

STUDY OF PARALLEL MR IMAGING TECHNIQUES

by

Wan Kim

May 15, 2015

A thesis submitted to the
Faculty of the Graduate School of
the University at Buffalo, State University of New York
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Electrical Engineering

UMI Number: 1594739

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 1594739

Published by ProQuest LLC (2015). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code



ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

Abstract

In MRI, it is more desirable to scan less data as possible because it reduces MRI scanning time. We want to get a clear image by reconstructing the signals we acquire from the MRI machine. Special scanning or sampling techniques are needed to overcome this issue based on various mathematical methods.

We present an improved random sampling pattern for SAKE (simultaneous auto-calibrating and k-space estimation) reconstruction and an iterative GRAPPA reconstruction using Wiener filter.

In our iterative method using Wiener filter, in contrast to the conventional GRAPPA where only the auto calibration signals (ACS) are used to find the convolution weights, our proposed method iteratively updates the convolution weights using both the acquired and reconstructed data from previous iterations in the entire k-space. To avoid error propagation, the method applies adaptive Wiener filter on the reconstructed data. Experimental results demonstrate that even with a smaller number of ACS lines the proposed method improves the SNR when compared to GRAPPA.

In compressed sensing MRI, it is very important to design sampling pattern for random sampling. For example, SAKE (simultaneous auto-calibrating and k-space estimation) is a parallel MRI reconstruction method using random undersampling. It formulates image reconstruction as a structured low-rank matrix completion problem. Variable density (VD) Poisson discs are typically adopted for 2D random sampling. The basic concept of Poisson disc generation is to guarantee samples are neither too close to nor too far away from each other. However, it is difficult to meet such a condition especially in the high density region. Therefore the sampling becomes inefficient. In this paper, we present an improved random sampling pattern for SAKE reconstruction. The pattern is generated based on a conflict cost with a probability model. The conflict cost measures how many dense samples already assigned are around a target location, while the probability model adopts the generalized Gaussian distribution which includes uniform and Gaussian-like distributions as special cases. Our method preferentially assigns a sample to a k-space location with the least conflict cost

on the circle of the highest probability. To evaluate the effectiveness of the proposed random pattern, we compare the performance of SAKEs using both VD Poisson discs and the proposed pattern. Experimental results for brain data show that the proposed pattern yields lower normalized mean square error (NMSE) than VD Poisson discs.

Acknowledgements

I would like to thank my advisor, Professor Leslie Ying. It was a nice chance to be in her lab, the Computational Biomedical Imaging Laboratory to study about MRI.

I would like to thank the committee members of my thesis, professor Nicholas Mastronarde, Weifeng Su even though they are so busy but make time for me.

I also would like to thank lab members of the Computational Biomedical Imaging Laboratory, Jingyuan Lyu, Yihang Zhou, Ukash Nakarmi and Ren He. Especially thank to Yihang who helps me to understand the basics of my works.

I always thank to my family who is always with me far from here and supports me mentally and financially.

Contents

Abstract	ii
Acknowledgements	iv
Contents	v
List of Figures	vii
1 Introduction	1
1.1 Motivation	1
1.2 Challenge	1
1.3 Outline of Thesis	2
2 Background	
2.1 Magnetic Resonance Imaging (MRI).	3
2.2 Parallel MRI	7
2.2.1 Generalized autocalibrating partially parallel acquisition (GRAPPA).	8
2.2.2 Nonlinear GRAPPA	9
2.3 Compressed Sensing (CS).	10
2.3.1 CS MRI	11
3 Iterative GRAPPA with Wiener filter	12
3.1 Introduction	12
3.2 Conventional GRAPPA	13

3.3 Proposed method	16
3.4 Results	19
3.5 Conclusion	29
4 Conflict-Cost Based Random Sampling Design for Parallel MRI with Low Rank Constraints	30
4.1 Introduction	30
4.2 Method.	31
4.3 Results.	36
4.4 Conclusion.	57
5 Conclusions and Future works	58
5.1 Thesis Summary	58
5.2 Recommendations for Future work	59
6 References	60

List of figures

2.1 patient in z-gradient field	4
2.2 k-space	5
2.3 GRAPPA reconstruction using the correlations along coils and blocks	8
3.1 Improved GRAPPA using Multicolumn multiline interpolation	14
3.2 Block diagram of the proposed iterative GRAPPA	16
3.3 NMSE Comparison of the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) according to various net reduction factors for brain_NL	21
3.4 NMSE Comparison of the MCMLI-FNF (linear) and MCMLI-FNF without hamming window (linearNW) according to various net reduction factors for brain_NL	22
3.5 NMSE Comparison of the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) according to various net reduction factors for brain_SAKE.	22
3.6 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) using ACS lines of 24 at R=3 (netR=2.49) for brain_NL.	23
3.7 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) using ACS lines of 24 at R=3 (netR=2.41) for brain_SAKE	24

3.8 NMSE Comparison of 6 GRAPPAs according to various net reduction factors for brain_NL	26
3.9 NMSE Comparison of 6 GRAPPAs according to various net reduction factors for brain_SAKE.	26
3.10 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative NL GRAPPA (iterNL) using ACS lines of 48 at R=3 (netR=2.15) for brain_NL	27
3.11 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative NL GRAPPA (iterNL) using ACS lines of 48 at R=3 (netR=2.02) for brain_NL	28
4.1 Flow chart of the reconstruction process	32
4.2 Input-output relation of μ -law	39
4.3 VD Poisson mask generated by μ -law from a Poisson disk at R=3.0	39
4.4 Shin and μ -law masks according to various reduction factor R	40
4.5 Proposed mask using different shape parameter at R=3	40
4.6 Magnification of center parts of various masks at R=3	41
4.7 mask generation using generalized Gaussian probability random sampling without conflict cost using different reduction factors	42
4.8 Mask generation using generalized Gaussian probability random sampling with conflict cost and without rounding compensation using different reduction factors	42
4.9 Proposed mask generation using different reduction factors	43

4.10 NMSE of SAKE reconstruction method using proposed random mask of R=3.0 according to the shape parameter a	45
4.11 Reference image and its CCF along kx-axis for brain-SAKE data.	46
4.12 Error image of proposed mask and its CCF along kx-axis for brain-SAKE data	46
4.13 Error image of VD mask and its CCF along kx-axis for brain-SAKE data . . .	47
4.14 Comparison of CCFs of two masks for brain-SAKE data	47
4.15 Effect of conflict cost	48
4.16 Comparison of SAKE reconstruction	49
4.17 For the first dataset at R=3	50
4.18 For the second dataset at R=3	51
4.19 Effect of nonlinear kernel according to the reduction factor R	53
4.20 Performance comparison of SAKE reconstruction method using Shin masks and one using μ -law mask, one using proposed masks, and nonlinear SAKE on using proposed masks according to the reduction factor R	54
4.21 Reference image, reconstructed, magnified reconstruction, and error images by various SAKEs at R=3	55
4.22 Reference image, reconstructed, magnified reconstruction, and error images by various SAKEs at R=3	56

Chapter 1

Introduction

1.1 Motivation

It takes a long time to get good quality for images of patients' inside bodies by the technologies such as X-rays and CTs. The tools for investigating the body of a patient should be fast in order to be safe and to get more money at the same processing time and benefit of less hardware cost. Hence, the speed is important but the image quality should be good enough for doctors to distinguish abnormal parts of the body like cancers from normal parts of the body. As a result, magnetic resonance imaging (MRI) came out.

There are a lot of efforts to make MR image acquisition faster in good quality. Compressed sensing proposed by D. Donoho is based on that one can use signal parts only that have meaningful information and we can abandon the remaining parts which are most of the signal to reconstruct the signal [1]. Parallel Imaging comes from the idea that we can reconstruct images faster by getting the part of images simultaneously using multiple coils [2]. There are various methods that can improve image quality further. Some of them however yield problems such as aliasing and artifacts. Therefore, many people try to solve those problems.

1.2 Challenge

Fast imaging is a challenge because people who investigate MRI try to reduce time by decreasing the acquisition time. Most of the time is for MRI scanning. At the same time, we want good image quality, which may be measured by signal noise ratio (SNR). However, if we reduce data too much, then there will be aliasing and noise so that it is so hard to figure

out what the image is. On the contrary, if we reduce too little then it will cause a lot of computation time. We thus need to deal with trade-off of the acquisition time and the image reduction. We use numerous mathematical tools to find the solution for this. Reducing aliasing and noise provides good image quality enough for the doctors. At the same time, the image reduction causes less time to get the images by removing what makes patients more uncomfortable being inside of an MRI machine. Efficient sampling design and improved algorithm using image processing techniques will be expected to reduce the aliasing and noise.

In the following experiments, we always use full data of brain images for reference data. We assume that we are doing MRI scanning by sampling parts of the full data. It means that we did not perform actual MRI scanning. We usually call it ‘in vivo data’.

1.3 Outline of Thesis

In this chapter, we have mentioned the motivation and challenge of the thesis. Chapter 2 introduces the background of MRI related to the proposed methods. We will talk about the recent MRI techniques including Parallel MRI and Compressed Sensing MRI. As their examples, we will also explain GRAPPA for Parallel MRI and Sparse MRI for Compressed MRI. Chapter 3 and Chapter 4 explain in detail our proposed methods we developed during the research of thesis. We will talk about their introduction, method, result, and conclusion in order. Chapter 3 explains the improved method of GRAPPA. Chapter 4 talks about the improved method of new random sampling for the MRI. Finally, in Chapter 5, we will conclude our whole work and suggest our future work.

Chapter 2

Background

2.1 Magnetic Resonance Imaging (MRI)

Medical imaging has been developed since Röntgen invented X-ray to see bones and living body in 1895. In 1970s, computerized tomography (CT) was invented and then nuclear magnetic resonance (NMR) in 1973. MRI is the same technique with NMR but has a different name because the term ‘nuclear’ sounds dangerous. MRI has many advantages including good contrast to distinguish tumors from normal organs. Also, MRI is safer to human body than previous methods because MRI uses magnetic fields while X-ray uses ionizing radiation. Besides, MRI can make 3D images and has better resolution than CT.

The MRI uses electromagnetic waves in radio frequencies (RF) and external magnetic fields and makes images by the computer. Most parts of human body consist of water which has Hydrogen nuclei or protons. When a proton is in the external magnetic fields, the proton starts to rotate and goes to a higher energy state at Larmor frequency. Larmor frequency is as follows [3]:

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

where the constant γ indicates the unique property of nuclei spinning and B_0 external magnetic field. For example, the precession frequency of a hydrogen nucleus is 42.6MHz/Tesla. This Larmor precession is important because after RF is turned off, the protons release excess energy, and we can get the signals from this spinning

When external magnetic fields B_0 are exposed on a human body, then spinning protons of the body are lined. Then if a RF pulse is sent to a patient body, a signal is received from the spins, which is the MRI signal. MRI gets the signals on 2D frequency domain called k-space. If we use constant magnetic field along the spatial directions, we cannot get the spatial

information. For this reason, for the spatial encoding to verify the spatial information, we change the magnetic field by using gradients along each direction.

$$B = B_0 + G_x x \quad (2.2)$$

We use three directions of gradient fields, G_x , G_y , G_z each for frequency encoding, phase-encoding, and slice selection. If we want to choose some part of the body, then we need to select a slice. The equation of slice selection is as follows,

$$\omega_{rf} = \gamma(B_0 + G_z \Delta z) \quad (2.3)$$

where Δz denotes slice thickness. Figure 2.1 shows how the location of the z gradient coils is related to magnetic field strength.

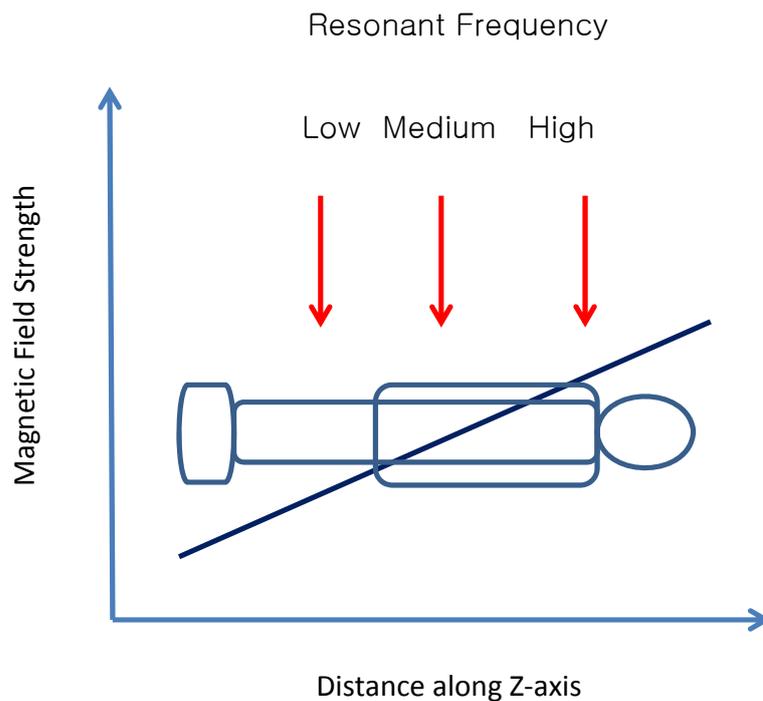


Figure 2.1 patient in z-gradient field

We can decide this parameter by changing the bandwidth of RF frequency ω_{rf} or slope of magnetic field and gradient. Ljunggren and Twieg invented the k-space method to reconstruct an image using Fourier transform of spin density in 1983.

$$S(t) = \int \rho(x) e^{-i\gamma(B_0 + G_x x)t} dx \quad (2.4)$$

where $\rho(x)$ denotes proton density and $e^{-i\gamma B_0 t}$ simple modulation factor. We can remove the factor by demodulation [4]. By adding more gradient G_y , we can represent 2D data space.

$$S(t) = \iint \rho(x, y) e^{-i\gamma(G_x x t + G_y y T)} dx dy \quad (2.5)$$

$$k_x = \gamma G_x t, \quad k_y = \gamma G_y T$$

We can consider the data space represented by k_x and k_y as new spatial frequency domain and the inventors called it k-space. The symbols k_x and k_y are called frequency encoding and phase encoding respectively according to the name of gradients. We can see that each of k_x and k_y depends on time and so the process is done sequentially. This is why we call MR imaging or MRI scanning. Figure 2.2 shows the k-space domain consisting of phase and frequency directions and also shows how the MRI scans sequentially.

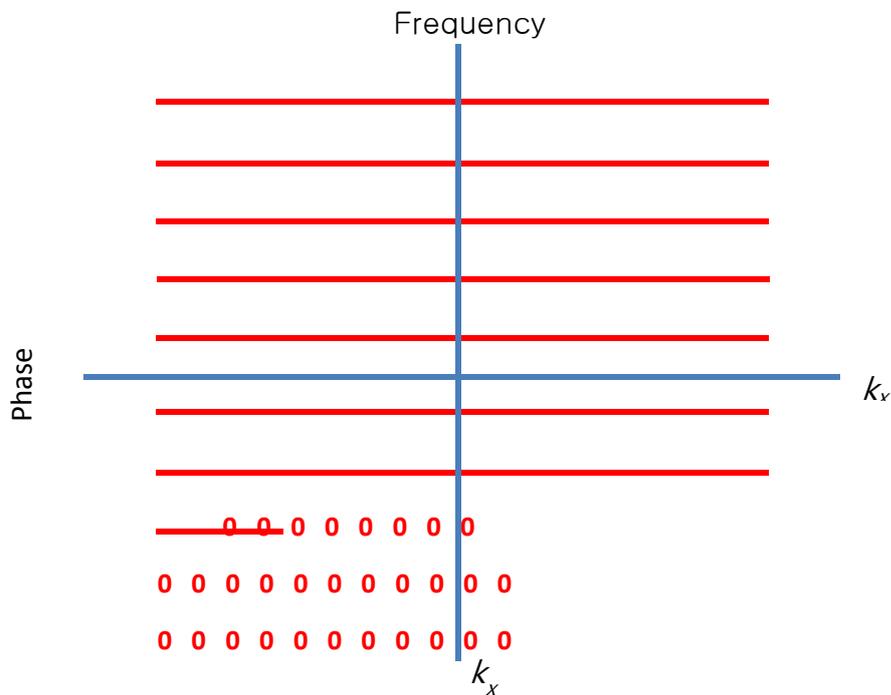


Figure 2.2 k-space

We can reconstruct the image by performing inverse Fourier transform on the k-space data. RF frequencies including 90, 180 degree pulse and three gradients are called pulse sequences. These sequences are key ingredients to make k-space trajectories.

