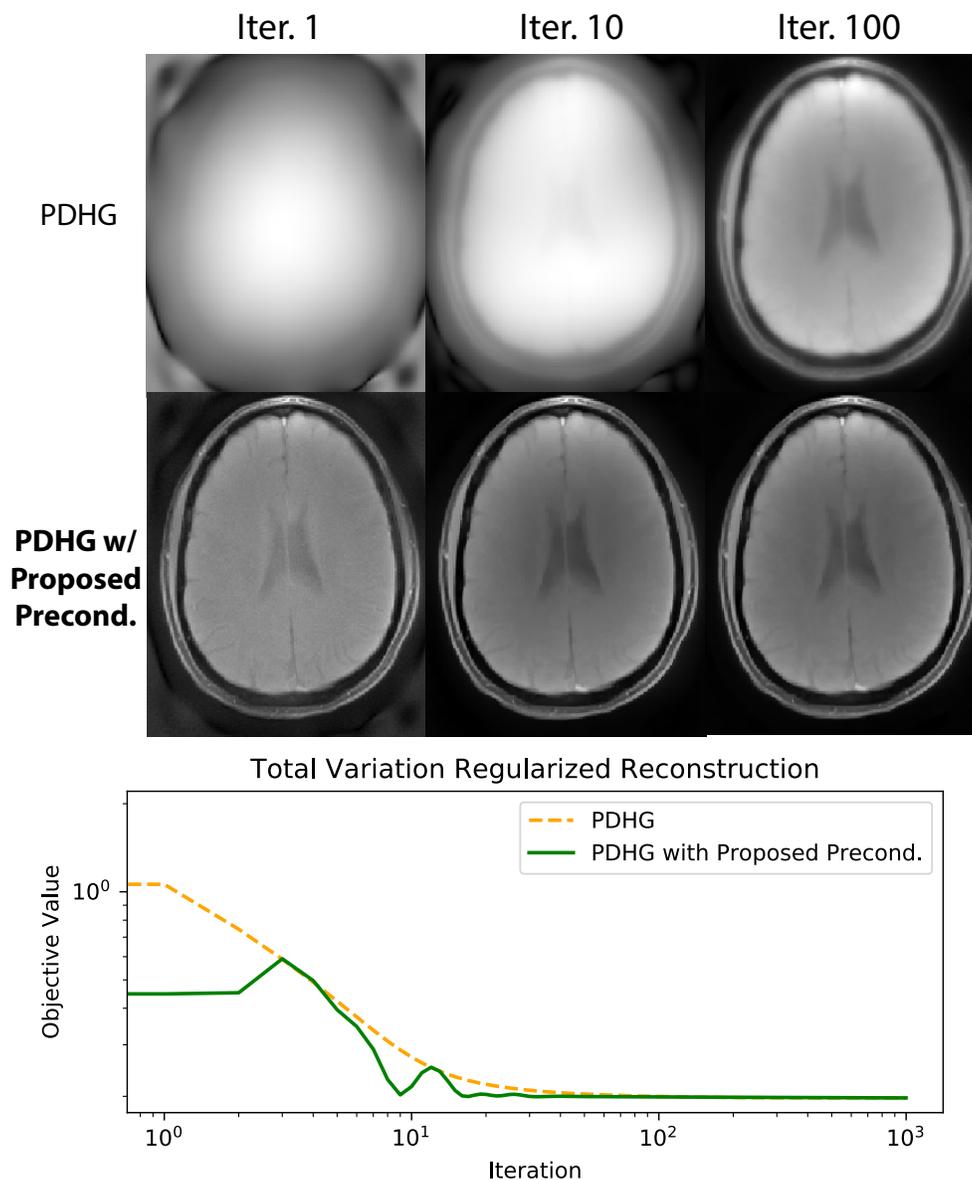


Figure 5.9: Iteration progression for total variation regularized reconstruction of the brain dataset, comparing PDHG with and without the proposed preconditioning. Both methods converged slower compared to the two other reconstruction experiments. Both visually and quantitatively in terms of objective value, the proposed method converges the fastest in about 30 iterations.

the proposed k-space diagonal preconditioning requires only element-wise multiplications.

For ℓ_1 -wavelet regularized reconstruction, the proposed preconditioning consistently accelerates the convergence compared to FISTA and PDHG. The experiments show that the

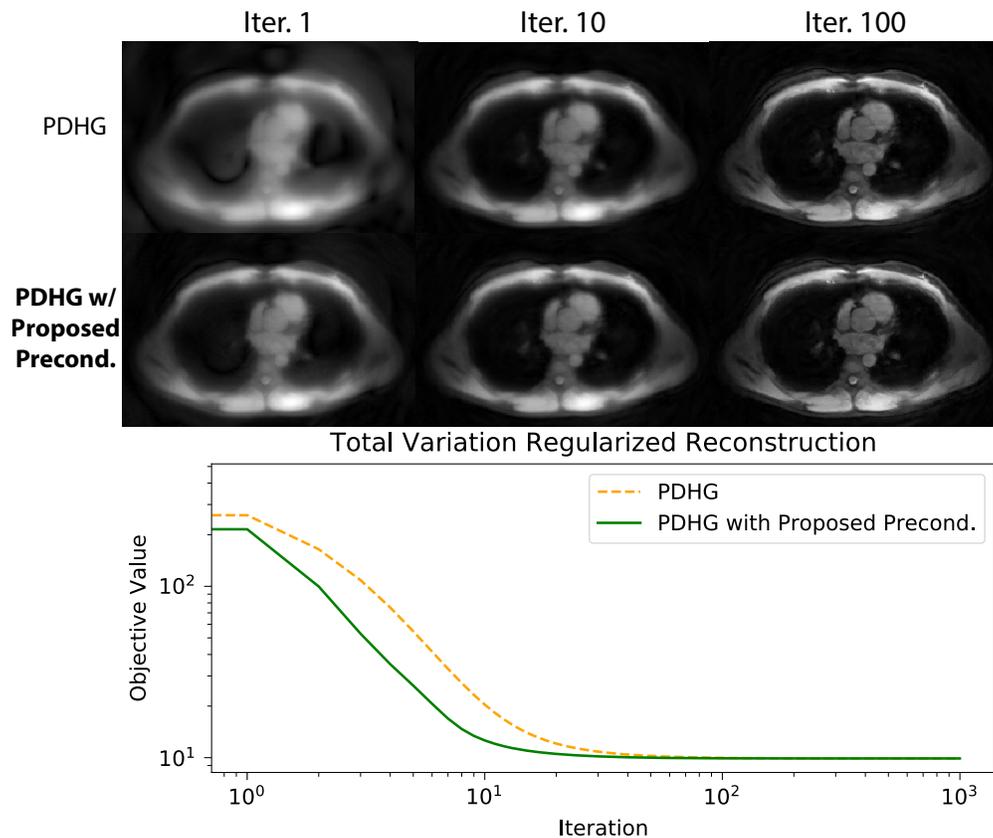


Figure 5.10: Iteration progression for total variation regularized reconstruction of the cardiac dataset, comparing PDHG with and without the proposed preconditioning.

proposed method can reach convergence in about ten iterations, without inner loops. For total variation reconstruction, the proposed preconditioning also accelerates the convergence compared to PDHG, but slower than the ℓ_1 -wavelet regularized reconstruction. This is because the total variation regularization function consists of a non-unitary operator in the non-linear function, and requires introducing auxiliary variables.

Finally, the experiment with the 3D UTE dataset in Figure 5.1 shows that the method offers orders of magnitude speedup for 3D datasets. This is expected because 3D trajectories have a higher variation in k-space density than 2D trajectories. In particular, the proposed method converged in about ten iterations, whereas other methods did not even after a few hundreds of iterations.

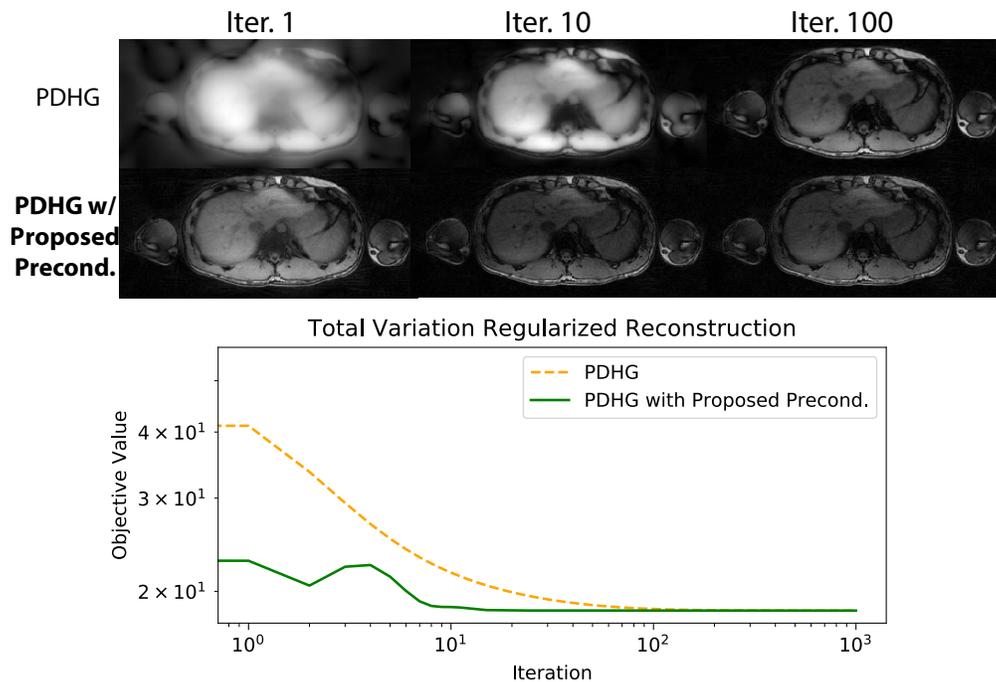


Figure 5.11: Iteration progression for total variation regularized reconstruction of the liver dataset, PDHG with and without the proposed preconditioning.