The field of view (FOV) is expressed as the size of an image obtained from MRI scanning. We can calculate FOV as follows [3]:

$$FOV = \frac{BW}{\gamma G} \quad (2.6)$$

where γ is constant, G is the slope of the gradient, and $BW = 2f_{max}$ is bandwidth. Also, we can calculate FOV in image space from the k-space as follows:

$$FOV \propto \frac{1}{\Delta k} \quad (2.7)$$

As Δk increases, which means that the k-space is undersampled, FOV decreases by image processing and this causes the image wrapping or aliasing artifacts.

Recently, parallel imaging techniques such as SENSE[5] and GRAPPA[6] have been introduced to get images faster by using multiple RF coils. Also, compressed sensing MRI has been introduced as a different approach by using compressed sensing theorem proposed by D. Donoho based on the property of sparse sampling.

In MRI, the speed of image acquisition is so important that reduced image data is acquired by scanning of a part of the whole image data. We call the factor reducing the image data or FOV, acceleration or reduction factor. This factor is defined by the ratio of the size of image to the size of sampled pixels in k-space domain. Later, this factor will be frequently used for both parallel MR imaging and compressed sensing MR imaging.

To measure the error between an original image and its reconstruction image, we always use NMSE (Normalized Mean Square Error) as follows [7].

$$NMSE = \frac{\sum_{\vec{r}} (I_{recon}(\vec{r}) - I_{ref}(\vec{r}))^2}{\sum_{\vec{r}} (I_{ref}(\vec{r}))^2} \quad (2.8)$$

where I_{ref} and I_{recon} denote an original reference image and its reconstruction image, respectively.

2.2 Parallel MRI

In clinical MR imaging, the image quality and imaging speed are important [8]. Existing MRI scanners have technical problems to switch the magnetic field gradients to increase the speed of MR Image acquisition. Recently, parallel MRI methods came out to get the sensitivities using multiple coils simultaneously without changing the gradients. We call it parallel imaging because it is done simultaneously.

We get the coil sensitivities used for reconstruction. Each coil location has an effect on the sensitivities. FOV is reduced by reduction factor. The reduction causes aliasing in images. Then we recover the original FOV by reconstruction. The values of reduction factors commonly used are 2 or 3. Commercially and clinically, two famous methods are widely used: SENSE and GRAPPA. The former is an image based method and the latter k-space based one.

Future aim of parallel imaging is getting better image quality by using increase of reduction factor, better calibration methods, and reconstruction algorithms. Also, hardware and MRI scanners will be developed based on these new techniques or theories.

2.2.1 Generalized autocalibrating partially parallel acquisition (GRAPPA) [6]

This is one of parallel imaging methods done in k-space or frequency domain. While SENSE uses estimation of coil sensitivities, but GRAPPA does not so [9]. This method skips some lines of the k-space so there are missing lines for a faster speed. To reconstruct or estimate the missing lines accurately, it uses auto calibration signal (ACS) lines, which means that they were used for calibration process automatically using the image itself. The ACS lines are located in a center part of the k-space. ACS lines are acquired in frequency encoding direction [9] and acquired reducing the number of phase encoding lines so that we take a part of the phase encoding lines and this reduces FOV and causes aliasing [8]. The values of these lines are an important factor to have an effect on image quality.

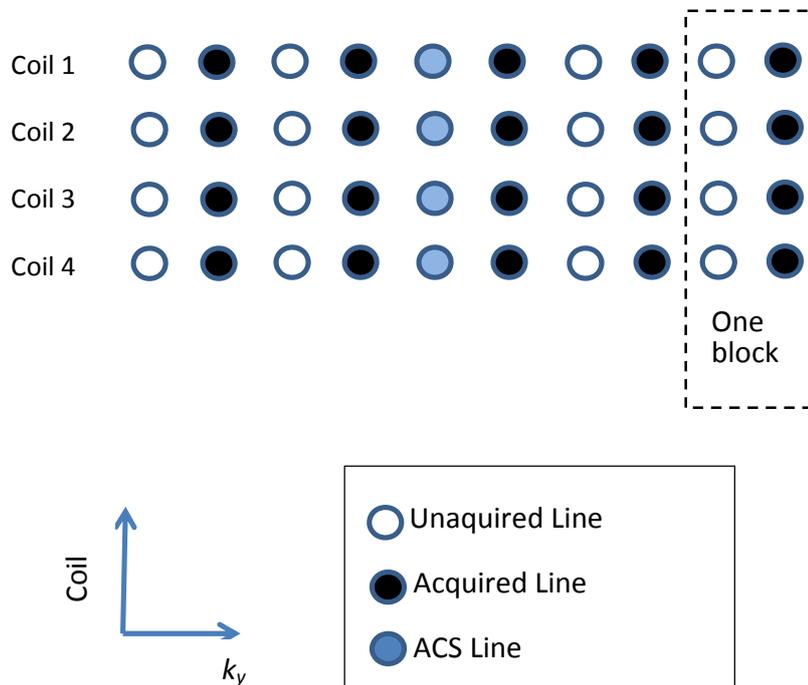


Figure 2.3 GRAPPA reconstruction using the correlations along coils and blocks [6]

Figure 2.3 shows how the acquired line, unacquired line and ACS lines are related to each other in GRAPPA reconstruction. We calculate the weights using the correlation between coils by fitting the acquired lines into ACS lines. Using these weights, we can reconstruct the missing lines. This seems like interpolation process because we interpolate the missing lines so sometimes we called the correlations between acquired lines and ACS lines, interpolation net [10]. After the missing lines are reconstructed, the uncombined coils were combined using sum of squares [11]. As we get more ACS lines acquired, we obtain better image quality. There are some studies about getting more correlations to increase the data to get more accurate weights. The original GRAPPA method uses the correlations along the coils and blocks. Block is a part of the whole k-space data, which have an acquired line and neighbor missing lines. It is for exploiting the correlations along the k_y direction. Then the improved GRAPPA method adopts even columns, which includes neighbors along the k_x direction [10]. It is for using the correlations along the k_x direction. Especially GRAPPA has net reduction factor because actual reduction factor will be thought including combination of ACS lines and acquired lines.

We can use various MRI images such as cardiac, lung, and abdominal images. Other GRAPPAs usually test on brain images or phantom images. There are various GRAPPAs but famous ones are using multicolumn multiline interpolation [10], regularization [12], iterative method [13], and nonlinear kernel [14]. They give solutions to the problems by statistical methods such as least square methods or goodness-of-fit. Also, recently there are some GRAPPAs using temporal correlations such as TGRAPPA [15] and k-tGRAPPA [16].

2.2.2 Nonlinear GRAPPA [14]

This method uses fine parameter tuning to get better reconstruction in GRAPPA. In the paper of nonlinear GRAPPA, to improve medical image quality further, the author uses polynomial kernel method called ‘kernel trick’ before the data reconstruction. Kernel trick is widely used in machine learning technique. The author also says that this method is known to reduce noise effectively in brain data. We map the original data \mathbf{x} into high dimensional

feature space F using nonlinear mapping $\phi(\cdot): x \rightarrow F$. So this is why the author called the method ‘nonlinear’ because we solve the problem in nonlinear space instead of linear space. We use the second-order polynomial kernel function as follows.

$$k(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle = (\gamma x_1^T x_2 + c)^d \quad \text{where } d = 2 \quad (2.9)$$

$$\begin{aligned} \phi(x) = [x_n^2, \dots, x_1^2, \sqrt{2}x_n x_{n-1}, \dots, \sqrt{2}x_n x_1, \sqrt{2}x_{n-1} x_{n-2}, \dots, \\ \sqrt{2}x_{n-1} x_1, \dots, \sqrt{2}x_2 x_1, \sqrt{2}c x_n, \dots, \sqrt{2}c x_1, c] \end{aligned} \quad (2.10)$$

For simplifying the complexity of the computation, we only use linear terms $\sqrt{2}c x_n, \dots, \sqrt{2}c x_1$ and part of nonlinear terms x_n^2, \dots, x_1^2 and remove other terms. As a result,

$$\phi(x) = [x_n^2, \dots, x_1^2, \sqrt{2}c x_n, \dots, \sqrt{2}c x_1] \quad (2.11)$$

2.3 Compressed Sensing [1]

Compressed sensing proposed by D. Donoho is based on that we can use signal parts only that have meaningful information and most of the signals can be thrown away. He assumes that the signal is sparse in certain domain, for example, wavelet or Fourier domain. We can consider the underdetermined system ($m \gg n$) as follows:

$$y = \Phi x \quad (2.12)$$

where $y \in \mathbb{R}^m$ denotes linear measurement or sample, $x \in \mathbb{R}^n$ recovery, and Φ encoding matrix [17]. E. Candes shows that we can recover the original signal even from much smaller samples than ones based on the Nyquist-sampling theorem [18] by solving the following convex optimization problem in l_1 norm [19].

$$\min \|\psi x\|_1 \quad \text{s. t. } \Phi x = y. \quad (2.13)$$

MRI is one of famous applications of compressed sensing.

2.3.1 CS MRI [2]

MRI needs to reduce acquired data but requires good image quality without aliasing and artifacts. This property is well fit to the theory of the compressed sensing because the MR images are sparse in k-space domain, in other words, Fourier domain so that the compressed sensing MRI is introduced. We can set the constrained optimization problem from the compressed sensing as follows

$$\min \|\psi x\|_1 \tag{2.14}$$

$$s. t. \|F_u x - y\|_2 < \varepsilon_1$$

where F_u means undersampled Fourier transform, y acquired MRI k-space data, and ψ the linear operator which makes data sparse. Numerous experiments were done but still not used clinically for hardware constraints. Hence, CS MRI is limitedly used in 3D reconstruction or 3D cardiac images (2D space and 1D time) in kt-space domain by using temporal correlation. Well known methods using temporal correlation are ktFocuss [20] and ktISD [21].

Chapter 3

Iterative GRAPPA

with Wiener Filter

3.1 Introduction

Partially parallel imaging (PPI) is designed for MR imaging to reduce the information and increase the imaging speed. GRAPPA is one of PPI modalities and uses relatively small amount of auto calibration signals (ACSs) in whole k-space. It utilizes the correlations between the coils to estimate the missing lines [6]. Since GRAPPA has been introduced, several iterative GRAPPA approaches have been proposed to estimate the missing lines more accurately [13,22,23]. They use arbitrary k-space sampling by using iteratively conjugate gradient method[22], later developed into SPIRiT (Iterative Self-consistent Parallel Imaging Reconstruction from Arbitrary k-space)[24], or iteratively estimate more accurate convolution weights using regularization [13] or Kalman filtering on weights [23].

In this chapter, we present a new iterative method using Wiener filter. Conventional GRAPPA methods use the auto calibration signals (ACS) to find the convolution weights because the convolution weights of the ACS is shift-invariant, which means that the GRAPPA assumes the weights can be used everywhere in k-space regardless of the locations [25].

We however note that the convolution weights obtained from ACS cannot represent the relationship between the acquired and missing data accurately because the information in ACS is quite different from the information in signals far from ACS. To address this issue, we propose a method that iteratively updates the convolution weights using both the acquired

and reconstructed data from previous iterations in the entire k-space. Among various conventional GRAPPAs, we adopt an improved GRAPPA using multicolumn multiline interpolation (MCMLI) with floating net-based fitting [10] for GRAPPA reconstruction in each iteration. In order to avoid error propagation due to iterations, the method applies adaptive Wiener filter on the reconstructed data where the power spectra are estimated from the neighboring data adaptively without any a priori information. Our experiments are performed on the brain data. We compared the proposed method with the conventional GRAPPA. The method shows improvement in SNR over the GRAPPA.

3.2 Conventional GRAPPA

Multicolumn multiline interpolation [10] chosen for one-step Grappa reconstruction uses the convolutional weights and the acquired data to reconstruct the missing lines as follows:

$$\hat{s}(k_y R + r, k_x; j) = \sum_{l=1}^L \sum_{b=-B_1}^{B_2} \sum_{h=-H_1}^{H_2} w(l, b, h; j, r) s((k_y + b)R, k_x + h; l)$$

for $r = 1, 2, \dots, (R - 1)$ (3.1)

where the symbol r is the order of missing lines under acceleration factor R and the indices b and h count blocks and columns, respectively. The symbols $s(k_y R, k_x; j)$ and $\hat{s}(k_y R + r, k_x; j)$ denote the acquired signal of j th coil and the estimate of the r th unacquired signal or missing line. The estimation of a missing line is obtained by the convolution (3.1) for $r = 1, \dots, (R - 1)$. It is assumed in (3.1) that the missing lines can be reconstructed by a weighted summation of neighbor acquired lines along blocks, columns, and coils.

In order to express (3.1) in a vector-matrix representation for easy calculation, we now reorder the elements $\hat{s}(k_y R + r, k_x; j)$ and $w(l, b, h; j, r)$ as the following. Fixing j and r and reordering the two-dimensional elements $\hat{s}(k_y R + r, k_x; j)$ of two indices k_y and k_x into one-dimensional elements of a new index n , we can express them as $\hat{s}(n)_{j,r}$. Similarly, fixing j

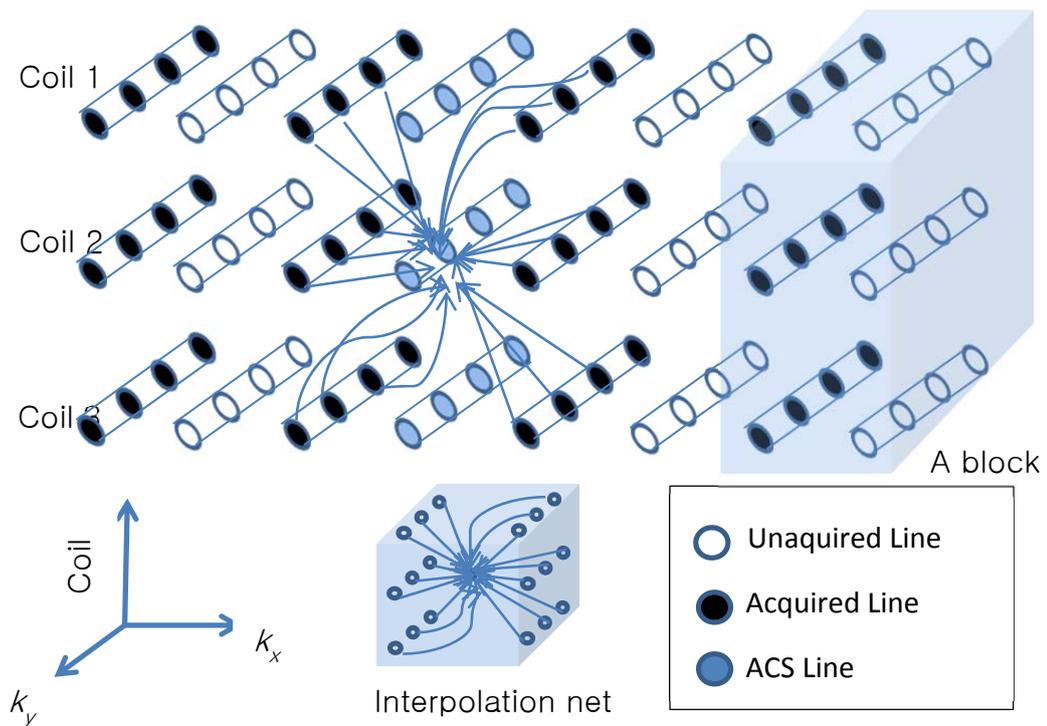
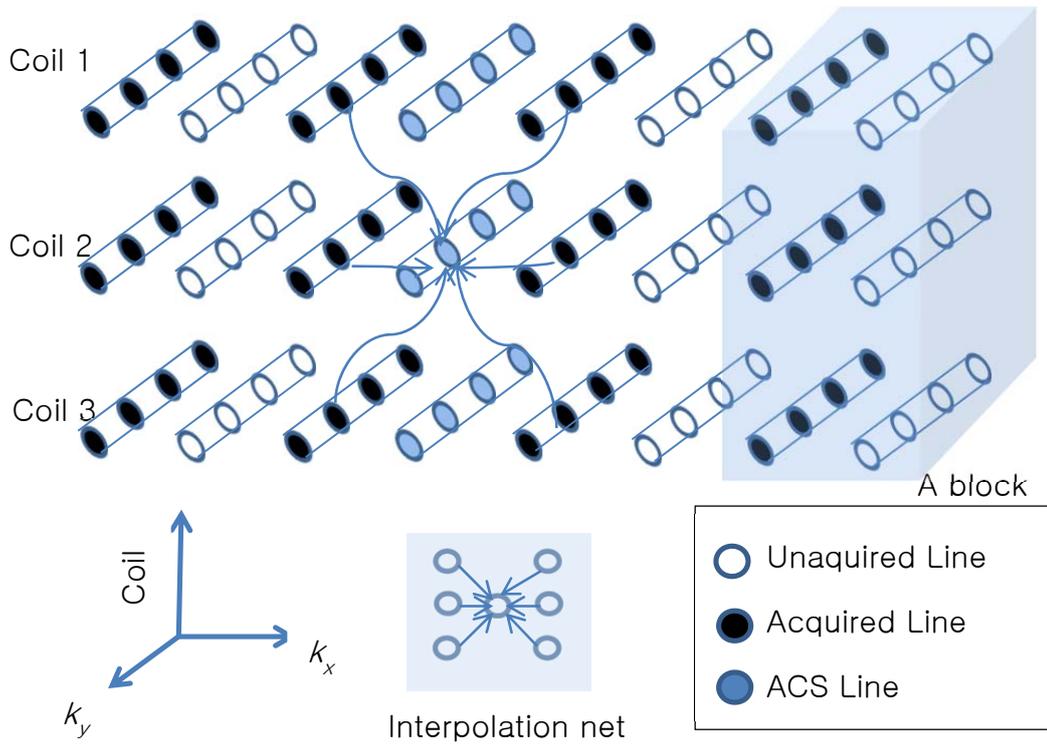


Figure 3.1 improved GRAPPA using Multicolumn multiline interpolation [10]

and r and reordering the three-dimensional elements $w(l, b, h; j, r)$ of three indices $l, b,$ and h into one-dimensional elements of a new index m , we can express them as $w(m)_{j,r}$. Then the elements $s((k_y + b)R, k_x + h; l)$ of three indices $l, b,$ and h and two indices k_y and k_x can be represented as $s(n; m)$. As a result, the equation (3.1) can be rewritten by

$$\hat{s}(n)_{j,r} = \sum_m s(n; m) w(m)_{j,r} \quad (3.2)$$

Or in a vector-matrix representation, we get

$$\hat{\mathbf{s}}_{j,r} = \mathbf{S}\mathbf{w}_{j,r} \quad \text{for } j = 1, \dots, N_l \text{ and } r = 1, \dots, (R - 1) \quad (3.3)$$

where $\hat{\mathbf{s}}_{j,r}$ and $\mathbf{w}_{j,r}$ are a vector of size $N_{yd}N_x \times 1$ and a vector of size $N_lN_bN_c \times 1$, respectively, and \mathbf{S} a matrix of size $N_{yd}N_x \times N_lN_bN_c$. The symbol N_{yd} denotes the number of undersampled k_y lines, N_x the number of k_x lines, N_l the number of coils, N_b is the number of blocks, and N_c the number of columns.

Minimizing the least-square error of MCMLI as in least square fitting, we can obtain the optimal weights $\mathbf{w}_{j,r}$. The least-square error of MCMLI is written as

$$\begin{aligned} J &= \|\hat{\mathbf{s}}_{j,r} - \mathbf{s}_{j,r}\|^2 \\ &= (\mathbf{S}\mathbf{w}_{j,r} - \mathbf{s}_{j,r})^T (\mathbf{S}\mathbf{w}_{j,r} - \mathbf{s}_{j,r}) \end{aligned} \quad (3.4)$$

To minimize the cost function J , we get [26]

$$\frac{\partial J}{\partial \mathbf{w}_{j,r}} = 2\mathbf{S}^T(\mathbf{S}\mathbf{w}_{j,r} - \mathbf{s}_{j,r}) = 0 \quad (3.5)$$

Or

$$\mathbf{w}_{j,r} = (\mathbf{S}^T\mathbf{S})^{-1}\mathbf{S}^T\mathbf{s}_{j,r} \quad (3.6)$$

where T denotes the conjugate transpose. Since we do not know the values of the r th missing lines, we substitute \mathbf{S} and $\mathbf{s}_{j,r}$ with ACS data of \mathbf{S} and $\mathbf{s}_{j,r}$ to get the weight matrix $\mathbf{w}_{j,r}$:

$$\mathbf{w}_{j,r} = (\mathbf{S}^{ACS T} \mathbf{S}^{ACS})^{-1} \mathbf{S}^{ACS T} \mathbf{s}_{j,r}^{ACS} \quad (3.7)$$

We call (3.6) or (3.7) the LS fitting for obtaining weights.

FNF is a scheme that floats the net-based fitting to increase the amount of data needed to obtain LS fitting as in (3.6) or (3.7) [10]. That is, they consider a new fitting from $s((k_y + b)R + d, k_x + h; l)$ to $\hat{s}(k_y R + r + d, k_x; j)$. Then we can obtain a new matrix \mathbf{S}_d and vector $\mathbf{s}_{j,r+d}$ corresponding to \mathbf{S} and $\mathbf{s}_{j,r}$. We see that if $d = 0$, then \mathbf{S}_d and $\mathbf{s}_{j,r+d}$ become \mathbf{S} and $\mathbf{s}_{j,r}$, respectively.

Averaging the weights obtained over $d = 0, \dots, R - 1$, we get a new weight computation as

$$\mathbf{w}_{j,r} = \frac{1}{R} \sum_{d=0}^{R-1} (\mathbf{S}_d^T \mathbf{S}_d)^{-1} \mathbf{S}_d^T \mathbf{s}_{j,r+d} \quad (3.8)$$

In case of using ACS data only, we get

$$\mathbf{w}_{j,r} = \frac{1}{R} \sum_{d=0}^{R-1} (\mathbf{S}_d^{ACS T} \mathbf{S}_d^{ACS})^{-1} \mathbf{S}_d^{ACS T} \mathbf{s}_{j,r+d}^{ACS} \quad (3.9)$$

3.3 Proposed Method

3.3.1 Overall System

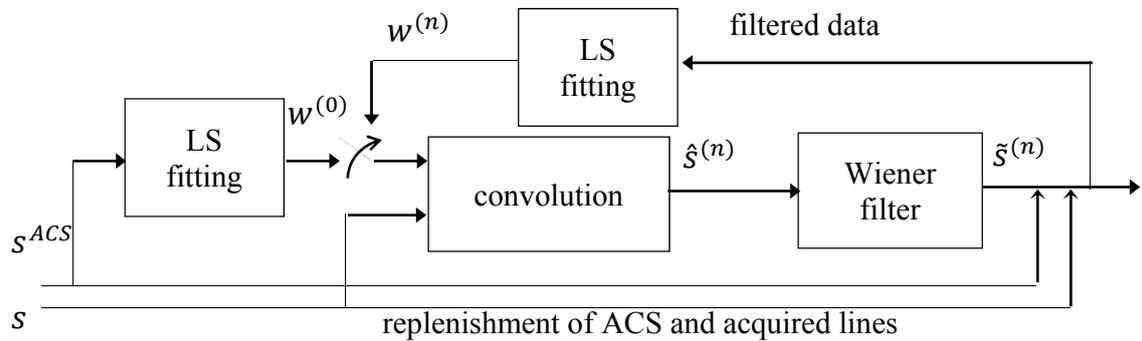


Figure 3.2 Block diagram of the proposed iterative GRAPPA

Figure 3.2 shows the block diagram of the proposed iterative GRAPPA. As mentioned in the previous section, we use MCMLI-FNF [10] for one-step GRAPPA.

(Step 0) At the iteration index $n=0$, we get the initial weights $w^{(0)}$ from ACS data based on (3.9) and set the initial filtered signal $\tilde{s}^{(0)}(k_y, k_x; j) = 0$. We then iterate the following five steps.

(Step 1) We update the iteration index $n \rightarrow n + 1$ and use the $(n-1)$ th weights $w^{(n-1)}$ and the acquired data to reconstruct the missing lines as follows:

$$\hat{s}^{(n)}(k_y R + r, k_x; j) = \sum_{l=1}^L \sum_{b=-B_1}^{B_2} \sum_{h=-H_1}^{H_2} w^{(n-1)}(l, b, h; j, r) s((k_y + b)R, k_x + h; l) \quad (3.10)$$

where R represents a reduction factor and the indices b and h count blocks and columns, respectively. The symbols $s(k_y R, k_x; j)$ denotes the acquired signal of j th coil and $\hat{s}^{(n)}(k_y R + r, k_x; j)$ the estimate of r th missing lines of j th coil at the n th iterations. The convolution by (3.10) is obtained for $r = 1, \dots, (R - 1)$. Since there is no estimation for $r=0$, it is filled with the part of the previous filtered signal $\tilde{s}^{(n-1)}(k_y R, k_x; j)$.

(Step 2) Apply an adaptive Wiener filter to all the reconstructed lines $\hat{s}^{(n)}$ obtained in Step 1 to reduce the aliasing noise [27] using

$$\tilde{s}^{(n)}(k_y, k_x; j) = g^{(n)}(k_y, k_x; j) \hat{s}^{(n)}(k_y, k_x; j), \quad (3.11)$$

where $g^{(n)}(k_y, k_x; j)$ represents the Wiener filter gain computed for the location $(k_y, k_x; j)$ at the n th iteration. How to get the gain will be described in the next section.

(Step 3) The portion of acquired data and ACS in $\tilde{s}^{(n)}$ are replenished by ACS and acquired data s because acquired data and ACS are original and always better than filtered data.

(Step 4) Find the convolutional weights $w^{(n)}$ by using (3.8) to fit the filtered data $\tilde{s}^{(n)}$ into acquired data s .

(Step 5) Repeat Step 1~4 until a stop condition holds.

3.2.2 Adaptive Wiener filter

The Wiener filter we use is a sort of denoising SNR filter. Using the Wiener filtering by locally estimating the signal power spectrum adaptively [27], we can effectively remove the noise instead of using a simple Wiener filter. For this purpose, the Wiener filter gain computed at the n th iteration is given as

$$g^{(n)}(k_y, k_x; j) = \frac{P_s^{(n)}(k_y, k_x; j)}{P_s^{(n)}(k_y, k_x; j) + \sigma_n^{2(n)}} \quad (3.12)$$

where $P_s(k_y, k_x; j)$ denotes the signal power spectrum. We estimate the signal power spectrum by soft thresholding the average $|\hat{s}^{(n)}(k_y, k_x; j)|^2$ in a neighborhood:

$$P_s^{(n)}(k_y, k_x; j) = \begin{cases} \frac{1}{N_s} \sum_{B(k_y, k_x)} |\hat{s}^{(n)}(k_y, k_x; j)|^2 - \sigma_n^{2(n)}, & \text{if } P_s^{(n)}(k_y, k_x; j) > \sigma_n^{2(n)} \\ 0, & \text{otherwise} \end{cases} \quad (3.13)$$

where $B(k_y, k_x)$ is the neighborhood of (k_y, k_x) and N_s is the size of $B(k_y, k_x)$ on k -space. If N_s is too small, then the power spectrum will be noisy. On the contrary, if N_s is too large, then it does not reflect local characteristics. We assume that the noise is stationary and white so that the noise power spectrum is

$$P_n^{(n)}(k_y, k_x) = \sigma_n^2 \quad (3.14)$$

where $\sigma_n^{2(n)}$ is the noise variance of the n th iteration. The noise variance is estimated from ACS lines and their reconstructed ones in a very high frequency (VHF) zone as follows:

$$\sigma_n^2 = \frac{\beta}{N_W} \sum_{(k_y, k_x; j) \in \text{VHF Zone}} \left| S^{\text{ACS}}(k_y, k_x; j) - \hat{S}^{(n)\text{ACS}}(k_y, k_x; j) \right|^2. \quad (3.15)$$

We estimate the noise variance in a VHF zone because that in the center part of the ACS lines is often severely high so that it is not matched to that in non ACS lines. The noise

variance is estimated by the mean squared errors between the reconstructed and acquired data in the ACS region. Since the VHF zone contains only high frequency components along k_x axis but belongs to low frequency region along k_y axis, the noise variance may be over-estimated yet and the parameter β is used for reducing the estimated value further.

3.2.3 Nonlinear GRAPPA

We previously mentioned this method in section 2.2.2. The method is known to reduce noise effectively in brain data [14].

For simplifying the complexity of the computation, we only use linear terms $\sqrt{2c}x_n, \dots, \sqrt{2c}x_1$ and part of nonlinear terms x_n^2, \dots, x_1^2 and remove other terms. As a result,

$$\phi(x) = [x_n^2, \dots, x_1^2, \sqrt{2c}x_n, \dots, \sqrt{2c}x_1] \quad (3.16)$$

We can simplify the kernel expression using vector presentation as follows.

$$\mathbf{x} \rightarrow [\mathbf{x}, \mathbf{x}^2] \quad (3.17)$$

where \mathbf{x}^2 denote the vector whose elements are the squares of elements of \mathbf{x} .

3.4 Results

We compare the performance of our method with the conventional GRAPPA for two sets of brain data. The first test data is a set of *in vivo* axial brain data scanned on a 3T MRI scanner (GE, Waukesha, Wisconsin, USA) using an eight-channel head coil with matrix size 256×256 [14]. The second test data is the *vivo axial* brain data scanned on a 1.5T MRI scanner (GE, Waukesha, Wisconsin, USA) using eight-channel receive-only head coil with data size 200×200 [28]. We choose $N_s = 7 \times 7$ in (3.13) and $\beta = \frac{3}{10}, \frac{1}{5}$ in (3.15) for linear and nonlinear GRAPPA cases, respectively. We set the VHF zone to be a third of the entire ACS so that the two third of low k_x -frequency components is excluded. The number of

columns and the number of blocks are $N_b = 2, N_h = 15$. The number of iterations is fixed as 5. We also fixed the reduction factor as $R=3$ which is commonly used in clinical MRI and changed the size of the ACS lines.

3.4.1 Case of linear GRAPPA

Figure 3.3 shows the NMSEs of the conventional MCMLI-FNF and the proposed iterative GRAPPA according to different net reduction factors (netR), which come from different ACS sizes of 48,42, ..., 18 at $R=3$. We see in Figure 3.3 that our method outperforms the conventional GRAPPA in NMSE over all netR.