5.7 Conclusion

I have shown a method to speed up non-Cartesian iterative reconstruction that retains the per-iteration computational efficiency of density compensation and reconstruction accuracy of preconditioning methods. In contrast to most existing preconditioning methods, the proposed technique does not increase the per-iteration computation time much compared to vanilla iterative methods, such as the conjugate gradient method. With the proposed preconditioning, iterative reconstruction can often reach convergence in about ten iterations.

5.8 Derivation for the Dual Problem

Here I will derive the dual problem through augmented Lagrangian. Let me first introduce a variable z to make the objective function (5.3) a constrained optimization problem:

$$\min_{\mathbf{x}, \mathbf{z}} \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2 + g(\mathbf{x})$$

subject to: $\mathbf{z} = \mathbf{Ax}$

Introducing a Lagrangian variable \mathbf{u} gives me,

$$\min_{\mathbf{x}, \mathbf{z}} \max_{\mathbf{u}} \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2 + g(\mathbf{x}) + \langle \mathbf{u}, (\mathbf{z} - \mathbf{A}\mathbf{x}) \rangle$$

Switching the min and the max, gives me the following dual problem:

$$\max_{\mathbf{u}} \min_{\mathbf{x}, \mathbf{z}} \frac{1}{2} \|\mathbf{z} - \mathbf{y}\|_2^2 + g(\mathbf{x}) + \langle \mathbf{u}, (\mathbf{z} - \mathbf{A}\mathbf{x}) \rangle$$

Minimizing over \mathbf{z} gives me $\mathbf{z} = \mathbf{y} - \mathbf{u}$. Substituting, and re-arranging, I obtain:

$$\max_{\mathbf{u}} -\frac{1}{2} \|\mathbf{u}\|_2^2 + \langle \mathbf{u}, \mathbf{y} \rangle + \min_{\mathbf{x}} g(\mathbf{x}) - \langle \mathbf{A}^H \mathbf{u}, \mathbf{x} \rangle$$

Using the definition of a conjugate function $g^*(\mathbf{x}^*) = \max_{\mathbf{x}} \langle \mathbf{x}^*, \mathbf{x} \rangle - g(\mathbf{x})$, I have,

$$\max_{\mathbf{u}} -\frac{1}{2} \|\mathbf{u}\|_2^2 + \mathbf{u}^H \mathbf{y} - g^*(-\mathbf{A}^H \mathbf{u})$$

5.9 Derivation for L2 optimized diagonal preconditioner

I am interested in solving the following minimization problem:

$$\min_{\mathbf{p}} \frac{1}{2} \|\text{diag}(\mathbf{p}) \mathbf{A} \mathbf{A}^H - \mathbf{I}\|_F^2$$

Expanding the objective function element-by-element, I obtain,

$$\min_{\mathbf{p}} \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^M |\mathbf{p}_i \mathbf{a}_i^H \mathbf{a}_j - \delta_{ij}|^2$$

where δ is the Dirac delta function.

Taking the gradient with respect to \mathbf{p}_i and setting it to zero, I obtain,

$$\mathbf{p}_i \sum_{j=1}^M |\mathbf{a}_i^H \mathbf{a}_j|^2 - \|\mathbf{a}_i\|_2^2 = 0$$

Using the fact that $\mathbf{a}_i^H \mathbf{a}_i$ is non-negative and re-arranging, I have,

$$\mathbf{p}_i^{-1} = \frac{\sum_{j=1}^M |\mathbf{a}_i^H \mathbf{a}_j|^2}{\|\mathbf{a}_i\|_2^2}$$

Chapter 6

Learning a Sparse Representation from Many Undersampled Datasets

6.1 Introduction

Compressed sensing [10] is now part of several commercial products [104, 105, 106] for accelerated scans. It has also been integrated into many advanced applications, including 4D flow [59], dynamic contrast enhanced (DCE) imaging [54], and lung imaging [5]. As shown in Chapter 2, there are three components for the application of compressed sensing MRI: 1) the image of interest is compressible using a signal transform. 2) MRI can acquire (pseudo)-randomized samples in k-space, which provides incoherent encoding of the underlying image. 3) sparsity/compressibility can be efficiently enforced in reconstruction to recover the compressed representation from the undersampled measurements.

While compressed sensing has delivered impressive results, it inherently treats each reconstruction as an independent optimization problem. Correlations between scans are not being used. The commonly used sparsifying transforms, such as the wavelet transform and total variation, are designed for general natural images, and not for a specific MR application. On the other hand, in clinics and hospitals, the same imaging protocol is often prescribed numerous times daily. These images exhibit very similar contrasts and statistics. Hence, a natural question is: Given many datasets of the same application, can a reconstruction leverage them to improve upon compressed sensing?

Recent deep learning based methods [107, 108, 109, 110] have shown that with a database of fully-sampled datasets, a learned reconstruction can improve reconstruction quality compared to compressed sensing reconstruction. In particular, in the training phase, these techniques often fit a deep neural network to map retrospectively undersampled datasets to corresponding fully-sampled datasets. Then, with enough training datasets, the network can capture implicitly the features unique to that particular application. Comparing to compressed sensing, learned reconstructions often results in improved image quality.

On the other hand, collecting a large number of fully-sampled datasets becomes im-

mentally difficult for 3D volumes, and impossible for most dynamic applications, such as dynamic contrast enhanced (DCE) MRI. The result is that often machine learning algorithms are trained on: 1) Natural images databases, which do not exploit the types of structures and features unique to MRI applications 2) “Fully sampled” DICOM magnitude images, which are often low-resolution, magnitude only, and may exhibit artifacts. 3) Parallel imaging and compressed sensing reconstruction as the gold standard, which puts a limit on the achievable image quality to be only as good as parallel imaging and compressed sensing.

In this chapter, I consider the problem of using many undersampled datasets to improve reconstruction of accelerated scans. In particular, I propose a machine learning based method for training convolutional sparse coding (CSC) [111] from undersampled data to improve reconstruction quality. CSC is a method that learns filters to sparsely represent images from a training database. Since images for each application exhibit similar contrast and statistics, filters learned on many instances of the same application will be more tailored to that application than generic filters (such as wavelets) do. As described in Chapter 2, compressed sensing reconstruction quality is directly correlated to sparsity in the transform domain. Hence, I can use filters learned with CSC as a sparse representation for CS reconstruction to improve image quality.

A closely related line of work is blind compressed sensing [112, 113], in which both the sparse representation and the sparse representation are optimized during compressed sensing reconstruction. While able to go beyond compressed sensing in certain applications, one limitation of these techniques is that the sparsifying transforms are only learned from single undersampled datasets. This is limiting because image features can be obscured by aliasing artifacts in one dataset, but be prominent in many others. The signal-to-noise of a single dataset also restricts the quality of the learned transform.

The proposed method can be thought of a blind compressed sensing method but applied on all undersampled datasets in a database at the same time. Since the sampling pattern is often different for each dataset, this provides diversity in the aliasing artifacts. Also, similar structures appear in different datasets, which averages noise. Of course, the optimization problem becomes much bigger than conventional blind compressed sensing problems. In Section 6.3, I will present a stochastic algorithm that can train on large-scale undersampled datasets efficiently.

6.2 Problem statement and Objective function

In this section, I will extend the conventional CSC formulation to learn a sparse representation from many undersampled datasets.

Let me begin by looking at the wavelet transform as the sparse representation, which is often used for compressed sensing. In particular, given an image \mathbf{x} , it is represented as,

$$\mathbf{x} = \sum_{j=1}^J \psi_j * \mathbf{c}_j \quad (6.1)$$

where ψ_j represents the j th wavelet filter, and \mathbf{c}_j is represents the sparse coefficients for the j th wavelet filter.

While the wavelet transform is known to sparsely represent natural images, including MRI, one might be able to design filters to produce sparser coefficients when many instances of similar images are given. CSC leverages the convolutional sparse structure of the wavelet transform. But instead of using the fixed wavelet filters, it finds filters ψ_j to sparsify images from a training database.