In implementing MCMLI-FNF, we need upward-shifting operation by $bR+d$ for a matrix array such as $s((k_y + b)R + d, k_x + h; l)$ which results in down-side boundary problem. We solve the boundary problem by choosing circular shifting [29]. The circular shifting is easy to implement but may bring about discontinuities in the boundary and so somewhat performance degradation. Our proposed method uses a half of Hamming window whose size is N_bR in the down-side boundary to overcome the degradation [29]. Figure 3.4 shows the effectiveness of using Hamming window. After applying Hamming window to the conventional MCMLI-FNF, the NMSE is reduced. Figure 3.5 shows the results from the second brain data. These results also show that our proposed method effectively reduces the noise.

The experiments on image quality for the first brain data are shown in Figure 3.6. Our iterative method reduces the reconstruction errors compared to the conventional method. Figure 3.7 shows the results for the second brain data. The NMSE using our proposed method is much below the conventional GRAPPA. The reconstructed images by our method are more accurate than those by the conventional method.

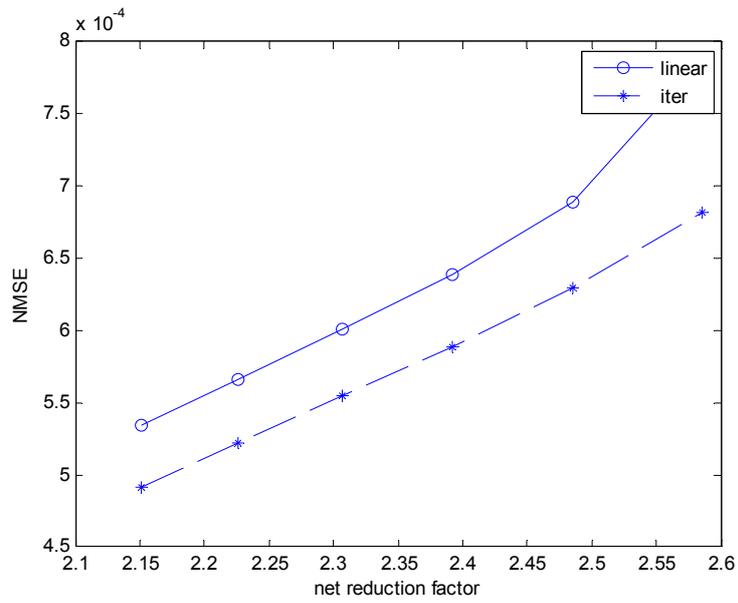


Figure 3.3 NMSE Comparison of the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) according to various net reduction factors for the first brain data.

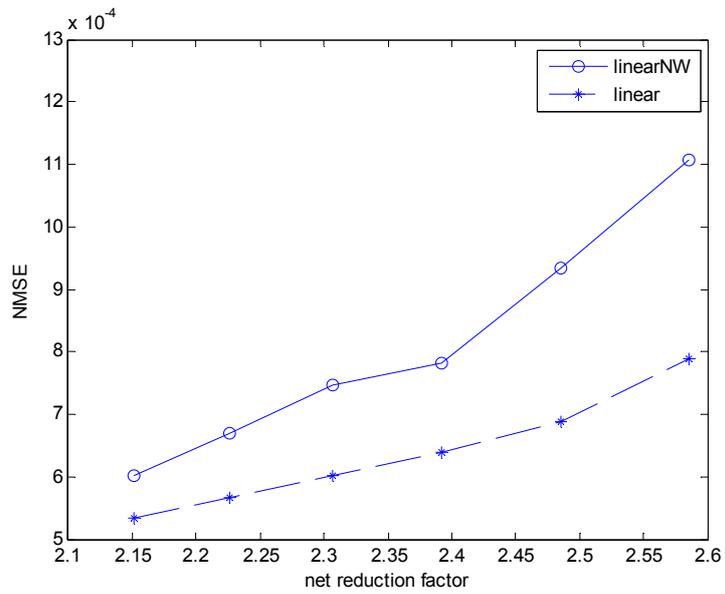


Figure 3.4 NMSE Comparison of the MCMLI-FNF (linear) and MCMLI-FNF without hamming window (linearNW) according to various net reduction factors for the first brain data.

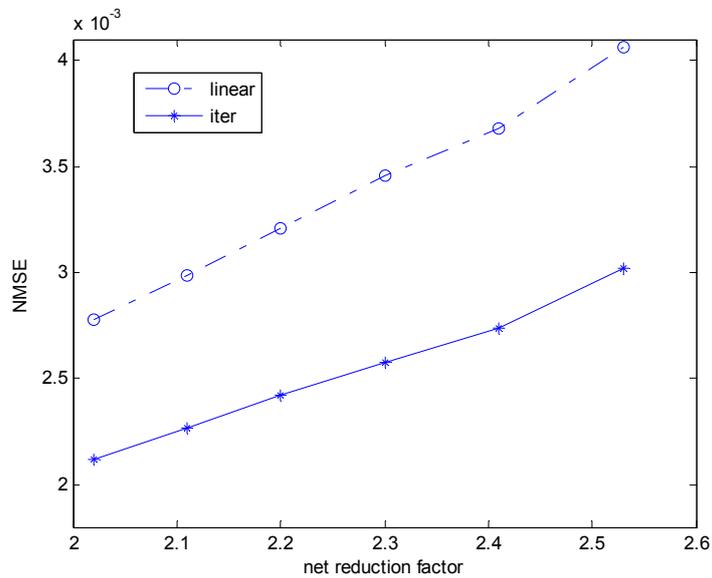
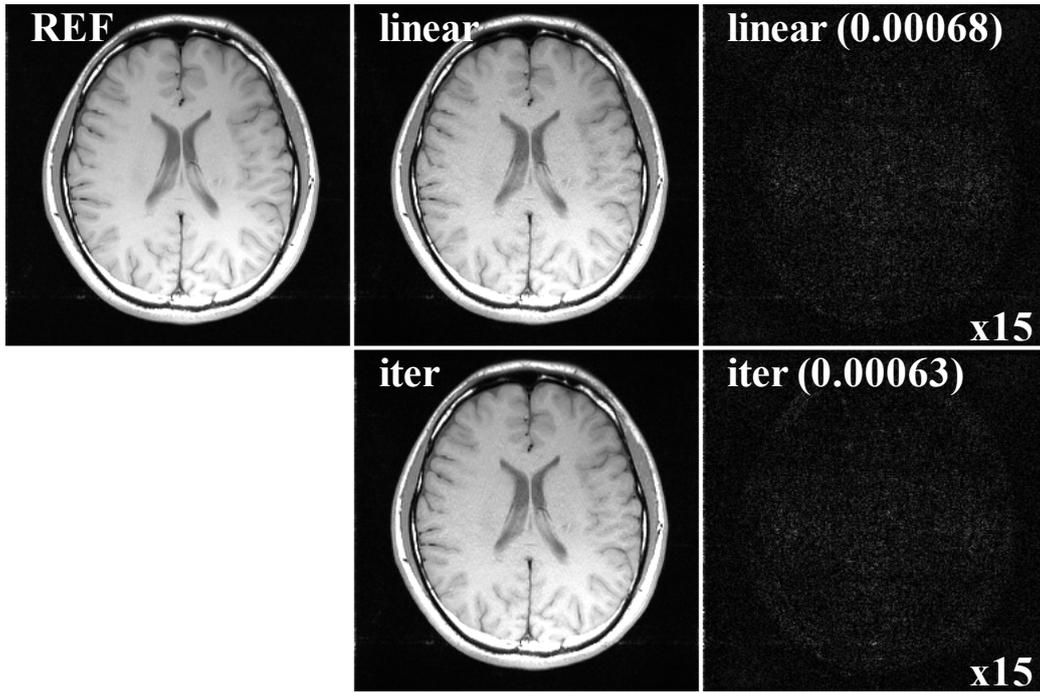
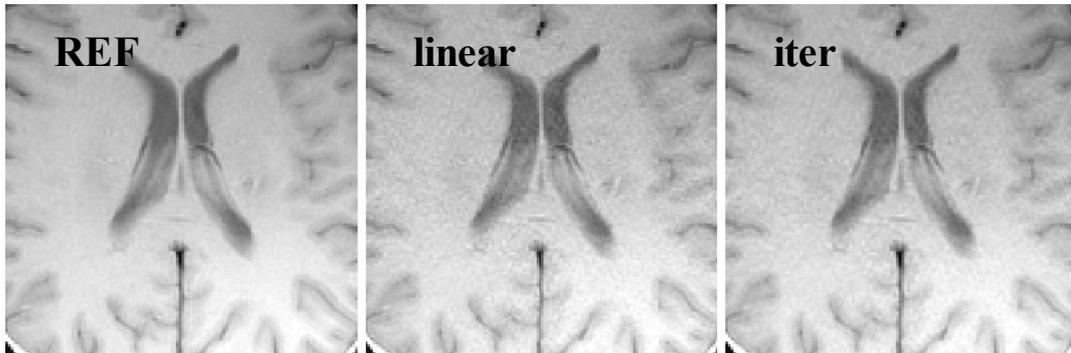


Figure 3.5 NMSE Comparison of the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) according to various net reduction factors for the second brain data.

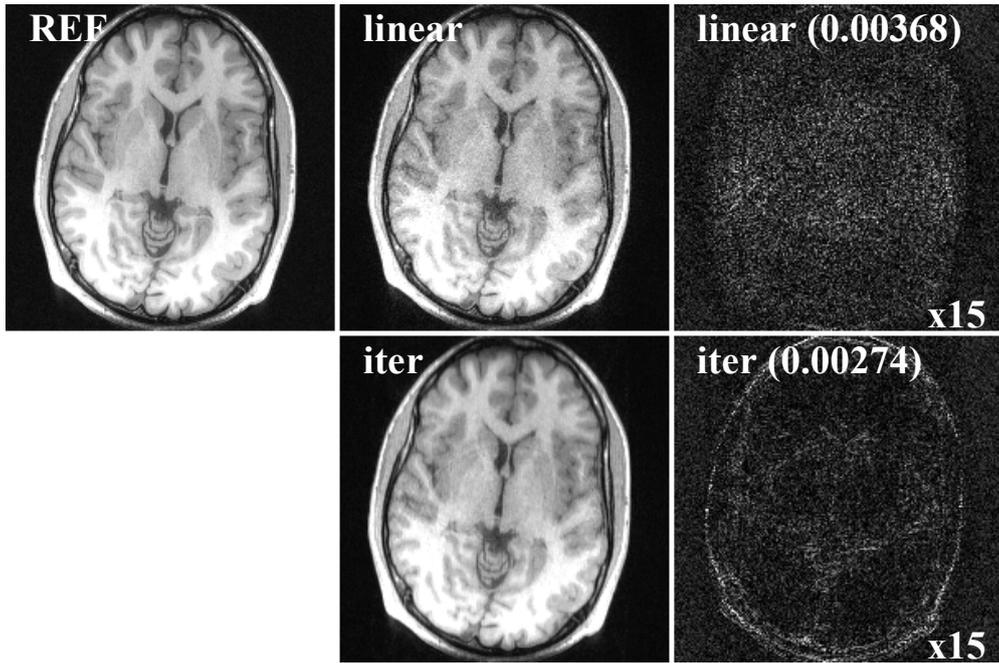


(a)

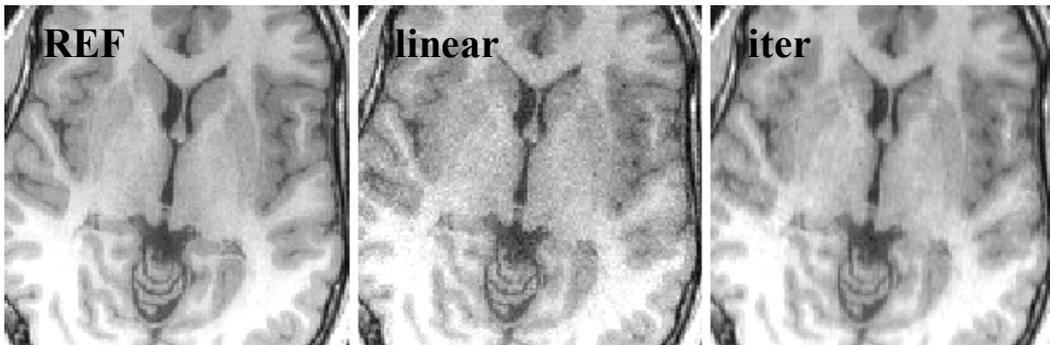


(b)

Figure 3.6 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) using ACS lines of 24 at R=3 (netR=2.49) for the first brain data.



(a)



(b)

Figure 3.7 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative GRAPPA (iter) using ACS lines of 24 at $R=3$ (netR=2.41) for the second brain data.

3.4.2 Case of nonlinear GRAPPA

Our proposed method is the combination of iterative Wiener filter and nonlinear kernel. Before performing nonlinear GRAPPAs, we normalized k-space data for stabilization of nonlinear GRAPPA so that all k-space data should be smaller than the maximum of sampled k-space data.

We compare our method with the conventional GRAPPA and nonlinear GRAPPA used in our lab. Also, we show the effect of the iterative Wiener filter and proposed nonlinear GRAPPA, which we can see here the improvement of our proposed method.

Figure 3.8 shows the NMSE comparison of various methods for the first brain data. Our proposed method is called ‘iterative NL Wiener’. In the figure, Chang’s linear GRAPPA means the version of removing constant and second-order kernels in Chang’s NL GRAPPA [14]. Chang’s linear GRAPPA shows the worst performance at netR=2.1. However, when the net reduction factor increases above netR=2.5, which means ACS lines are a few as 18, the performance of all nonlinear method is getting worse. Our proposed iterative NL Wiener filter method keeps the performance lowest. Figure 3.9 shows the NMSE comparison of various methods for the second brain data. Our proposed method shows further improved performance than the conventional method.

The experiments on image quality for the first brain data are shown in Figure 3.10. Our iterative NL method reduces the reconstruction errors compared to the conventional GRAPPA. Figure 3.11 shows the results for the second brain data. The NMSE using our proposed nonlinear method is much lower than the conventional GRAPPA. The reconstructed images using our method are more accurate than those using GRAPPA.

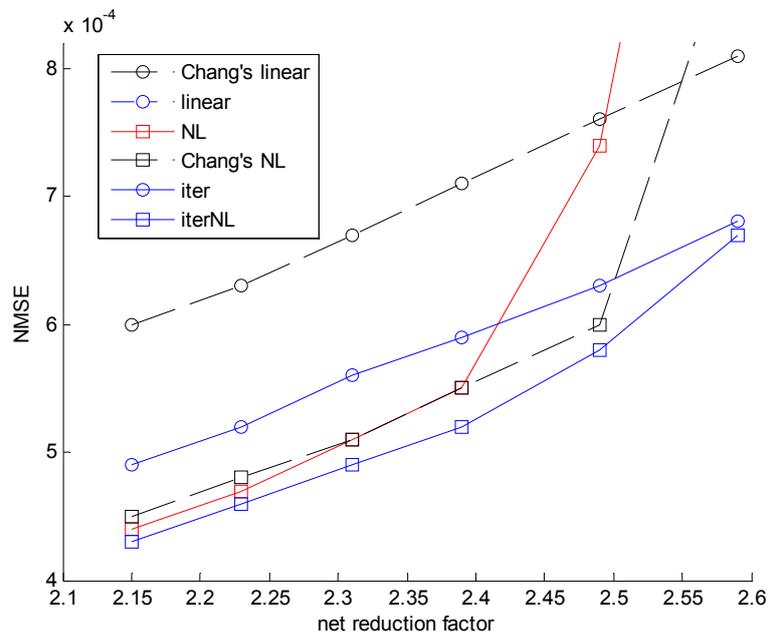


Figure 3.8 NMSE Comparison of 6 GRAPPAs according to various net reduction factors for the first brain data.

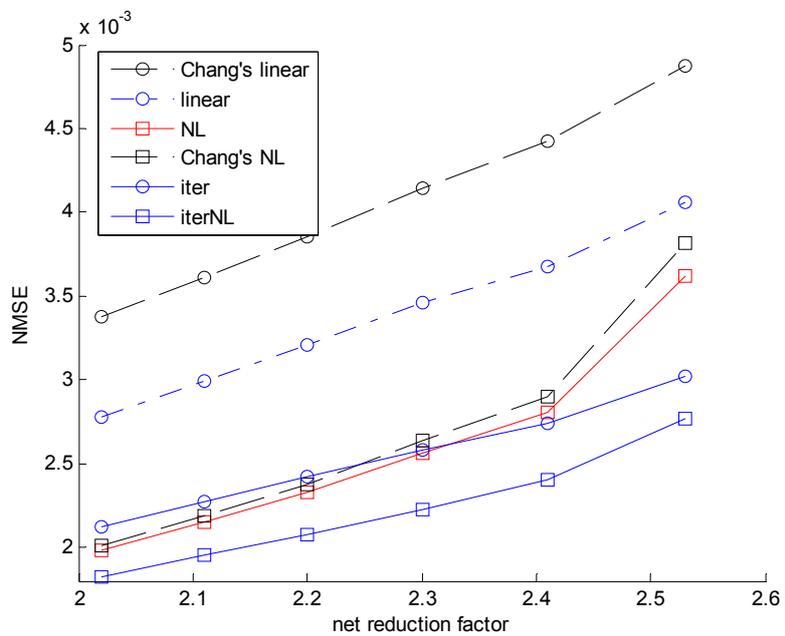
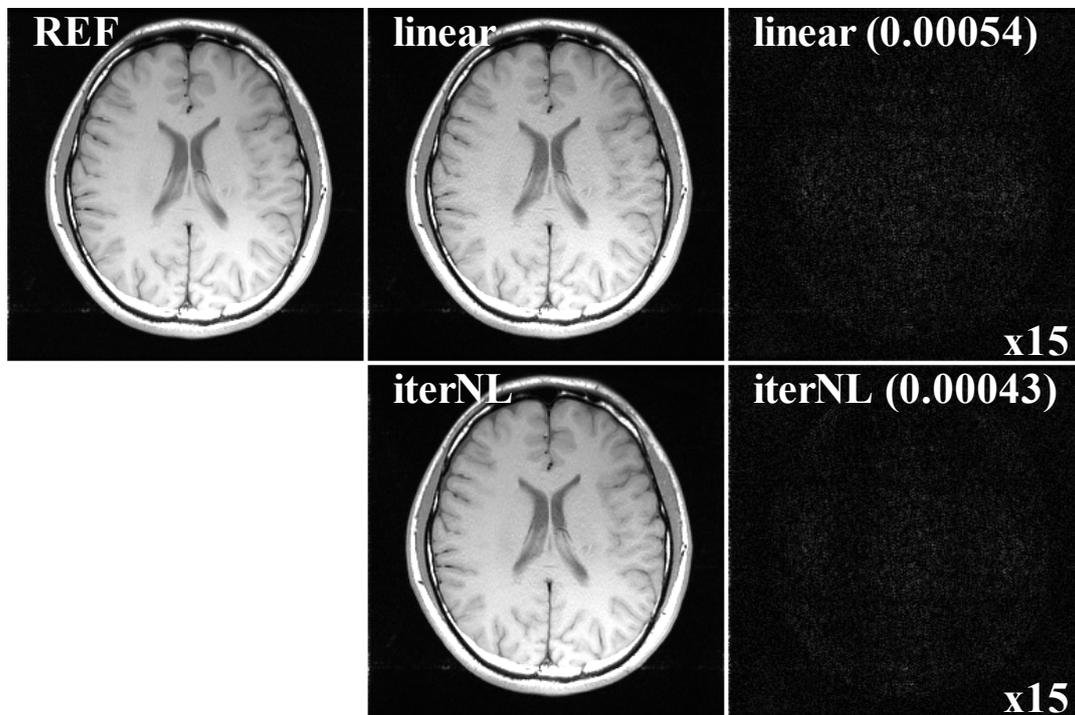
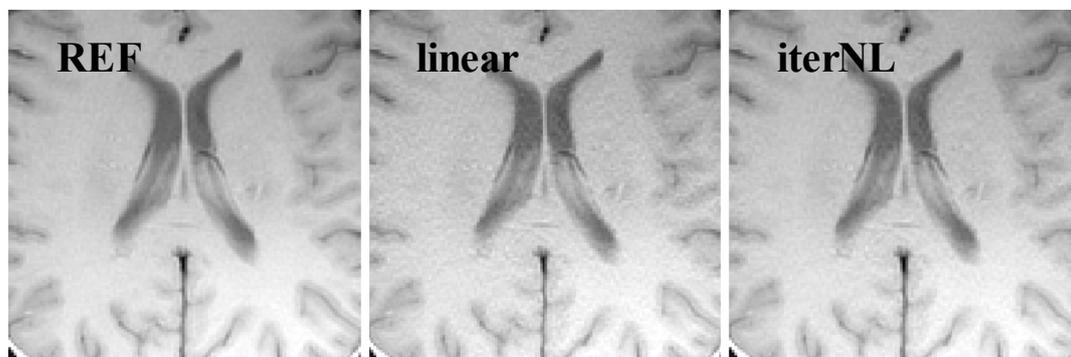


Figure. 3.9 NMSE Comparison of 6 GRAPPAs according to various net reduction factors for the second brain data.

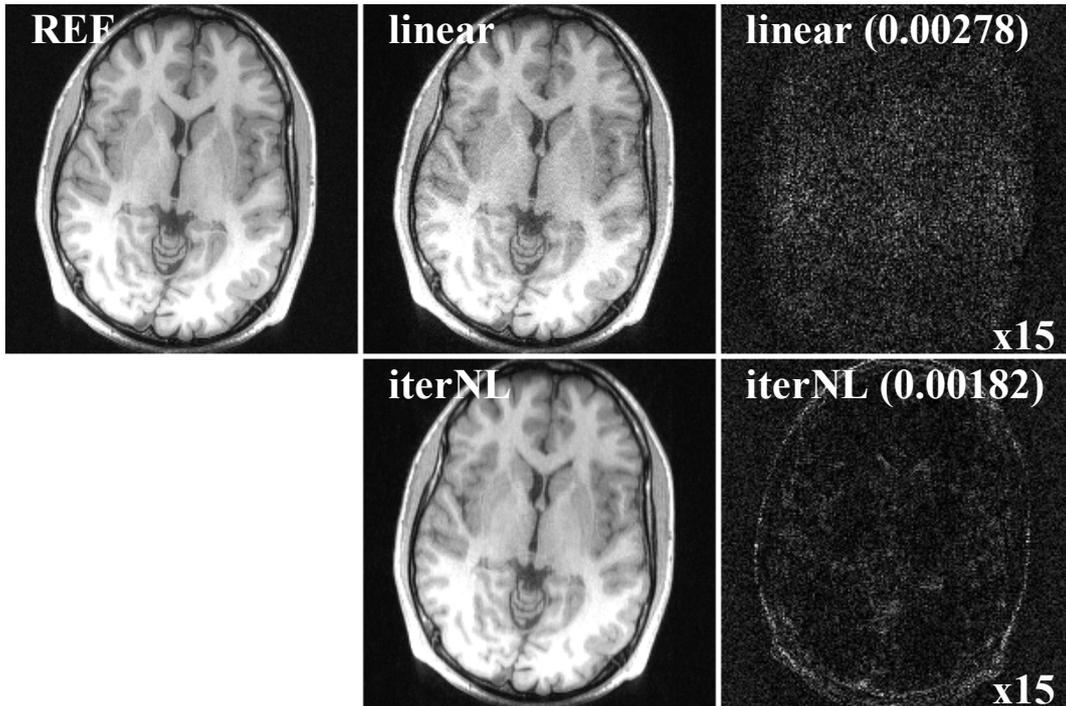


(a)

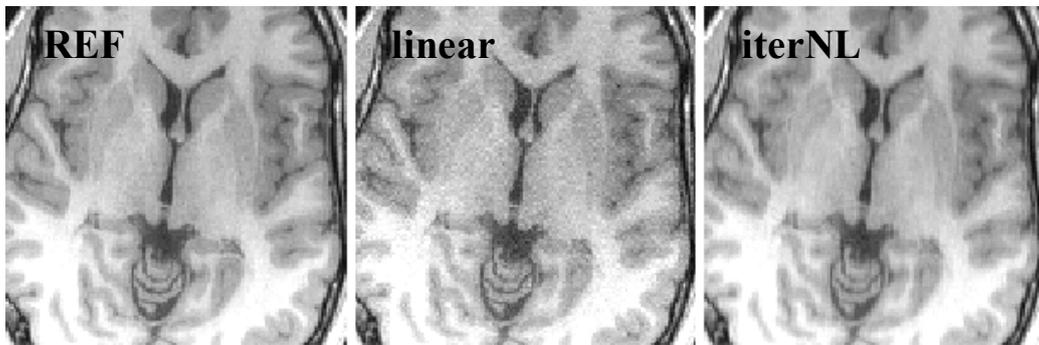


(b)

Figure 3.10 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative NL GRAPPA (iterNL) using ACS lines of 48 at R=3 (netR=2.15) for the first brain data.



(a)



(b)

Figure 3.11 Original image, reconstructed images, and error images (15x) for the conventional MCMLI-FNF (linear) and the proposed iterative NL GRAPPA (iterNL) using ACS lines of 48 at $R=3$ (netR=2.02) for the second brain data.

3.5 Conclusion

We have proposed a new iterative GRAPPA method using adaptive Wiener filter. Our proposed method outperforms the conventional GRAPPA when the reduction factor is $R=3$ in case of various sized ACS lines. Our proposed method improves the SNR when compared to the conventional GRAPPA. Especially, it was shown to greatly outperform Chang's nonlinear GRAPPA in small ACSs.

Chapter 4

Conflict-Cost Based

Random Sampling Design for

Parallel MRI

with Low Rank Constraints

4.1 Introduction

Compressed sensing is introduced to enable exact recovery of sparse signals from data sampled below what the Shannon-Nyquist sampling theorem [1] requires. Compressed sensing becomes very useful in MRI [2] because it allows the image to be acquired with much faster speed. For success of compressed sensing, the sampling needs to satisfy the incoherent requirement. Random sampling of the k-space (data domain) has been adopted in Sparse MRI [30] and many subsequent studies. It allows the aliasing artifacts to be noise-like such that they can be effectively removed using some nonlinear reconstruction algorithms. Many different random sampling patterns have been introduced such as power density function [30], Poisson disk [31-32], and variable density (VD) Poisson disk sampling [33] in order to make the sampling more effectively and satisfy the incoherent requirement for compressed sensing.

The challenge in sampling pattern design is how to sample most efficiently. In MRI, we usually take advantage of the fact that most important information is around the low

frequencies, especially zero frequency or direct current (DC), when we sample the Fourier domain (called k-space in MRI) of natural images. Therefore most random sampling methods consider full sampling in the center part and random sampling in the outer region.

Recently, Poisson disc sampling pattern or its advanced versions, VD Poisson sampling patterns have been used extensively. Poisson disc sampling pattern requires any two points to be separated by at least a minimum distance [34]. However, in the case of reduction factor of 2 or 3 which is typically used to obtain high quality reconstructions, it is difficult for all samples to satisfy the minimum distance requirement. This is because Poisson disc method usually uses dart throwing [34] to generate the random pattern and the pattern can be arbitrary. Also it is usually computationally expensive to generate a Poisson disc pattern, which can be an issue for real implementation or experiments [35].

In this work, we propose a new random sampling pattern using a probability model and cost function. In order to produce a desired random sampling pattern, we will first allocate samples to a set with the same probability using the probability function of a generalized Gaussian shape. We next will use a cost function to assign samples to points of the least cost so as to make the sampling pattern more incoherent. Then we reconstruct a brain image using SAKE (simultaneous auto-calibrating and k-space estimation) [28] with the proposed sampling pattern. We finally compare the performance of our random sampling mask with that of VD Poisson disk.

4.2 Method

Figure 4.1 shows the whole process of the proposed reconstruction. We first make a random sampling mask. Then we use the mask to randomly sample the k-space and apply SAKE to the k-space data randomly sampled to get the reconstruction image. We will give the explanation about section 4.2.1 Probability Model and section 4.2.2 Sample Selection Algorithm. The new method exploits a generalized Gaussian probability including a wide range of distributions from a uniform one to a Gaussian-like one severely centralized in a center frequency. Next is the algorithm of generating the proposed random sampling mask.

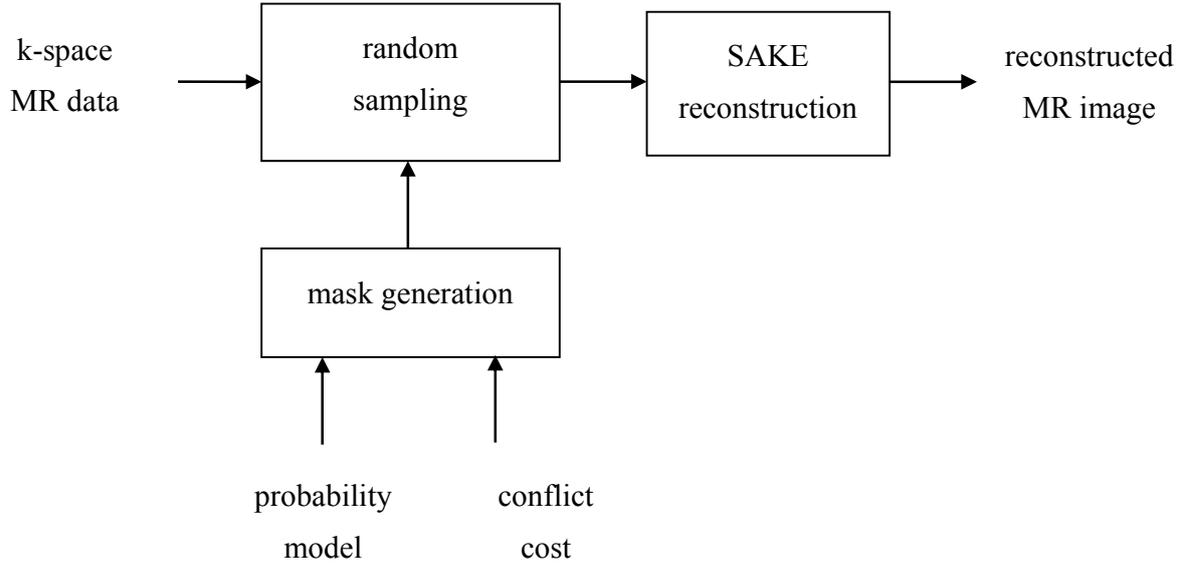


Figure 4.1 Flow chart of the reconstruction process

4.2.1 Probability Model

In order to control the distribution of sampling locations, one needs a probability model. Let $p(\vec{r})$ be the probability of assigning a sample to a spatial point \vec{r} where $\vec{r} = [r_x, r_y]$. Then $p(\vec{r})$ becomes the success probability of a Bernoulli random process. If $p(\vec{r})$ is a constant with regard to \vec{r} , then it shows a uniform distribution as in Poisson disks. As a good probability model, we choose an exponential function which is bounded in $[0, 1]$ while the power density function [30], which is used in some conventional random sampling for compressed sensing, does not satisfy the bounded property. Introducing a shape parameter into the function as in a generalized Gaussian probability density function (pdf) [36], we can control how much concentrated samples are around DC. Since DC and near-DC Fourier components contain meaningful information more than other regions, it is important to sample densely in those regions to produce good SNR.