In particular, given many images $\{\mathbf{x}_i\}_{i=1}^I$ from a database, CSC considers the following problem:

$$\underset{\{\mathbf{c}_{ij}, \psi_j: \|\psi_j\| \leq 1\}_{i,j}}{\text{minimize}} \quad \frac{1}{2} \sum_{i=1}^I \left\| \sum_{j=1}^J \psi_j * \mathbf{c}_{ij} - \mathbf{x}_i \right\|_2^2 + \lambda \sum_{j=1}^J \|\mathbf{c}_{ij}\|_1 \quad (6.2)$$

where \mathbf{c}_{ij} represents the sparse coefficients for the i th image and j th filter and ψ_j represents the j th filter. The ℓ_1 -norm is used to enforce sparsity on the coefficients. To prevent the filters scaling arbitrarily to infinity, the filters ψ_j are enforced to have ℓ_2 norm less than one.

Since both the filters and the coefficients are being optimized, CSC can adapt the filters to the given images to produce sparse coefficients. Note that the CSC objective function is non-convex, so the global minimum may not be attainable efficiently. In practice, alternating minimization is often applied on the objective function to converge to a stationary point.

To extend the CSC formulation to undersampled datasets, I propose to incorporate the multichannel MR acquisition model. In particular, I consider many k-space datasets $\{\mathbf{y}_i \in \mathbb{C}^{M_i}\}_{i=1}^I$ from a database. Each k-space data \mathbf{y}_i may be undersampled and corresponds to an underlying image $\mathbf{x}_i \in \mathbb{C}^N$, which is related by,

$$\mathbf{y}_i = \mathbf{F}_i \mathbf{S} \mathbf{x}_i + \mathbf{w}_i \quad (6.3)$$

where $\mathbf{F}_i \in \mathbb{C}^{M_i C \times NC}$ is a Fourier sampling operator, $\mathbf{S} \in \mathbb{C}^{NC \times N}$ is a sensitivity map operator, and $\mathbf{w}_i \in \mathbb{C}^{M_i C}$ is a vector representing white Gaussian noise. Note that the Fourier sampling operator can be different for each dataset with different sampling patterns.

Convolutional Sparse Model



Figure 6.1: Illustration of the convolutional sparse model for images.

Then, combining with the convolutional sparse model, the overall forward model is given by,

$$\mathbf{y}_i = \mathbf{F}_i \mathbf{S} \left(\sum_{j=1}^J \boldsymbol{\psi}_j * \mathbf{c}_{ij} \right) + \mathbf{w}_i. \quad (6.4)$$

Extending the CSC objective function (6.2), I consider the following objective function to learn sparsifying filters from undersampled datasets:

$$\underset{\{\mathbf{c}_{ij}, \boldsymbol{\psi}_j : \|\boldsymbol{\psi}_j\| \leq 1\}_{i,j}}{\text{minimize}} \quad \frac{1}{2} \sum_{i=1}^I \left\| \mathbf{F}_i \mathbf{S} \left(\sum_{j=1}^J \boldsymbol{\psi}_j * \mathbf{c}_{ij} - \mathbf{y}_i \right) \right\|_2^2 + \lambda \sum_{j=1}^J \|\mathbf{c}_{ij}\|_1 \quad (6.5)$$

6.3 Training using Stochastic Alternating Minimization

The above objective function (6.5) is convex when either $\{\hat{\mathbf{c}}_{ij}\}_{ij}$ or $\{\hat{\boldsymbol{\psi}}_j\}_j$ is fixed. Hence, a natural way of optimizing it is through alternating minimization. However, applying alternating minimization directly on the objective function (6.5) would require immense computation and memory costs. In particular, all datasets and their corresponding coefficients have to be loaded into memory. Fourier transforms must also be performed to all datasets even for a single gradient evaluation.

Instead, similar to existing machine learning training pipeline, I propose to apply stochastic optimization techniques to efficiently optimize the objective function. Concretely, in each iteration, I randomly pick a data index i in $\{1, \dots, I\}$, and perform alternating minimization stochastically as follows:

$$\{\hat{\mathbf{c}}_{ij}\}_{j=1}^J \in \underset{\{\mathbf{c}_{ij}\}_j}{\text{argmin}} \quad \frac{1}{2} \left\| \mathbf{F}_i \mathbf{S} \left(\sum_{j=1}^J \hat{\boldsymbol{\psi}}_j * \mathbf{c}_{ij} \right) - \mathbf{y}_i \right\|_2^2 + \lambda \sum_{j=1}^J \|\mathbf{c}_{ij}\|_1 \quad (6.6)$$

$$\{\hat{\boldsymbol{\psi}}_j\}_{j=1}^J \in \underset{\{\boldsymbol{\psi}_j : \|\boldsymbol{\psi}_j\|_2 \leq 1\}_j}{\text{argmin}} \quad \frac{1}{2} \left\| \mathbf{F}_i \mathbf{S} \left(\sum_{j=1}^J \boldsymbol{\psi}_j * \hat{\mathbf{c}}_{ij} \right) - \mathbf{y}_i \right\|_2^2 + \sum_{j=1}^J \frac{1}{2\alpha} \|\boldsymbol{\psi}_j - \hat{\boldsymbol{\psi}}_j\|_F^2 \quad (6.7)$$

where α is a step-size to control how much to update the filters. This algorithm is similar to that in Chapter 4. Again, the $\hat{\boldsymbol{\psi}}_j$ update step is generally referred to as an implicit update or stochastic proximal update [75]. It is known to converge to an approximate stationary point in expectation, and more robust to the step-size choice α than stochastic gradient methods. Each subproblem is convex and can be approximately solved using the Fast Iterative Soft-Thresholding Algorithm (FISTA) [11].

Finally, I note that a feature of the proposed algorithm is that only coefficients for one dataset $\{\boldsymbol{\alpha}_{ij}\}_j$ needs to be stored in each iteration. This is in contrast to the stochastic gradient method, which needs to store the coefficients for all datasets during training.

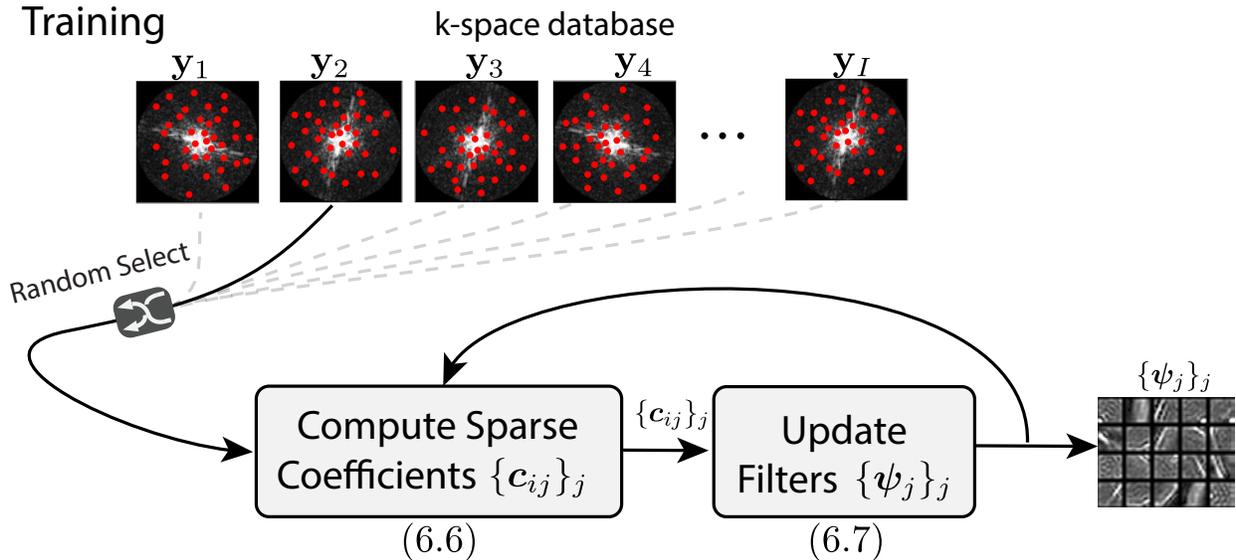


Figure 6.2: Illustration of the training algorithm. In each iteration, a data index i is randomly picked in $\{1, \dots, I\}$, and alternating minimization performed.

6.4 Reconstruction using the learned filters

When reconstructing new scans, the filters are fixed and only the coefficients are computed. The final image is reconstructed by convolving the coefficients with the filters. Concretely, given a new k-space dataset \mathbf{y} , the reconstructed image $\hat{\mathbf{x}}$ is computed as follows:

$$\begin{aligned} \{\hat{\mathbf{c}}_j\}_{j=1}^J &\in \operatorname{argmin}_{\{\mathbf{c}_j\}_j} \frac{1}{2} \left\| \mathbf{FS} \left(\sum_{j=1}^J \hat{\psi}_j * \mathbf{c}_j \right) - \mathbf{y} \right\|_2^2 + \lambda \sum_{j=1}^J \|\mathbf{c}_j\|_1 \\ \hat{\mathbf{x}} &= \sum_{j=1}^J \hat{\psi}_j * \hat{\mathbf{c}}_j \end{aligned} \quad (6.8)$$

The subproblem can again be approximated using FISTA. Hence, the resulting reconstruction has similar computational complexity as conventional compressed sensing reconstructions.