The generalized Gaussian probability model of sample allocation using a shape parameter is formulated as

$$p(\vec{r}) = \exp \left\{ -\frac{1}{\mu_R} \left(\frac{\|\vec{r} - \vec{r}_0\|}{\|\vec{r}_0\|} \right)^\alpha \right\} \quad (4.1)$$

where \vec{r}_0 denotes a center point and α the parameter controlling the probability shape. The parameter μ_R depends on the reduction factor R , which satisfies the following constraint:

$$\sum_{\vec{r} \in I} p(\vec{r}) = \frac{|I|}{R} \quad (4.2)$$

where I stands for the set of all locations in the whole k-space and $|I|$ the size of I . Also, Eq. (4.2) is equivalent to the total number of the samples. Given an R , the parameter μ_R is determined based on (4.2) not explicitly but by numerical iteration such as bisection method. The number of samples is an integer that is not exactly the same every time, but we can get a number close to the reduction factor. We will talk about this more on section 4.2.3.

Grouping all points into sets of points of the same probabilities, we rewrite (4.2) into the following:

$$\sum_n \sum_{\vec{r} \in S_n} p(\vec{r}) = \frac{|I|}{R} \quad (4.3)$$

where S_n , $n=1, 2, \dots$, represents the set of all points of the same probability. We now determine the number of samples to be allocated to the points of the same probability as

$$N_n = \sum_{\vec{r} \in S_n} p(\vec{r}) \quad (4.4)$$

4.2.2 Conflict Cost

Considering the incoherence requirement of compressed sensing, assigning samples purely randomly according to the probability model may not meet the incoherence requirements. To address the issue, we introduce conflict cost for each sample allocation in order to distribute samples evenly. We define the conflict cost of a point \vec{r} to a sample \vec{s} as

$$\Delta c(\vec{r}; \vec{s}) = e^{(-\gamma \|\vec{r} - \vec{s}\|)} \quad \text{for } \vec{r} \in B(\vec{s}), \quad (4.5)$$

where $B(\vec{s})$ denotes the set of neighbor points within a distance d around \vec{s} and γ the exponential slope parameter. The slope parameter should be larger than 0 because farther neighbor samples have to contribute less to the cost. Therefore, the total conflict cost of a point \vec{r} is obtained by summing the conflict costs between the point \vec{r} and all its neighbor samples.

4.2.3 Sample Allocation Algorithm

Let M be a sampling matrix, and initialize $M(\vec{r}) = 0$ for $\vec{r} \in I$. Whenever a sample is allocated to a point \vec{r} , the corresponding mask value becomes $M(\vec{r}) = 1$. A conceptual sample allocation algorithm is given in [37] and a more practical one is shown below.

Next are the steps for the sample allocation algorithm based on the probability model and conflict cost.

Step 1: Initialize a sampling mask and a conflict cost array as $M(\vec{r}) = 0$ and $c(\vec{r}) = 0$ for $\vec{r} \in I$.

Step 2: Compute $p(\vec{r})$ for $\vec{r} \in I$ and the parameter μ_R according to (1) and (2).

Step 3: Group all points into sets S_n of points of the same probabilities. Let $n = 1, \dots, N_g$. Then $excess = 0$ and $S_{prev} = \emptyset$ where \emptyset means an empty set.

Step 4: Do the following operations for the set S_{N_g} of the highest probability to the set S_1 of the lowest probability.

(1) Compute the number of samples N_n to be allocated to S_n according to (4). Then get $N'_n = \text{round}(N_n - excess)$.

(2) Do the following operations for $m = 1$ to $m = N'_n$.

- Set $S_n \leftarrow S_n \cup S_{prev}$ and $S_{prev} = \emptyset$.
- Sort the conflict costs of all points \vec{r} in S_n .

- Find the points of minimum conflict cost.
- If there are multiple minimum points, then select a point \vec{s} randomly.
- Set $M(\vec{s}) = 1$ and $c(\vec{s}) = \infty$.
- Set $c(\vec{r}) \leftarrow c(\vec{r}) + \Delta c(\vec{r}; \vec{s})$ for $\vec{r} \in B(\vec{s})$, where $\Delta c(\vec{r}; \vec{s})$ is given in (5).

(3) If $N'_n > 0$, then $excess \leftarrow N'_n - (N_n - excess)$. Else $S_{prev} \leftarrow S_n$.

The reason why we set the conflict cost of \vec{s} to be infinity is that the same point should not be selected as a sample point again in the following steps. For more practical implementation, we round N_n in Step 4 (1) into an integer because the number of samples should be an integer. However, excess or shortage in sum of probabilities caused by the integer rounding should be compensated in the next iteration. Especially, S_{prev} is for considering S_n of $N_n = 0$ in the next iteration.

Next are the steps for the image reconstruction algorithm.

4.2.4 Image reconstruction

We use SAKE as an image reconstruction. SAKE is one of the parallel imaging reconstruction algorithms using low rank completion. Next is a brief review about SAKE.

Unlike GRAPPA [6], one of the well-known parallel imaging techniques, which needs ACS (auto calibration signal) lines to reconstruct images from undersampled multi-channel data, SAKE is an algorithm to reconstruct images from randomly undersampled data without calibration data. SAKE first connects all the coil data in series and the connected data are reformulated as the shape of the Hankel matrix used for Cadzow's signal enhancement. As a result, the reconstruction becomes a low rank matrix completion problem in k-space. The solution is obtained by singular value thresholding, especially hard-thresholding. The SAKE reconstruction is formulated as follows:

$$\text{Minimize } \|Dx - y\|^2 \tag{4.6}$$

subject to $\text{rank}(A) = k$,

$$x = H^+(A)$$

where D denotes the sampling operation related to the sampling matrix M mentioned in section 2.3, x the desired image, y the acquired data, A the low-rank data matrix and H^+ the pseudo-inverse operator.

We use the nonlinear kernel method again, which is explained in section 3.2.3. We hereafter call the original SAKE reconstruction ‘linear’ but the reconstruction with the nonlinear kernel ‘nonlinear’.

4.3 Results

In order to evaluate the effectiveness of the proposed random masks, we compare the SAKE reconstructions using VD Poisson discs [33] and the proposed random masks. The VD Poisson disk is generated from a Poisson disk using μ -law [38].

In general, the information of a Poisson disk is expressed as locations of samples on $[-1,1] \times [-1,1]$. If we map the continuous space linearly into digital image space $[1, N_y] \times [1, N_x]$, we get the very Poisson disk. However, if we map the space nonlinearly into the image space, we may get variable density (VD) Poisson disk. In this case, we choose the μ -law for the nonlinearity, which is one of non-uniform quantization methods and expressed as

$$r' = \text{sgn}(r) \frac{\ln(1 + \mu|r|)}{\ln(1 + \mu)} \quad (4.11)$$

where μ is the parameter for controlling the nonlinearity. The symbol r is the input of μ -law and r' the output. In relation with disk generation, r corresponds to the radius or distance of each sample location from the origin and r' that of the new location. Due to the nonlinearity, the effective sizes of cells around $(0,0)$ increase so that the densities become lower and those of cells near boundaries decrease so that the densities become higher.

In order to get the effect that sample densities around the center become denser and those near mask boundaries become rarer, we choose the modified μ -law as follows.

$$r' = \text{sgn}(r) \left[1 - \frac{\ln\{1 + \mu(1 - |r|)\}}{\ln(1 + \mu)} \right] \quad (4.12)$$

Figure 4.2 shows the input-output relation of the original μ -law and that of the modified μ -law. We see that the modified μ -law compresses an interval near the center but expands that near boundaries. Figure 4.3 shows a Poisson mask and a VD Poisson mask generated by μ -law from the Poisson disk at $R=3.0$. We see that using the modified nonlinearity, the densities around the center become higher and the densities near boundaries become lower. We generate Poisson disks using the Poisson disk generation code from the website <http://www.cs.virginia.edu/~gfx/pubs/antimony/> [39]. For fair comparison, we also used three masks provided by one of the authors of SAKE [28], called Shin masks here. We select the parameter μ as $\mu=0.4(R-1)$ to generate a VD Poisson disk so that the μ -law masks seem similar to Shin masks.

Figure 4.4 shows Shin and μ -law masks. The μ -law patterns are shown to be very close to the Shin masks. We will compare the performance of μ -law VD Poisson disk with that of our proposed mask. In the following experiments, we generate our masks and μ -law masks randomly fifty times at each reduction factor R to investigate the averaged behavior of the masks. Because of the way that the VD Poisson masks are generated, it is hard to get the exact reduction factor we want, while our masks show exact reduction factors. We place a fully sampled area called core in the center of all the masks and choose the circle of radius 3.

In (4.5), the slope parameter γ is set to be $\ln 4$ and the maximum neighbor distance $d=[1 + R]$. The MRI data first used for test is the vivo axial brain data scanned on a 1.5T MRI scanner (GE, Waukesha, Wisconsin, USA) using an eight-channel receive-only head coil with data size 200×200 [28]. The second scanned dataset was an axial brain image acquired on a GE 3T scanner (GE Healthcare, Waukesha, WI) using an 8-channel head coil with data size 256×256 from <http://www.acsu.buffalo.edu/~jlv27/> [14].

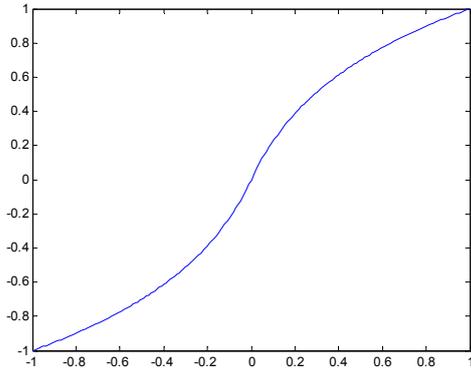
Figure 4.5 shows the masks according to various shape parameters α at $R=3$. The mask of $\alpha = 0$ is close to uniform random mask and $\alpha = 2.0$ shows the Gaussian mask. We see that

one can easily control the concentration of samples around DC in the proposed mask by varying the shape parameter.

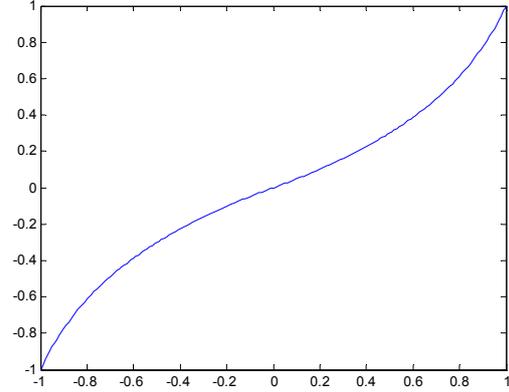
The magnification of center parts of the masks is shown in Figure 4.6. We can see that samples in the center part of our mask are more uniformly distributed and especially along circles than those of VD Poisson disks by μ -law and Shin.

Table 1 shows means and standard deviations of three types of masks at different reduction factors. This represents that our masks show exact reduction factors but VD Poisson masks yield reduction factors somewhat deviated from desired ones and varied.

When we look at mask generation cases of no conflict cost, no rounding compensation and normal in section 4.2.3, the differences are clear as shown in Figure 4.7-9. When we do not have conflict cost, we can see in case of low reduction factor near 2 or 3 that there are many samples are closely located which do not satisfy with incoherence. However, our method shows good incoherence even in low reduction factor. If we do not use the rounding compensation, then sometimes rounding cause lack of samples and we can see the circular borders like waves. Also, we had experiments on our proposed method and one without rounding compensation. Even though it is observed that there is no big improvement in NMSEs and MCCs, we see that our masks hold the continuity in sample density well.

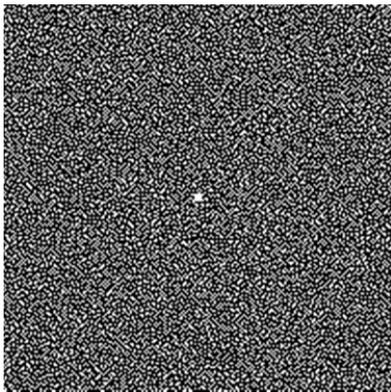


(a)

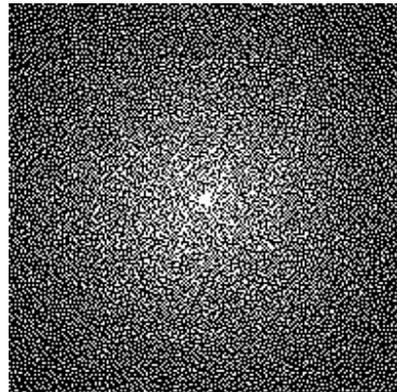


(b)

Figure 4.2 Input-output relation of μ -law: (a) μ -law of $\mu = 5$ and (b) its modified one



(a)



(b)

Figure 4.3 VD Poisson mask generated by μ -law from a Poisson disk at $R=3.0$: (a) Poisson and (b) VD Poisson

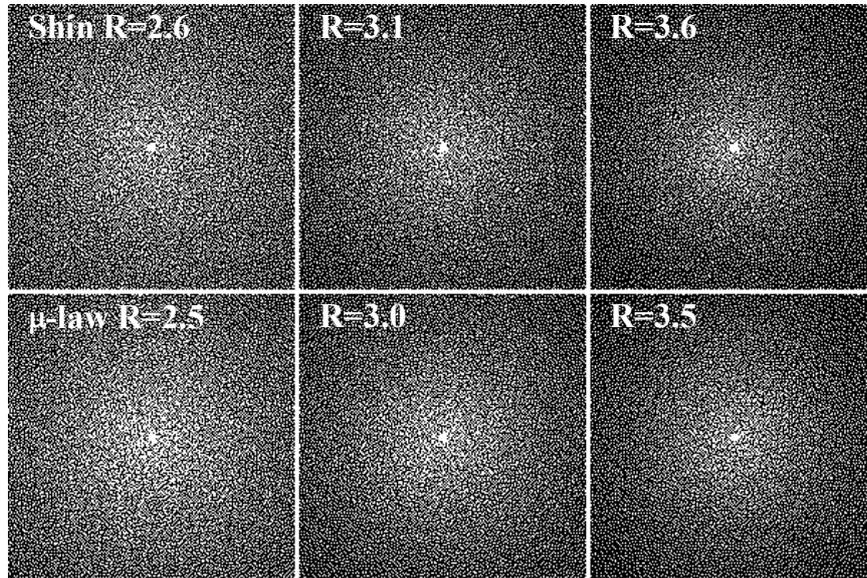


Figure 4.4 Shin and μ -law masks according to various reduction factor R .

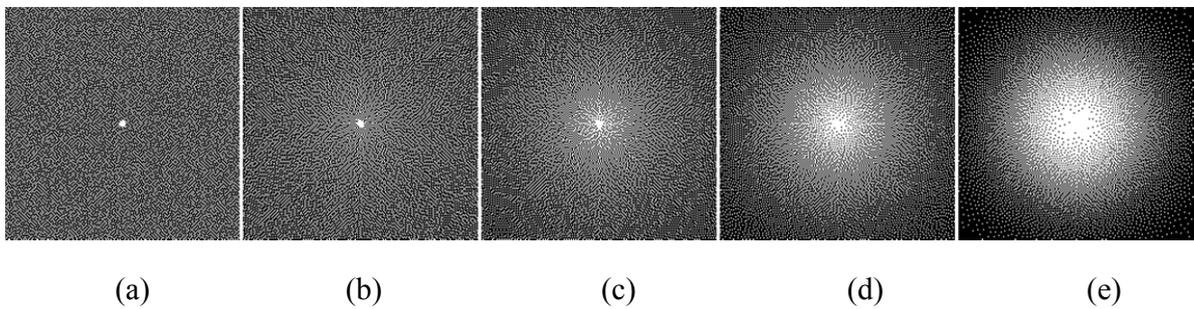
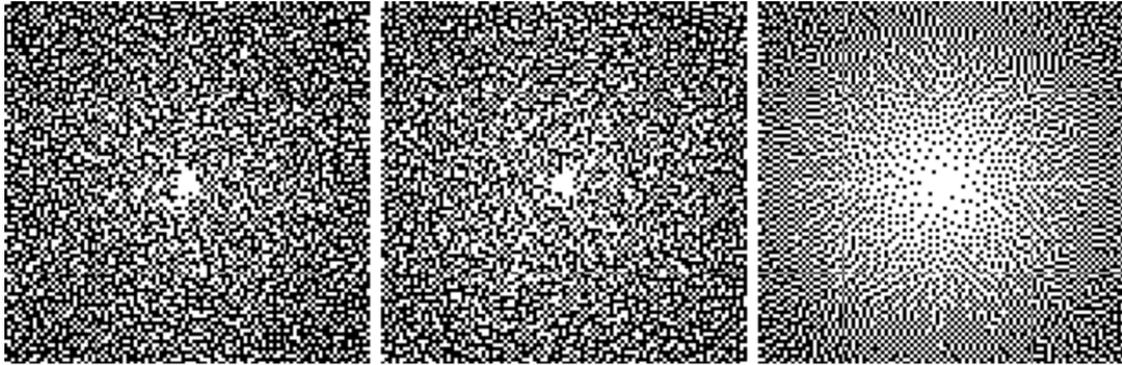


Figure 4.5 Proposed mask using different shape parameter at $R=3$: (a) 0.0, (b) 0.2 (c) 0.5, (d) 1.0, and (e) 2.0



(a)

(b)

(c)

Figure 4.6 Magnification of center parts of various masks at $R=3$: (a) Shin, (b) μ -law, and (c) proposed masks.

Table 4.1 Means and standard deviations of three types of masks at different reduction factors.

	R=2.5	R=3.0	R=3.5
μ -law	(2.505, 0.013)	(3.010, 0.016)	(3.494, 0.024)
Proposed	(2.500, 0.000)	(3.000, 0.000)	(3.500, 0.000)

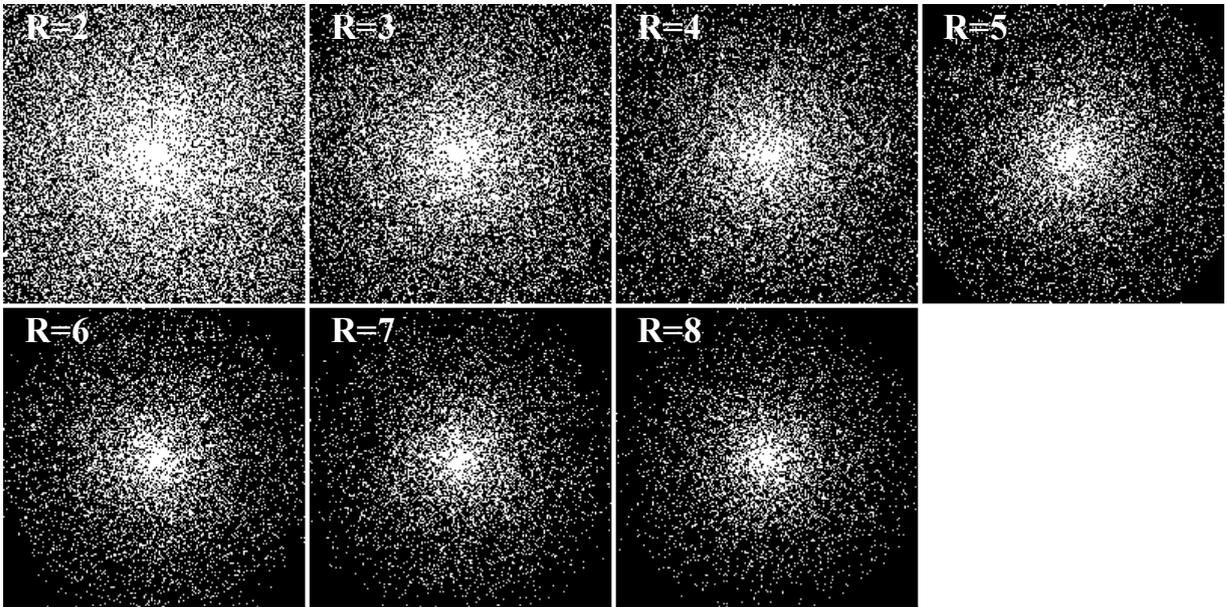


Figure 4.7 Mask generation using generalized Gaussian probability random sampling without conflict cost according to various reduction factors

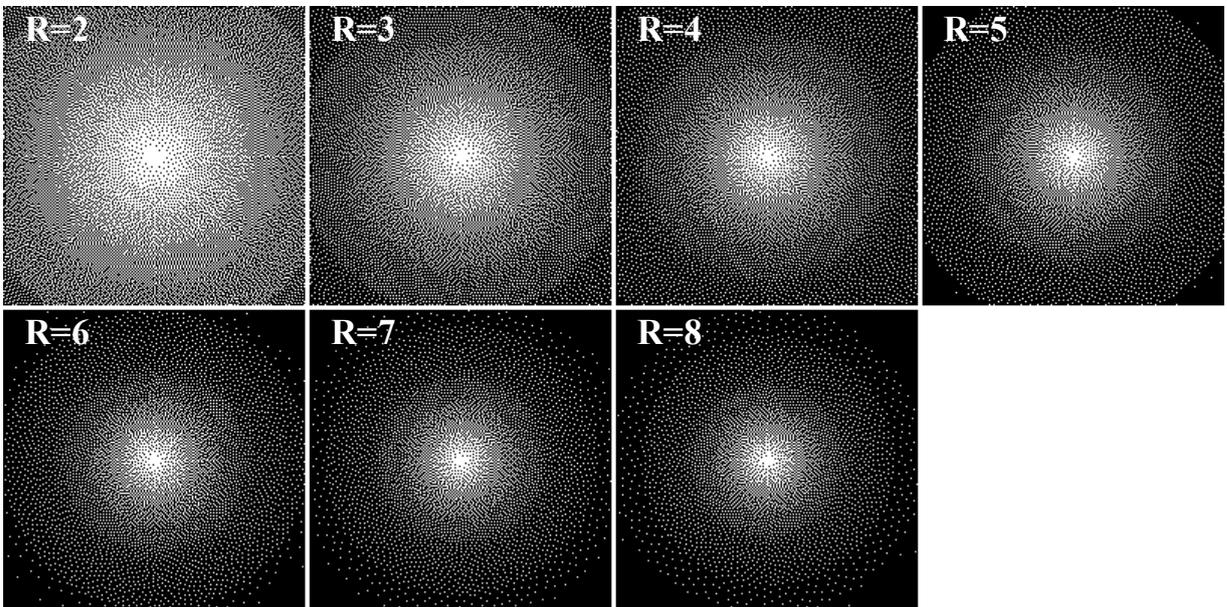


Figure 4.8 Mask generation using generalized Gaussian probability random sampling with conflict cost and without rounding compensation according to various reduction factors

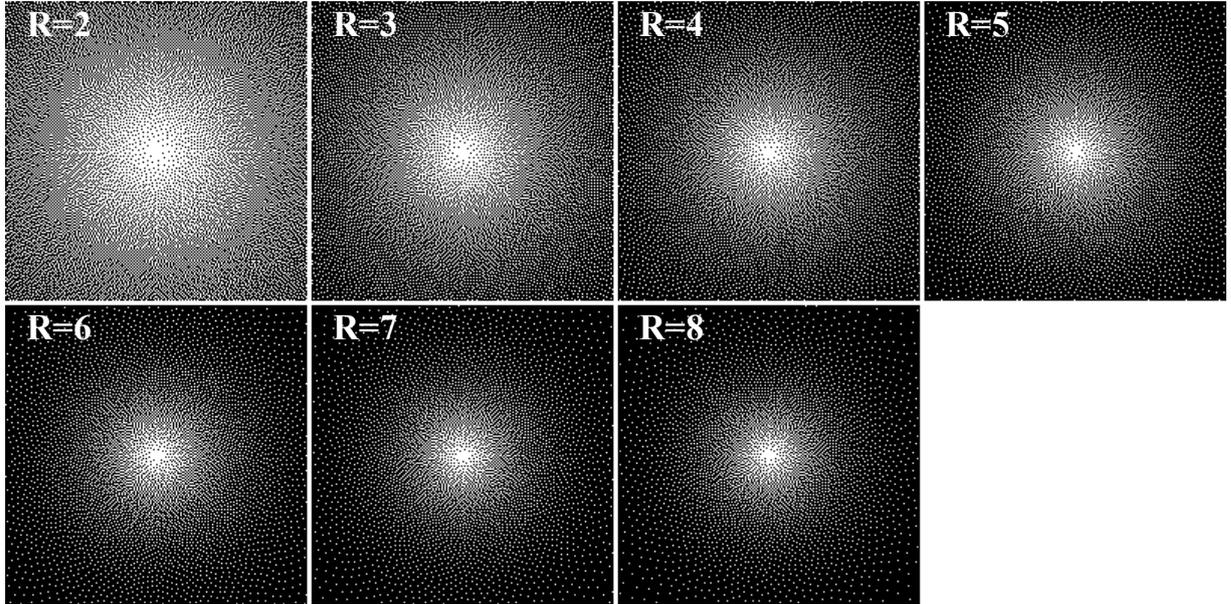


Figure 4.9 Proposed mask generation according to various reduction factors

4.3.1 Case of linear SAKE

We choose the SAKE parameters as $w_{nthresh}=1.8$ and $sakeiter=20$ except $sakeiter=15$ in case of our mask for `brain_SAKE` data. Figure 4.10 shows the normalized mean square error (NMSE) of SAKE using the proposed mask with $R = 3.0$ according to the shape parameter α . The number of SAKE iterations is set to 15 for $\alpha > 0.5$, and to 100, 45, and 30 for $\alpha = 0, 0.2, 0.4$, respectively. In the figure, the NMSE of $\alpha = 1.0$ is shown to be minimum and other shape parameters yield higher NMSEs. We can see that NMSE of $\alpha = 1.0$ is somewhat less than that of $\alpha = 2.0$ which indicates Gaussian function.

Correlation coefficient is widely used in signal and image processing community. In order to analyze the correlation of sampling noise, we will introduce correlation coefficient into our experiment. The correlation coefficient of two random variables X and Y is defined by

$$\rho = E \left[\frac{XY - E[X]E[Y]}{\sqrt{\text{Var}(X)\text{Var}(Y)}} \right] \quad (4.13)$$

We then consider a random process $X(\vec{r})$ which is a function of location \vec{r} . Then the correlation coefficient function (CCF) can be defined by

$$\rho(\vec{q}) = \mathop{\text{E}}_{p \in I} \left[\frac{X(\vec{r})X(\vec{r} + \vec{q}) - \text{E}[X(\vec{r})]\text{E}[X(\vec{r} + q)]}{\sqrt{\text{Var}(X(\vec{r}))\text{Var}(X(\vec{r} + q))}} \right] \quad (4.14)$$

where \vec{q} is called lag or displacement. The CCF of the reference image at $R=3$ for brain_SAKE data is shown in Figure 4.11. We can see in Figure 4.11 that the correlation coefficient of displacement 1 is very high as 0.938.

The CCF of the error image of proposed mask at $R=3$ for brain_SAKE data is shown in Figure 4.12. We can see in Fig. 4.12 that the correlation coefficient of displacement 1 is very low as 0.054, which means that it is close to white noise.

The CCF of the error image of VD Poisson mask at $R=3$ for brain_SAKE data is shown in Fig. 4.13. We can see in Fig. 4.13 that the correlation coefficient of displacement 1 is somewhat high as 0.125, which means that it is less incoherent than that of proposed mask. Figure 4.14 shows the comparison of CCFs of two masks for brain-SAKE data.

The effect of conflict cost for the proposed generalized Gaussian (GG) mask is shown in Figure 4.15 as a function of the reduction factor R . Figure 4.15(a) shows the NMSE comparison of SAKE using the proposed mask with and without conflict cost. We can clearly see that our method with conflict cost has less noise than that without. Figure 4.15(b) shows the maximum correlation coefficients (MCC) between pixels and their eight nearest neighbors for reconstruction error images. Since the ideal reconstruction has noise-like reconstruction errors, we can say that the smaller MCC is, the more incoherent the sampling mask is [40].

The normalized mean square errors (NMSEs) of various types of masks applied to SAKE according to $R=3.0$ are plotted in Figure 4.16 (a). The NMSEs curve for all method show the mean curve with the minimum and maximum for fifty masks. We see in the figure that the proposed random mask shows the best performance and least deviations all over the reduction factors. Figure 4.16(b) shows that the MCCs of our proposed masks are less than

half of those of VD Poisson disk all over the reduction factors. The correlation coefficients for our proposed method are also shown to be less varied than those of VD Poisson disk. It suggests that our mask is more incoherent than the VD Poisson masks.

The experiments on the brain data [28] are shown in Figure 4.17. Our method using the proposed masks pattern reduces the reconstruction errors compared to the original method using the VD Poisson disks.

Figure 4.18 shows the results from the second brain data. The NMSE using our sampling mask is below the mask used in the original SAKE paper. The reconstructed images of the brain using our method are more accurate than those using the VD Poisson disk.

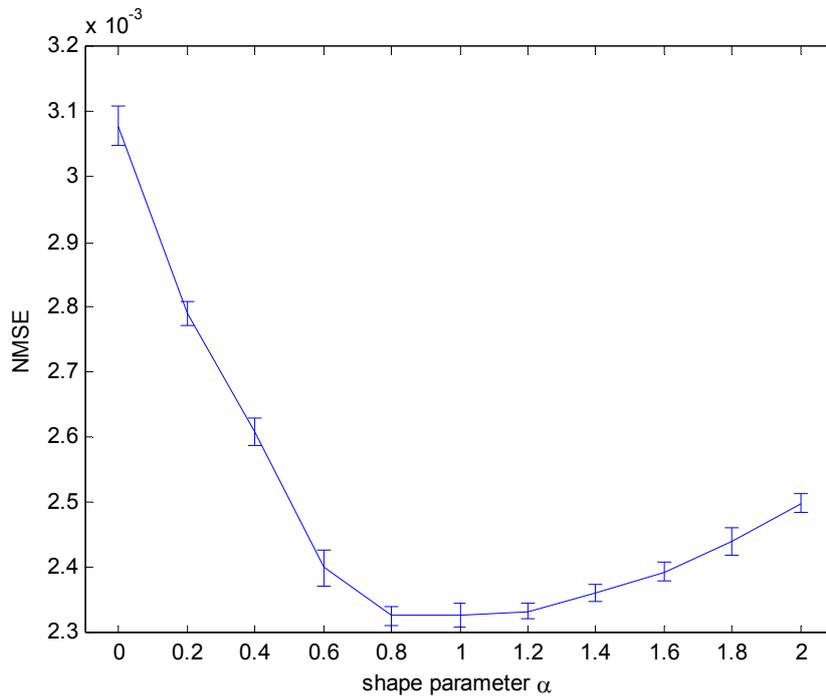


Figure 4.10 NMSE of SAKE reconstruction method using proposed random mask of $R=3.0$ according to the shape parameter α .