6.5 Methods

Twenty fully sampled-3D knee datasets [114] were downloaded from mridata.org [77] and used for training and evaluation. The k-space datasets were acquired on a 3T GE Discovery MR 750, with a fast spin-echo sequence (CUBE), and an 8-channel HD knee coil. Scan parameters include a matrix size of $320 \times 320 \times 256$, a field of view of $16 \times 16 \times 15.3 \text{ cm}^3$,

Reconstruction

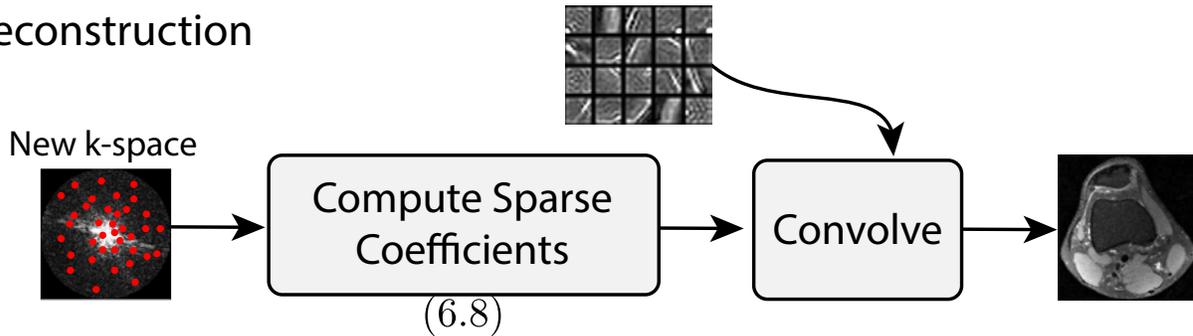


Figure 6.3: Illustration of the reconstruction pipeline. When reconstructing new scans, the filters are fixed and only the coefficients are computed. The final image is reconstructed by convolving the coefficients with the filters.

receiver bandwidth of 50 kHz, and TE/TR of 25ms/1550ms. The proposed method was implemented in Tensorflow, and run on an Nvidia Titan Xp GPU.

To obtain more training datasets, I consider 2D slices extracted from the 3D volumes, instead of using the 3D volumes directly. A total of 5120 slices from 16 cases were used for training, 640 slices from 2 cases were used for validation, and 640 slices from the remaining 2 cases were used for testing. The datasets were normalized with respect to the maximum value of each 3D volume.

The filters $\{\hat{\psi}_j\}_j$ were initialized as unit norm white Gaussian noise, and the coefficients $\{\hat{c}_{ij}\}_{i,j}$ were initialized as zeros. Each sub-problem in the alternating minimization was approximated using FISTA with 100 iterations. The $\{\hat{\psi}_j\}_j$ sub-problem was initialized with the previous iterate solution. The $\{\hat{c}_{ij}\}_{i,j}$ sub-problem was initialized with zeros to be consistent with the actual reconstruction, which would not have a previous iterate solution.

Using the proposed method, I first trained a set of filters from the fully-sampled training datasets, with 256 11×11 filters. I then under-sampled each 3D dataset with 8x Poisson-disk sampling masks with 24×24 calibration region. The sampling masks for each volume were different. Using sensitivity maps estimated with ESPIRiT [101], I trained another set of filters from under-sampled datasets. Regularization parameters were selected to minimize mean squared error in the validation datasets. ℓ_1 -wavelet regularized reconstruction was also implemented, and compared. The regularization parameters were also selected to minimize mean squared error in the validation datasets.

6.6 Results

In the spirit of reproducible research, I provide a software package in Python to reproduce the results described in this chapter. The software package can be downloaded from:

https://github.com/mikgroup/csc_mri.git

Figure 6.4 shows the filters learned from the fully-sampled datasets. Compared to the wavelet filters, these filters display edge-like features with various orientations, instead of just horizontal and vertical. This can provide a more edge aware reconstruction as shown below in Figure 6.6. On the other hand, the filters notably lack smooth filters. Even those that display mostly constant amplitude show impulse-like and noise-like structures. Finally, I note that the phase of the filters is largely constant, with most variations coming from zero-crossings. This is because the datasets are acquired using a fast spin-echo sequence, which removes the image phase.

The filters learned from the under-sampled datasets are shown in Figure 6.5. These filters show similar structures as the ones learned from fullysampled datasets, with edge-like and impulse-like features. One difference is that the filters seem less noisy than the ones learned from fully-sampled datasets. This can also be seen in the phase of the filters.

Filters Trained on Fully-sampled Datasets

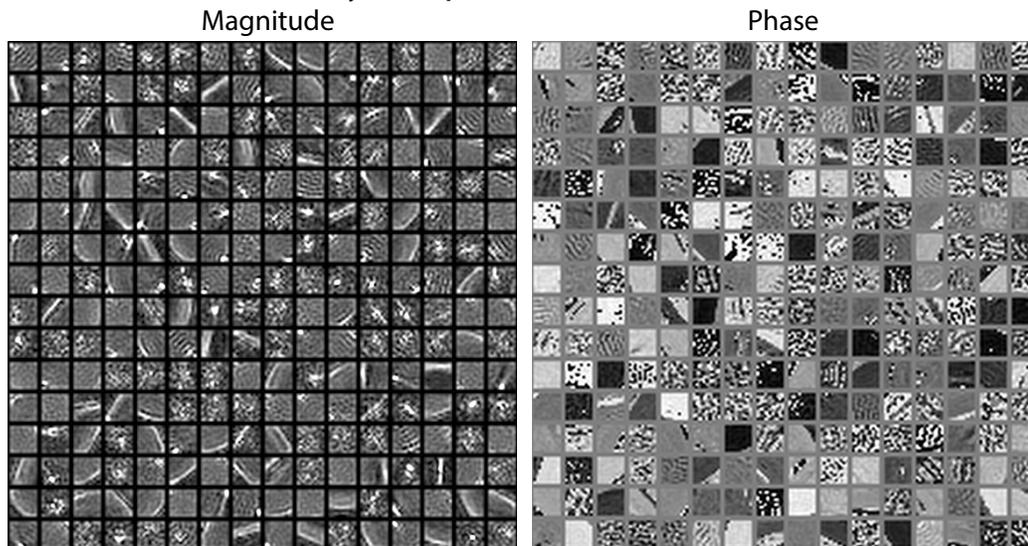


Figure 6.4: Filters learned from fully-sampled datasets. Compared to wavelet filters, these filters display edge-like features with various orientations, instead of just horizontal and vertical. On the other hand, the filters notably lack smooth filters.

Figure 6.6 shows a representative ℓ_1 regularized reconstructions with the wavelet transform, and the learned filters. In general, reconstructions with the learned filters show sharper edges. As mentioned, this may be due to the prevalent edge structures in the filters. However, increased noise-like artifacts are also observed in the reconstructed image with learned filters, compared to that with the wavelet transform. This may be due to the lack of smooth filters in the learned transform. Such behavior is prevalent in the testing datasets.

Figure 6.7 shows a box-plot of PSNR difference between the learned reconstructions versus ℓ_1 wavelet reconstruction over the 640 test slices. Reconstruction with the filters

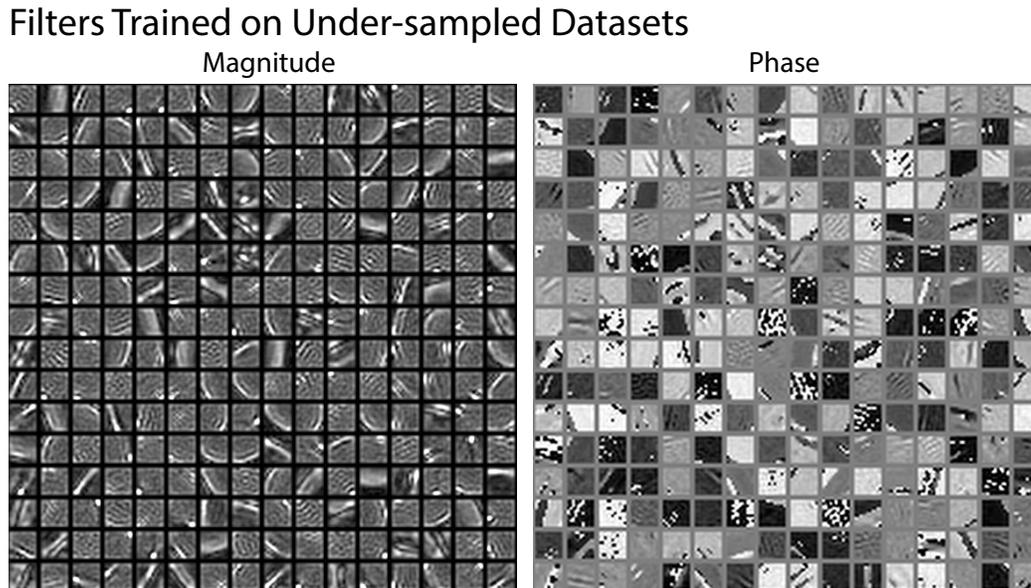


Figure 6.5: Filters learned from under-sampled datasets. These filters show similar structures as the ones learned from fullysampled datasets, with edge-like and impulse-like features. One difference is that the filters seem less noisy than the ones learned from fully-sampled datasets. This can also be seen in the phase of the filters.

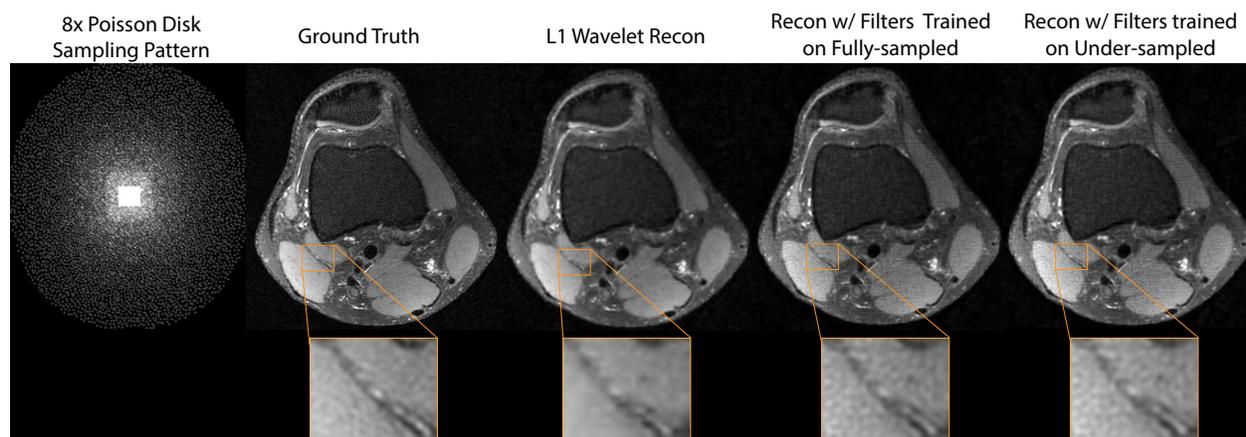


Figure 6.6: A representative ℓ_1 -regularized reconstructions with wavelet transform, and the learned filters. In general, reconstructions with the dictionaries showed sharper edges, but increased noise artifacts as well.

learned from fully-sampled datasets increases the average PSNR by 0.168 dB compared with ℓ_1 -wavelet reconstruction, and reconstruction with the filters learned from under-sampled

datasets increases the average PSNR by 0.322 dB. One surprising observation is that the filters learned from under-sampled datasets result in reconstruction with higher average PSNR. Since only slices from two 3D volumes were used for testing, it is not clear whether this is due to small sample size.

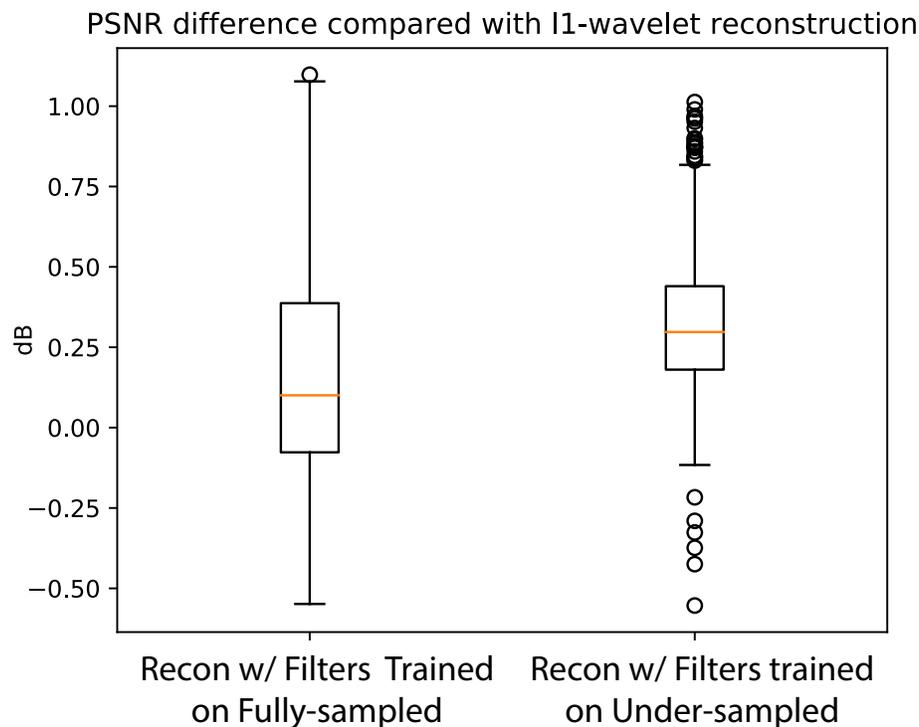


Figure 6.7: A box-plot of PSNR difference between the learned reconstructions versus ℓ_1 -wavelet reconstruction over the 640 test slices.

6.7 Discussion

I have presented a machine learning method based on CSC to learn a sparse representation from undersampled datasets. In the experiments, I have shown that filters learned with CSC result in improved reconstruction quality in terms of PSNR. What is surprising is that an extended CSC formulation incorporating undersampling also improves reconstruction quality. This shows that undersampled datasets can be leveraged for improving reconstruction quality. As shown in Figure 6.6, the resulting images display sharper features compared to ℓ_1 -wavelet reconstruction.

On the other hand, the reconstructions using the learned filters exhibit noise-like artifacts on smooth regions. This may be because the learned filters in Figures 6.4 and 6.5 mostly

contain edges and impulses, rather than smooth patches. Interestingly, the filters learned from undersampled datasets contain less noise and result in slightly higher average PSNR in Figure 6.7. One potential reason for this difference is that, for the undersampled datasets, variable density sampling patterns were used. Hence, the CSC data consistency emphasizes low frequencies when trained on undersampled datasets. It is possible a formulation with emphasis on the low-frequency representation can further learn an improved sparsity representation.

Another potential reason for the increase noise amplification in Figure 6.6 is that a sparsity enforcing formulation may not encourage smooth filters. An observation supporting this is that the wavelet coarse scale produces non-sparse coefficients. Hence, compressed sensing reconstruction often do not impose sparsity on the coarse scale. In this case, other structural constraints might be needed rather than sparsity.

Finally, a main contribution of this chapter is the stochastic alternating minimization algorithm for the objective function (6.5). The algorithm only requires storing one coefficient at each iteration. It also converges to a stationary point in expectation. However, unlike alternating minimization for non-convex low rank factorization [115, 74], alternating minimization on CSC does not have theoretical guarantees on its convergence to global minimum. In my experience, I have found that CSC training is quite sensitive to the hyper-parameters, such the step-size, the number of iterations for the sub-problem and the regularization parameter. For example, too high of a regularization parameter can set most coefficients to zero in early iterations. This causes only a few filters get updated afterwards. A change in the algorithm might mitigate this. It is also possible that the ℓ_1 norm is too “harsh”, and a smoothed version can lead to improved convergence.

6.8 Conclusion

I have shown a method based on CSC to learn a sparse representation from many undersampled datasets. I have presented an efficient algorithm to achieve this. The resulting technique can be useful a wide range of applications, especially for dynamic MRI.

Chapter 7

Summary and Future Work

In this dissertation, I have presented techniques to reconstruct high resolution 3D dynamic MRI. Below are a summary of contributions and future directions.

7.1 Summary of Contributions

Multiscale Low rank Matrix Model

I have presented a multiscale low rank matrix decomposition method that combines both multiscale modeling and low rank matrix decomposition. Using a convex formulation, I can solve for the decomposition efficiently and exactly, provided that the multiscale signal components are incoherent. I demonstrate its effectiveness in four applications, including illumination normalization for face images, motion separation for surveillance videos, multi-scale modeling of DCE-MRI and collaborative filtering exploiting age information.

High Resolution Volumetric Dynamic MRI from Non-Gated Acquisitions

I have presented a method to reconstruct high spatiotemporal resolution 3D dynamic MRI data from continuous non-gated acquisitions. The results showed transient dynamics, such as localized contrast dynamics, that are inherently lost in low frame-rate reconstructions. The proposed method can also resolve detailed pulmonary structures for respiratory motion at variable rates. Finally, the proposed reconstruction is robust to bulk motion even without explicitly modeling it because the artifacts are localized in time.

Accelerating Convergence for Non-Cartesian MRI Reconstruction

I have shown a method to speed up non-Cartesian iterative reconstruction that retains the per-iteration computational efficiency of density compensation and reconstruction accuracy of preconditioning methods. In contrast to most existing preconditioning methods, the proposed technique does not increase the per-iteration computation time much compared

to vanilla iterative methods, such as the conjugate gradient method. With the proposed preconditioning, iterative reconstruction can often reach convergence in about ten iterations.

Learning a Sparse Representation from Many Undersampled Datasets

I have shown a method based on CSC to learn a sparse representation from many undersampled datasets. I have presented an efficient algorithm to achieve this. The resulting technique can be useful a wide range of applications, especially for dynamic MRI.

7.2 Future work

While this dissertation has made progress toward the ideal scan – a 3D dynamic MR scan that can resolve all dynamics at a high spatiotemporal resolution – it has not achieved this goal yet. Below are a few directions I find promising:

Low rank models incorporating motion

Low rank models in dynamic imaging have difficulties in representing large motion over time. This can be seen in Chapter 4, when bulk motions result in severe artifacts. Integrating motion, even only simple translation motion, with low rank models might greatly improve reconstruction quality.

Cloud computing

All reconstructions in this dissertation were performed on shared memory workstations. However, as Moore’s law is approaching its end, cloud computing is becoming an attractive alternative for high performance computing. Besides implementation, several aspects can be explored for the algorithm: asynchronous iterations, privacy preserving reconstructions, and communication avoiding updates.

Beyond diagonal k-space preconditioners

This dissertation explored the use of diagonal k-space preconditioners. An obvious extension is to use non-diagonal ones. How to efficiently compute them and whether these provide additional speedup will be interesting to explore.

Improved convolutional sparse coding

Finally, I believe the formulation for convolutional sparse coding has a lot of room for improvement. Innovations in its formulation and algorithm can potentially make a huge difference in reconstruction quality.

Bibliography

- [1] D. G. Nishimura. *Principles of magnetic resonance imaging*. Stanford Univ., 1996.
- [2] J. Hennig, K. Scheffler, J. Laubenberger, and R. Strecker. “Time-resolved projection angiography after bolus injection of contrast agent”. In: *Magnetic resonance in medicine* 37.3 (1997), pp. 341–345.
- [3] F. R. Korosec, R. Frayne, T. M. Grist, and C. A. Mistretta. “Time-resolved contrast-enhanced 3D MR angiography”. In: *Magnetic Resonance in Medicine* 36.3 (1996), pp. 345–351.
- [4] M. Saranathan, D. W. Rettmann, B. A. Hargreaves, S. E. Clarke, and S. S. Vasanawala. “Differential subsampling with cartesian ordering (DISCO): A high spatio-temporal resolution dixon imaging sequence for multiphasic contrast enhanced abdominal imaging”. In: *Journal of Magnetic Resonance Imaging* 35.6 (2012), pp. 1484–1492.
- [5] W. Jiang, F. Ong, K. M. Johnson, S. K. Nagle, T. A. Hope, M. Lustig, and P. Larson. “Motion robust high resolution 3D free-breathing pulmonary MRI using dynamic 3D image self-navigator”. In: *Magnetic Resonance in Medicine* 79 (), pp. 2954–2967. DOI: 10.1002/mrm.26958.
- [6] L. Feng, L. Axel, H. Chandarana, K. T. Block, D. K. Sodickson, and R. Otazo. “XD-GRASP: golden-angle radial MRI with reconstruction of extra motion-state dimensions using compressed sensing”. In: *Magnetic resonance in medicine* 75.2 (2016), pp. 775–788.
- [7] A. G. Christodoulou, J. L. Shaw, C. Nguyen, Q. Yang, Y. Xie, N. Wang, and D. Li. “Magnetic resonance multitasking for motion-resolved quantitative cardiovascular imaging”. In: *Nature Biomedical Engineering* 2.4 (2018), p. 215.
- [8] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger. “SENSE: sensitivity encoding for fast MRI”. In: *Magnetic resonance in medicine* 42.5 (1999), pp. 952–962.
- [9] K. P. Pruessmann, M. Weiger, P. Bornert, and P. Boesiger. “Advances in sensitivity encoding with arbitrary k-space trajectories”. In: *Magnetic Resonance in Medicine* 46 (2001), pp. 638–651. DOI: 10.1002/mrm.1241.

- [10] M. Lustig, D. Donoho, and J. M. Pauly. “Sparse MRI: The application of compressed sensing for rapid MR imaging”. In: *Magnetic Resonance in Medicine* 58.6 (2007), pp. 1182–1195. DOI: 10.1002/mrm.21391.
- [11] A. Beck and M. Teboulle. “A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems”. In: *SIAM Journal on Imaging Sciences* 2 (2009), pp. 183–202. DOI: 10.1137/080716542.
- [12] S. G. Mallat. “A theory for multiresolution signal decomposition: the wavelet representation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1989). DOI: 10.1515/9781400827268.494.
- [13] E. Candes, L. Demanet, D. Donoho, and L. Ying. “Fast discrete curvelet transforms”. In: *Multiscale Modeling & Simulation* (2006). DOI: 10.1109/iadcc.2014.6779451.
- [14] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. “Shiftable multiscale transforms”. In: *IEEE Transactions on Information Theory* 38.2 (1992), pp. 587–607. DOI: 10.1109/18.119725.
- [15] D. L. Donoho and I. M. Johnstone. “Ideal spatial adaptation by wavelet shrinkage”. In: *Biometrika* 81.3 (2006), pp. 1–31. DOI: 10.2307/2337118.
- [16] D. L. Donoho. “Compressed sensing”. In: *IEEE Transactions on Information Theory* 52.4 (2006), pp. 1289–1306. DOI: 10.1109/TIT.2006.871582.
- [17] E. J. Candes, J. Romberg, and T. Tao. “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information”. In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 489–509. DOI: 10.1109/TIT.2005.862083.
- [18] D. L. Donoho and X. Huo. “Uncertainty principles and ideal atomic decomposition”. In: *IEEE Transactions on Information Theory* 47.7 (2001), pp. 2845–2862. DOI: 10.1109/18.959265.
- [19] J. L. Starck, Y. Moudden, J. Bobin, M. Elad, and D. L. Donoho. “Morphological component analysis”. In: *Optics & Photonics 2005* 5914 (2005), 59140Q–59140Q–15. DOI: 10.1117/12.615237.
- [20] S. S. Chen, D. L. Donoho, and M. A. Saunders. “Atomic decomposition by basis pursuit”. In: *SIAM Journal on Scientific Computing* 20.1 (1998), pp. 33–61. DOI: 10.1137/s1064827596304010.
- [21] Z. P. Liang. “Spatiotemporal imaging with partially separable functions”. In: *The Proceedings of the International Symposium of Biomedical Imaging: From Nano to Macro*. IEEE, 2007, pp. 988–991. ISBN: 1-4244-0671-4. DOI: 10.1109/isbi.2007.357020.
- [22] R. Basri and D. W. Jacobs. “Lambertian reflectance and linear subspaces”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25.2 (2003), pp. 218–233. DOI: 10.1109/TPAMI.2003.1177153.

- [23] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. “Eigentaste: A Constant Time Collaborative Filtering Algorithm”. In: *Information Retrieval* 4.2 (2001), pp. 133–151. DOI: 10.1023/A:1011419012209.
- [24] M. Fazel. “Matrix rank minimization with applications”. PhD thesis. Stanford University, 2002.
- [25] E. J. Candes and B. Recht. “Exact Matrix Completion via Convex Optimization”. In: *Foundations of Computational Mathematics* 9.6 (2009), pp. 717–772. DOI: 10.1007/s10208-009-9045-5.
- [26] M. Fazel, T. K. Pong, D. Sun, and P. Tseng. “Hankel Matrix Rank Minimization with Applications to System Identification and Realization”. In: *SIAM Journal on Matrix Analysis and Applications* 34.3 (2013), pp. 946–977. DOI: 10.1137/110853996.
- [27] E. J. Candes, Y. C. Eldar, T. Strohmer, and V. Voroninski. “Phase Retrieval via Matrix Completion”. In: *SIAM Journal on Imaging Sciences* 6.1 (2013), pp. 199–225. DOI: 10.1137/110848074.
- [28] V. Chandrasekaran, S. Sanghavi, and P. A. Parrilo. “Rank-sparsity incoherence for matrix decomposition”. In: *SIAM Journal on Optimization* 21.2 (2011), pp. 572–596. DOI: 10.1137/090761793.
- [29] A. Agarwal, S. Negahban, and M. J. Wainwright. “Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions”. In: *The Annals of Statistics* 40.2 (2012), pp. 1171–1197.
- [30] R. R. Coifman and D. L. Donoho. “Translation-Invariant De-Noising”. In: *Wavelets and Statistics*. New York, NY: Springer New York, 1995, pp. 125–150. ISBN: 978-0-387-94564-4. DOI: 10.1007/978-1-4612-2544-7_9.
- [31] H. Xu, C. Caramanis, and S. Sanghavi. “Robust PCA via outlier pursuit”. In: *Advances in Neural Information Processing Systems* 23 (2010), pp. 2496–2504.
- [32] E. J. Candes and T. Tao. “The Power of Convex Relaxation: Near-Optimal Matrix Completion”. In: *IEEE Transactions on Information Theory* 56.5 (2010), pp. 2053–2080. DOI: 10.1109/TIT.2010.2044061.
- [33] B. Recht, M. Fazel, and P. A. Parrilo. “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization”. In: *SIAM review* 52.3 (2010), pp. 471–501.
- [34] D. Hsu, S. M. Kakade, and T. Zhang. “Robust Matrix Decomposition With Sparse Corruptions”. In: *IEEE Transactions on Information Theory* 57.11 (2011), pp. 7221–7234. DOI: 10.1109/TIT.2011.2158250.
- [35] E. J. Candes, X. Li, Y. Ma, and J. Wright. “Robust principal component analysis?” In: *Journal of the ACM* 58.3 (2011), pp. 1–37. DOI: 10.1145/1970392.1970395.
- [36] M. B. McCoy and J. A. Tropp. “The achievable performance of convex demixing”. In: *arXiv preprint. arXiv:1309.7478v1* (2013). arXiv: 1309.7478v1 [cs.IT].

- [37] M. B. McCoy and J. A. Tropp. “Sharp recovery bounds for convex demixing, with applications”. In: *Foundations of Computational Mathematics* 14.3 (2014), pp. 1–51. DOI: 10.1007/s10208-014-9191-2.
- [38] M. B. McCoy, V. Cevher, Q. T. Dinh, and A. Asaei. “Convexity in source separation: Models, geometry, and algorithms”. In: *IEEE Signal Processing Magazine* 31.3 (2014), pp. 87–95. DOI: 10.1109/msp.2013.2296605.
- [39] B. R. Bakshi. “Multiscale PCA with application to multivariate statistical process monitoring”. In: *AIChE Journal* 44.7 (1998), pp. 1596–1610. DOI: 10.1002/aic.690440712.
- [40] R. Kondor, N. Teneva, and V. Garg. “Multiresolution Matrix Factorization”. In: *The 31st International Conference on Machine Learning*. 2014, pp. 1620–1628.
- [41] R. Kakarala and P. O. Ogunbona. “Signal analysis using a multiresolution form of the singular value decomposition”. In: *Image Processing* (2001). DOI: 10.1109/83.918566.
- [42] M. Vozalis and K. Margaritis. “Using SVD and demographic data for the enhancement of generalized Collaborative Filtering”. In: *Information Sciences* 177.15 (2007), pp. 3017–3037. DOI: 10.1016/j.ins.2007.02.036.
- [43] J. Wright, A. Ganesh, K. Min, and Y. Ma. “Compressive principal component pursuit”. In: *Information and Inference: A Journal of the IMA* (2013). DOI: 10.1093/imaiai/iat002.
- [44] R. Foygel and L. Mackey. “Corrupted sensing: Novel guarantees for separating structured signals”. In: *IEEE Transaction on Information Theory* 60.2 (2014), pp. 1223–1247. DOI: 10.1109/tit.2013.2293654.
- [45] A. S. Bandeira and R. van Handel. “Sharp nonasymptotic bounds on the norm of random matrices with independent entries”. In: *Ann. Probab.* 44.4 (2016), pp. 2479–2506. DOI: 10.1214/15-AOP1025.
- [46] A. Belloni, V. Chernozhukov, and L. Wang. “Square-root lasso: pivotal recovery of sparse signals via conic programming”. In: *Biometrika* (2011).
- [47] G. A. Watson. “Characterization of the subdifferential of some matrix norms”. In: *Linear Algebra and Its Applications* (1992). DOI: 10.1016/0024-3795(92)90407-2.
- [48] S. Boyd. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Foundations and Trends in Machine Learning* 3.1 (2010), pp. 1–122. DOI: 10.1561/22000000016.
- [49] N. Parikh and S. Boyd. “Proximal algorithms”. In: *Foundations and Trends in optimization* (2013).
- [50] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 2012. ISBN: 0521839408.

- [51] M. A. T. Figueiredo and R. D. Nowak. “An EM algorithm for wavelet-based image restoration”. In: *IEEE Transactions on Image Processing* 12.8 (2003), pp. 906–916. DOI: 10.1109/TIP.2003.814255.
- [52] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. “From few to many: illumination cone models for face recognition under variable lighting and pose”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.6 (2001), pp. 643–660. DOI: 10.1109/34.927464.
- [53] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. “Statistical Modeling of Complex Backgrounds for Foreground Object Detection”. In: *IEEE Transactions on Image Processing* 13.11 (2004), pp. 1459–1472. DOI: 10.1109/TIP.2004.836169.
- [54] T. Zhang, J. Y. Cheng, A. G. Potnick, R. A. Barth, M. T. Alley, M. Uecker, M. Lustig, J. M. Pauly, and S. S. Vasanawala. “Fast pediatric 3D free-breathing abdominal dynamic contrast enhanced MRI with high spatiotemporal resolution.” In: *Journal of Magnetic Resonance Imaging* 41.2 (2015), pp. 460–473. DOI: 10.1002/jmri.24551.
- [55] R. Otazo, E. Candes, and D. K. Sodickson. “Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components.” In: *Magnetic Resonance in Medicine* 73.3 (2015), pp. 1125–1136. DOI: 10.1002/mrm.25240.
- [56] M. Griffin, T. M. Grist, and C. J. Francois. “Dynamic four-dimensional MR angiography of the chest and abdomen”. In: *Magnetic resonance imaging clinics of North America* 17.1 (2009), pp. 77–90.
- [57] K. M. Johnson, S. B. Fain, M. L. Schiebler, and S. Nagle. “Optimized 3D ultra-short echo time pulmonary MRI”. In: *Magnetic resonance in medicine* 70.5 (2013), pp. 1241–1250.
- [58] M. Markl, A. Frydrychowicz, S. Kozerke, M. Hope, and O. Wieben. “4D flow MRI”. In: *Journal of Magnetic Resonance Imaging* 36.5 (2012), pp. 1015–1036.
- [59] J. Y. Cheng, K. Hanneman, T. Zhang, M. T. Alley, P. Lai, J. I. Tamir, M. Uecker, J. M. Pauly, M. Lustig, and S. S. Vasanawala. “Comprehensive motion-compensated highly accelerated 4D flow MRI with ferumoxytol enhancement for pediatric congenital heart disease”. In: *Journal of Magnetic Resonance Imaging* 43.6 (2016), pp. 1355–1368.
- [60] F. Han, Z. Zhou, M. Cao, Y. Yang, K. Sheng, and P. Hu. “Respiratory motion-resolved, self-gated 4D-MRI using rotating cartesian k-space (ROCK)”. In: *Medical physics* 44.4 (2017), pp. 1359–1368.
- [61] J. D. Trzasko, A. Manduca, and E. Borisch. “Local versus global low-rank promotion in dynamic MRI series reconstruction”. In: *Proc Intl Soc Mag Reson Med* 19. 2011. DOI: 10.5296/jsss.v1i2.5769.s851.

- [62] H. M. Hudson and R. S. Larkin. “Accelerated image reconstruction using ordered subsets of projection data”. In: *IEEE transactions on medical imaging* 13.4 (1994), pp. 601–609.
- [63] H. Erdogan and J. A. Fessler. “Ordered subsets algorithms for transmission tomography”. In: *Physics in Medicine & Biology* 44.11 (1999), p. 2835.
- [64] R. Gordon, R. Bender, and G. T. Herman. “Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography”. In: *Journal of theoretical Biology* 29.3 (1970), pp. 471–481.
- [65] M. Mardani, G. B. Giannakis, and K. Ugurbil. “Tracking tensor subspaces with informative random sampling for real-time mr imaging”. In: *arXiv preprint arXiv:1609.04104* (2016).
- [66] P. T. Gurney, B. A. Hargreaves, and D. G. Nishimura. “Design and analysis of a practical 3D cones trajectory”. In: *Magnetic Resonance in Medicine* 55.3 (2006), pp. 575–582.
- [67] H. Pedersen, S. Kozerke, S. Ringgaard, K. Nehrke, and W. Y. Kim. “k-t PCA: temporally constrained k-t BLAST reconstruction using principal component analysis”. In: *Magnetic resonance in medicine* 62.3 (2009), pp. 706–716.
- [68] B. Zhao, J. P. Haldar, C. Brinegar, and Z.-P. Liang. “Low rank matrix recovery for real-time cardiac MRI”. In: *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*. IEEE. 2010, pp. 996–999.
- [69] S. G. Lingala, Y. Hu, E. DiBella, and M. Jacob. “Accelerated dynamic MRI exploiting sparsity and low-rank structure: kt SLR”. In: *IEEE transactions on medical imaging* 30.5 (2011), pp. 1042–1054.
- [70] M. Chiew, S. M. Smith, P. J. Koopmans, N. N. Graedel, T. Blumensath, and K. L. Miller. “k-t FASTER: acceleration of functional MRI data acquisition using low rank constraints”. In: *Magnetic resonance in medicine* 74.2 (2015), pp. 353–364.
- [71] S. Goud, Y. Hu, and M. Jacob. “Real-time cardiac MRI using low-rank and sparsity penalties”. In: *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*. IEEE. 2010, pp. 988–991.
- [72] S. Burer and R. D. Monteiro. “A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization”. In: *Mathematical Programming* 95.2 (2003), pp. 329–357.
- [73] P. Jain, P. Netrapalli, and S. Sanghavi. “Low-rank matrix completion using alternating minimization”. In: *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*. ACM. 2013, pp. 665–674.
- [74] M. Hardt. “Understanding alternating minimization for matrix completion”. In: *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*. IEEE. 2014, pp. 651–660.

- [75] E. K. Ryu and S. Boyd. “Stochastic proximal iteration: a non-asymptotic improvement upon stochastic gradient descent”. In: *Author website: <http://www.math.ucla.edu/~eryu/papers/spi.pdf>* (2014).
- [76] H. Robbins and S. Monro. “A stochastic approximation method”. In: *Herbert Robbins Selected Papers*. Springer, 1985, pp. 102–109.
- [77] F. Ong, S. Amin, and M. Lustig. “mridata.org: An Open Archive for Sharing MRI Raw Data”. In: *Proc Intl Soc Mag Reson Med 26*. 2018.
- [78] L. Wissmann, C. Santelli, W. P. Segars, and S. Kozerke. “MRXCAT: Realistic numerical phantoms for cardiovascular magnetic resonance”. In: *Journal of Cardiovascular Magnetic Resonance* 16.1 (2014), p. 63.
- [79] C. Ahn, J. Kim, and Z. Cho. “High-speed spiral-scan echo planar NMR imaging-I”. In: *IEEE transactions on medical imaging* 5.1 (1986), pp. 2–7.
- [80] C. H. Meyer, B. S. Hu, D. G. Nishimura, and A. Macovski. “Fast spiral coronary artery imaging”. In: *Magnetic resonance in medicine* 28.2 (1992), pp. 202–213.
- [81] D. C. Alsop, J. A. Detre, X. Golay, M. Gunther, J. Hendrikse, L. Hernandez-Garcia, H. Lu, B. J. MacIntosh, L. M. Parkes, M. Smits, et al. “Recommended implementation of arterial spin-labeled perfusion MRI for clinical applications: A consensus of the ISMRM perfusion study group and the European consortium for ASL in dementia”. In: *Magnetic resonance in medicine* 73.1 (2015), pp. 102–116.
- [82] P. Lauterbur. “Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance”. In: *Nature* 242.5394 (1973), p. 190.
- [83] G. H. Glover and J. M. Pauly. “Projection reconstruction techniques for reduction of motion effects in MRI”. In: *Magnetic resonance in medicine* 28.2 (1992), pp. 275–289.
- [84] C.-M. Tsai and D. G. Nishimura. “Reduced aliasing artifacts using variable-density k-space sampling trajectories”. In: *Magnetic Resonance in Medicine* 43 (2000), pp. 452–458.
- [85] J. Jackson, C. Meyer, D. Nishimura, and A. Macovski. “Selection of a convolution function for Fourier inversion using gridding (computerised tomography application)”. In: *IEEE Transactions on Medical Imaging* 10 (1991), pp. 473–478. DOI: 10.1109/42.97598.
- [86] R. D. Hoge, R. K. Kwan, and B. G. Pike. “Density compensation functions for spiral MRI”. In: *Magnetic Resonance in Medicine* 38 (1997), pp. 117–128. DOI: 10.1002/mrm.1910380117.
- [87] J. G. Pipe and P. Menon. “Sampling density compensation in MRI: Rationale and an iterative numerical solution”. In: *Magnetic Resonance in Medicine* 41 (1999), pp. 179–186.

- [88] H. Moriguchi and J. L. Duerk. “Modified block uniform resampling (BURS) algorithm using truncated singular value decomposition: fast accurate gridding with noise and artifact reduction”. In: *Magnetic resonance in medicine* 46.6 (2001), pp. 1189–1201.
- [89] A. Chambolle and T. Pock. “A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging”. In: *Journal of Mathematical Imaging and Vision* 40 (2011), pp. 120–145. DOI: 10.1007/s10851-010-0251-1.
- [90] B. P. Sutton, D. C. Noll, and J. A. Fessler. “Fast, Iterative Image Reconstruction for MRI in the Presence of Field Inhomogeneities”. In: *IEEE Transactions on Medical Imaging* 22 (2003), pp. 178–188. DOI: 10.1109/TMI.2002.808360.
- [91] S. Ramani and J. A. Fessler. “Parallel MR Image Reconstruction Using Augmented Lagrangian Methods”. In: *IEEE Transactions on Medical Imaging* 30 (2011), pp. 694–706. DOI: 10.1109/TMI.2010.2093536.
- [92] A. E. Yagle. “New Fast Preconditioners for Toeplitz-Like Linear Systems”. In: *IEEE* 2 (2002). DOI: 10.1109/ICASSP.2002.5744057.
- [93] D. S. Weller, S. Ramani, and J. A. Fessler. “Augmented Lagrangian with Variable Splitting for Faster Non-Cartesian L1-SPIRiT MR Image Reconstruction”. In: *IEEE Transactions on Medical Imaging* 33 (2014), pp. 351–361. DOI: 10.1109/tmi.2013.2285046.
- [94] M. Lustig and J. M. Pauly. “SPIRiT: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space”. In: *Magnetic Resonance in Medicine* 64 (2010), pp. 457–471. DOI: 10.1002/mrm.22428.
- [95] T. F. Chan. “An optimal circulant preconditioner for Toeplitz systems”. In: *SIAM journal on scientific and statistical computing* 9 (1988), pp. 766–771.
- [96] M. J. Muckley, D. C, and J. A. Fessler. “Fast, Iterative Subsampled Spiral Reconstruction via Circulant Majorizers”. In: *Proc Intl Soc Mag Reson Med* 26. 2018.
- [97] K. Koolstra, J. v. Gemert, P. Bornert, A. Webb, and R. Remis. “Accelerating compressed sensing in parallel imaging reconstructions using an efficient circulant preconditioner for cartesian trajectories”. In: *Magnetic Resonance in Medicine* (2018). DOI: 10.1002/mrm.27371.
- [98] J. D. Trzasko, A. Manduca, Y. Shu, J. H. III, and M. A. Bernstein. “A Preconditioned ADMM Strategy for Field-Corrected Non-Cartesian MRI Reconstruction”. In: *Proc Intl Soc Mag Reson Med* 22. 2014.
- [99] N. Komodakis and J.-C. Pesquet. “Playing with Duality: An overview of recent primal-dual approaches for solving large-scale optimization problems”. In: *IEEE Signal Processing Magazine* 32 (2015), pp. 31–54. DOI: 10.1109/MSP.2014.2377273.
- [100] T. Pock and A. Chambolle. “Diagonal preconditioning for first order primal-dual algorithms in convex optimization”. In: *2011 International Conference on Computer Vision*. 2011, pp. 1762–1769. DOI: 10.1109/ICCV.2011.6126441.

- [101] M. Uecker, P. Lai, M. J. Murphy, P. Virtue, M. Elad, J. M. Pauly, S. S. Vasanawala, and M. Lustig. “ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA”. In: *Magnetic Resonance in Medicine* 71 (), pp. 990–1001. DOI: 10.1002/mrm.24751.
- [102] S. v. d. Walt, S. C. Colbert, and G. Varoquaux. “The NumPy array: a structure for efficient numerical computation”. In: *Computing in Science & Engineering* 13.2 (2011), pp. 22–30.
- [103] R. Okuta, Y. Unno, D. Nishino, S. Hido, and C. Loomis. “CuPy: A NumPy-Compatible Library for NVIDIA GPU Calculations”. In: *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Thirty-first Annual Conference on Neural Information Processing Systems (NIPS)*. 2017.
- [104] K. King. “HyperSense Enables Shorter Scan Times Without Compromising Image Quality”. In: *GE SIGNA Pulse of MR* 22 (2016), pp. 48–51.
- [105] L. Geerts-Ossevoort, E. d. Weerdt, A. Duijndam, G. v. IJperen, H. Peeters, M. Doneva, M. Nijenhuis, and A. Huang. “Compressed SENSE: Speed done right. Every time.” In: *Philips FieldStrength Magazine* (2018), p. 6619.
- [106] C. Forman, J. Wetzl, C. Hayes, and S. M. “Compressed Sensing: a Paradigm Shift in MRI”. In: *MAGNETOM Flash* (2016), p. 6619.
- [107] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll. “Learning a variational network for reconstruction of accelerated MRI data”. In: *Magnetic resonance in medicine* 79.6 (2018), pp. 3055–3071.
- [108] S. Wang, Z. Su, L. Ying, X. Peng, S. Zhu, F. Liang, D. Feng, and D. Liang. “Accelerating magnetic resonance imaging via deep learning”. In: *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*. IEEE. 2016, pp. 514–517.
- [109] J. Sun, H. Li, and Z. Xu. “Deep ADMM-Net for compressive sensing MRI”. In: *Advances in Neural Information Processing Systems*. 2016, pp. 10–18.
- [110] D. Lee, J. Yoo, and J. C. Ye. “Deep artifact learning for compressed sensing and parallel MRI”. In: *arXiv preprint arXiv:1703.01120* (2017).
- [111] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. “Deconvolutional networks”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2010, pp. 2528–2535. DOI: 10.1109/CVPR.2010.5539957.
- [112] S. Ravishankar and Y. Bresler. “Efficient blind compressed sensing using sparsifying transforms with convergence guarantees and application to magnetic resonance imaging”. In: *SIAM Journal on Imaging Sciences* 8.4 (2015), pp. 2519–2557.
- [113] S. G. Lingala and M. Jacob. “Blind compressed sensing with sparse dictionaries for accelerated dynamic MRI”. In: *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*. IEEE. 2013, pp. 5–8.

- [114] K. Epperson, A. M. Sawyer, M. Lustig, M. Alley, M. Uecker, P. Virtue, P. Lai, and S. Vasanawala. “Creation of fully sampled MR data repository for compressed sensing of the knee”. In: *Section for Magnetic Resonance Technologists*. 2013.
- [115] R. Ge, J. D. Lee, and T. Ma. “Matrix completion has no spurious local minimum”. In: *Advances in Neural Information Processing Systems*. 2016, pp. 2973–2981.