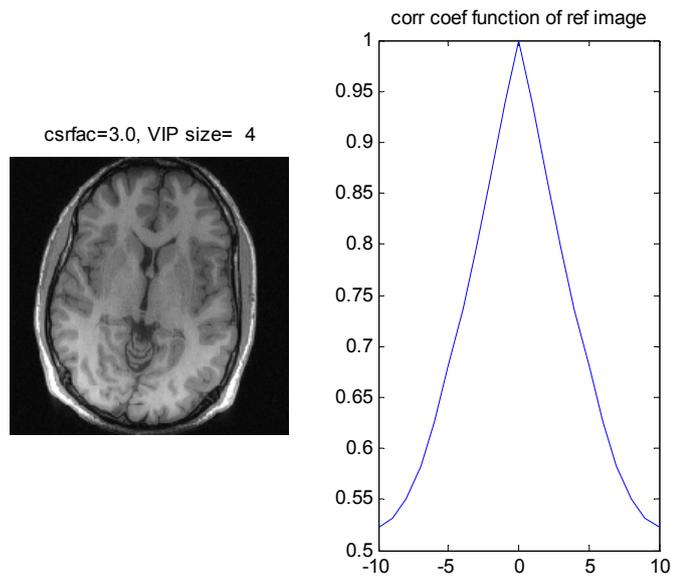


Figure 4.11 Reference image and its CCF $\rho(\tau, 0)$ of horizontal lag for brain-SAKE data.

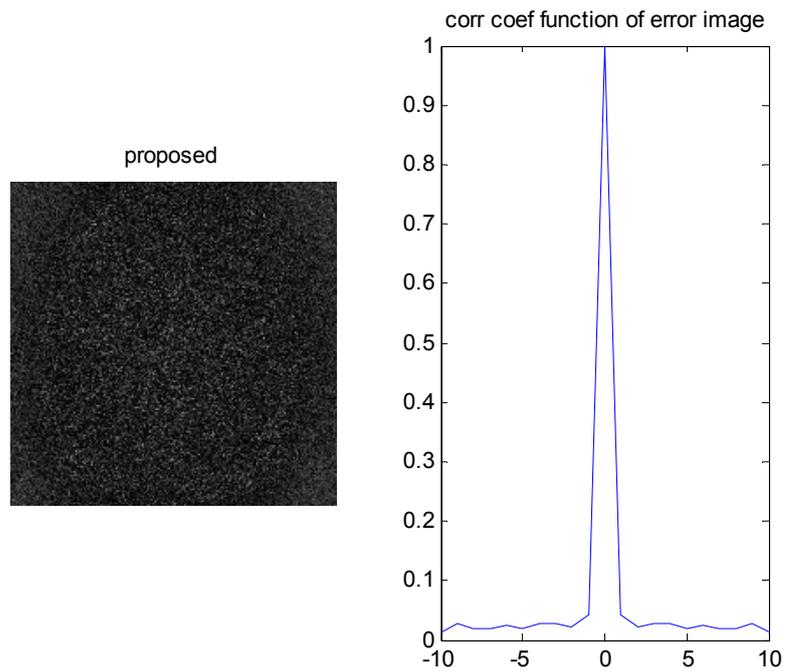


Figure 4.12 Error image of proposed mask and its CCF $\rho(\tau, 0)$ of horizontal lag for brain-SAKE data.

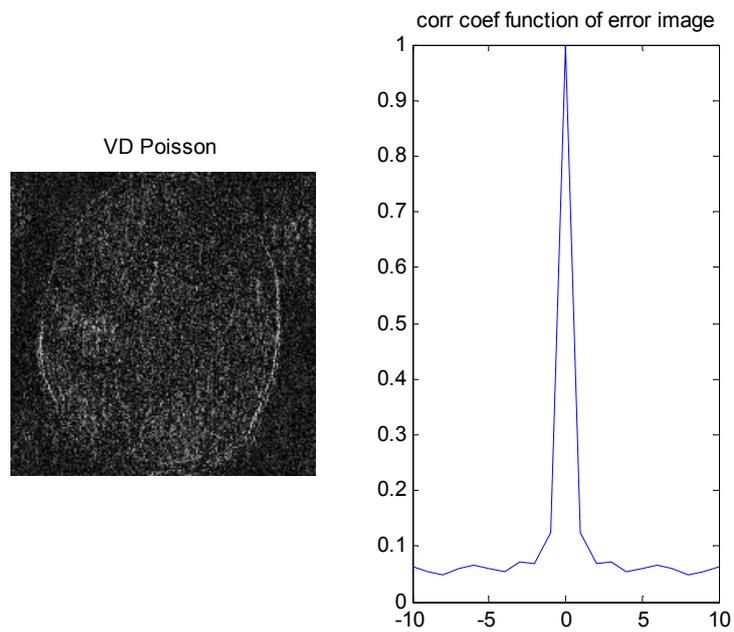


Figure 4.13 Error image of VD mask and its CCF $\rho(\tau, 0)$ of horizontal lag for brain-SAKE data.

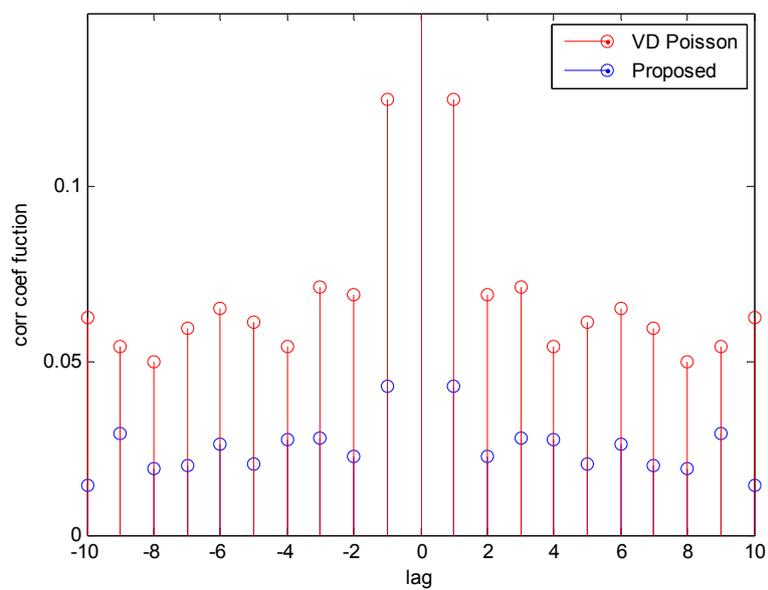
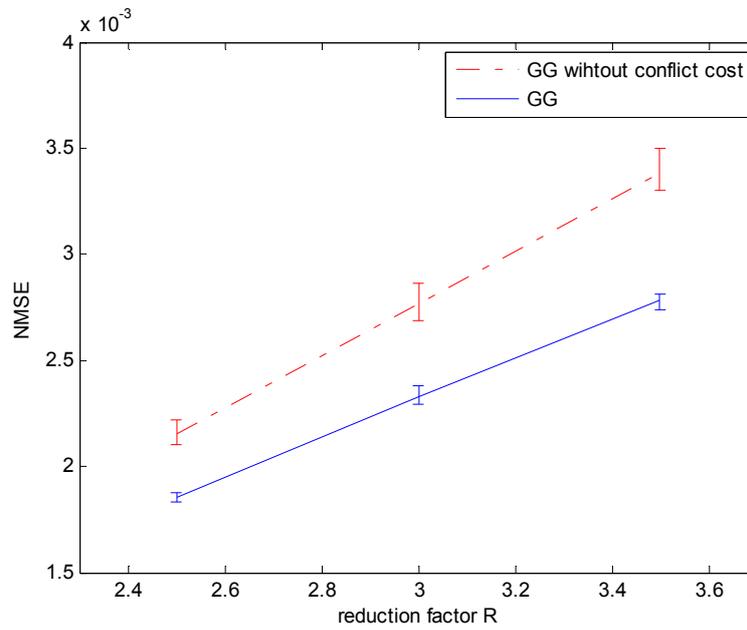
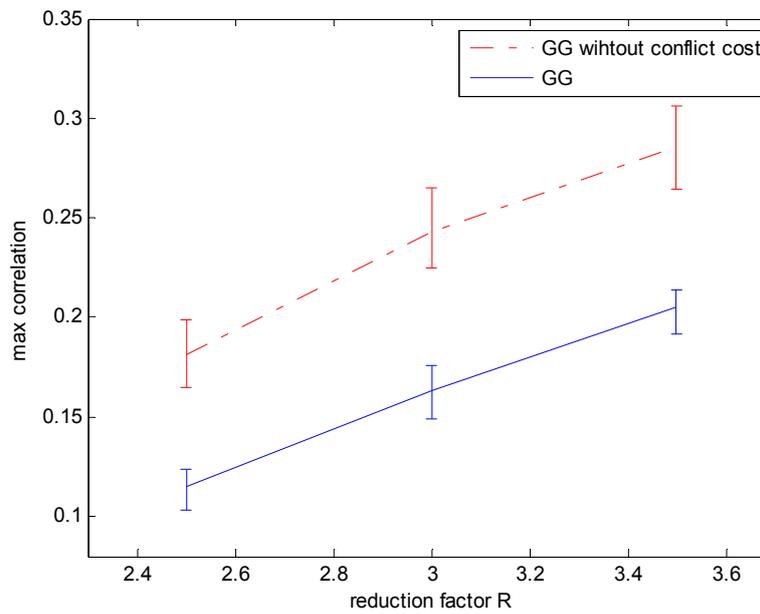


Figure 4.14 Comparison of CCFs of two masks for brain-SAKE data.

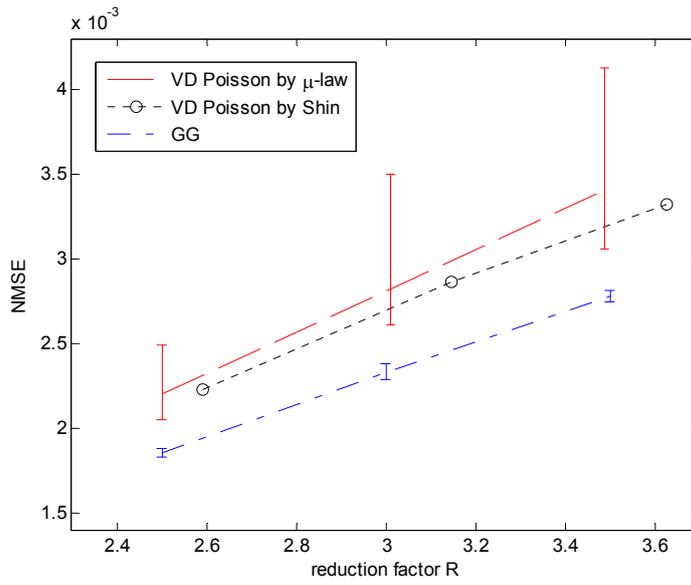


(a)

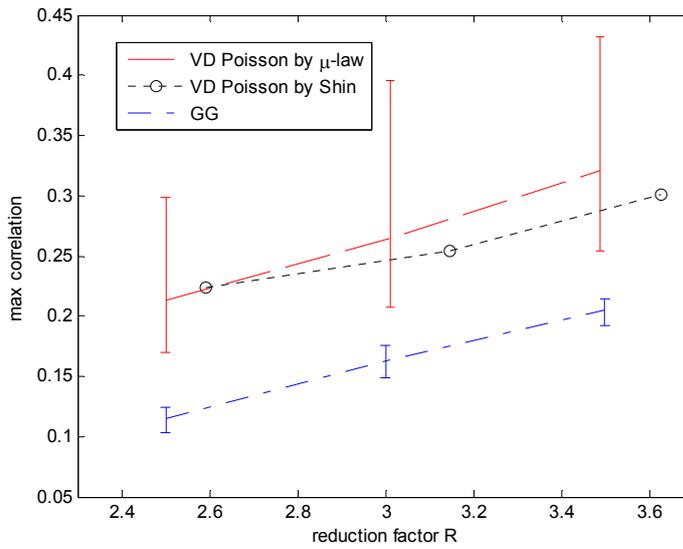


(b)

Figure 4.15 Effect of conflict cost. Comparison of SAKE reconstruction (a) NMSEs and (b) maximum correlations using the proposed generalized Gaussian (GG) masks with and without conflict cost as a function of the reduction factor R .

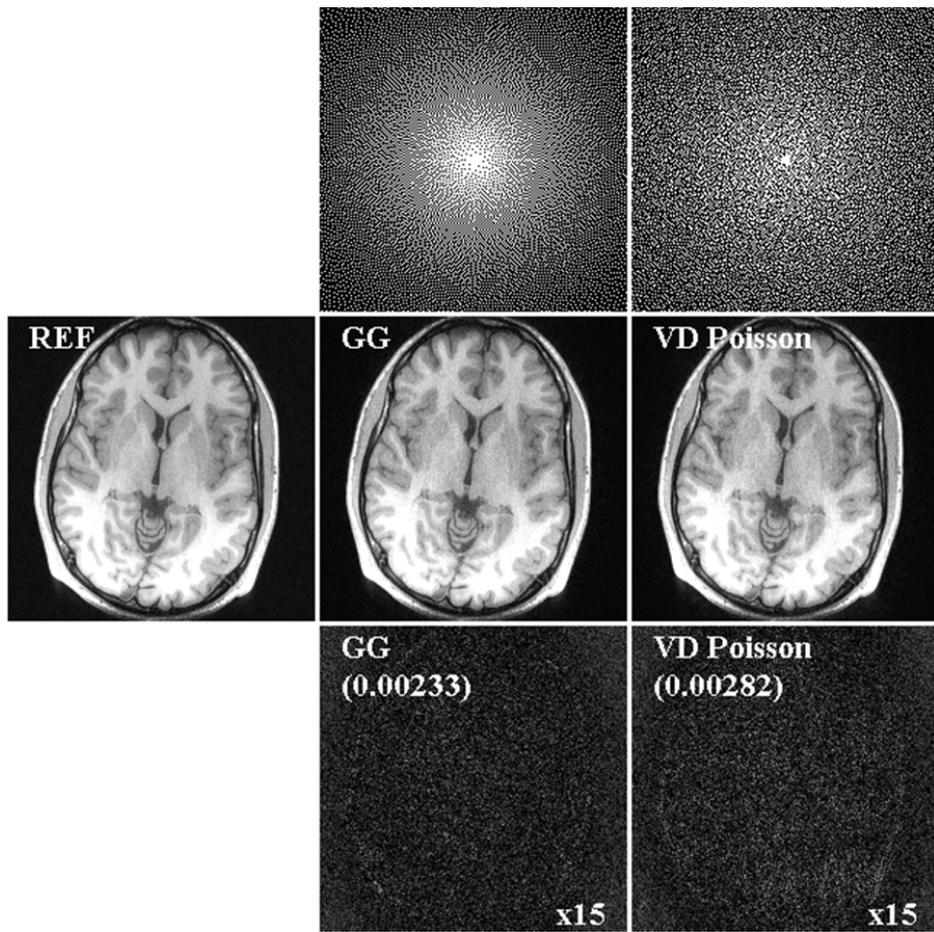


(a)

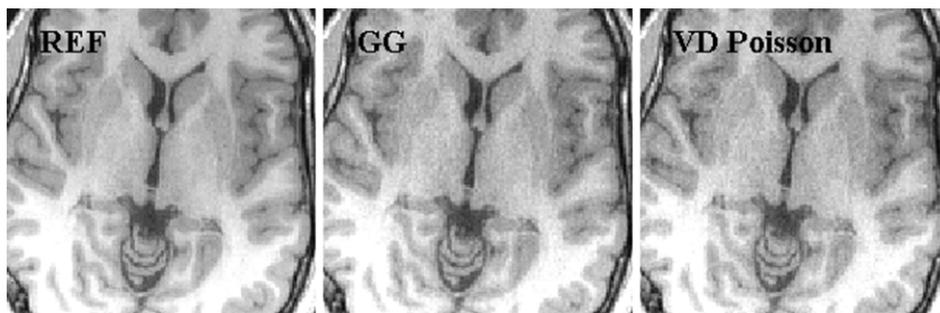


(b)

Figure 4.16. Comparison of SAKE reconstruction: (a) NMSEs and (b) maximum correlations using VD Poisson disk sampling of Shin, μ -law, and the proposed GG masks as a function of the reduction factor R .

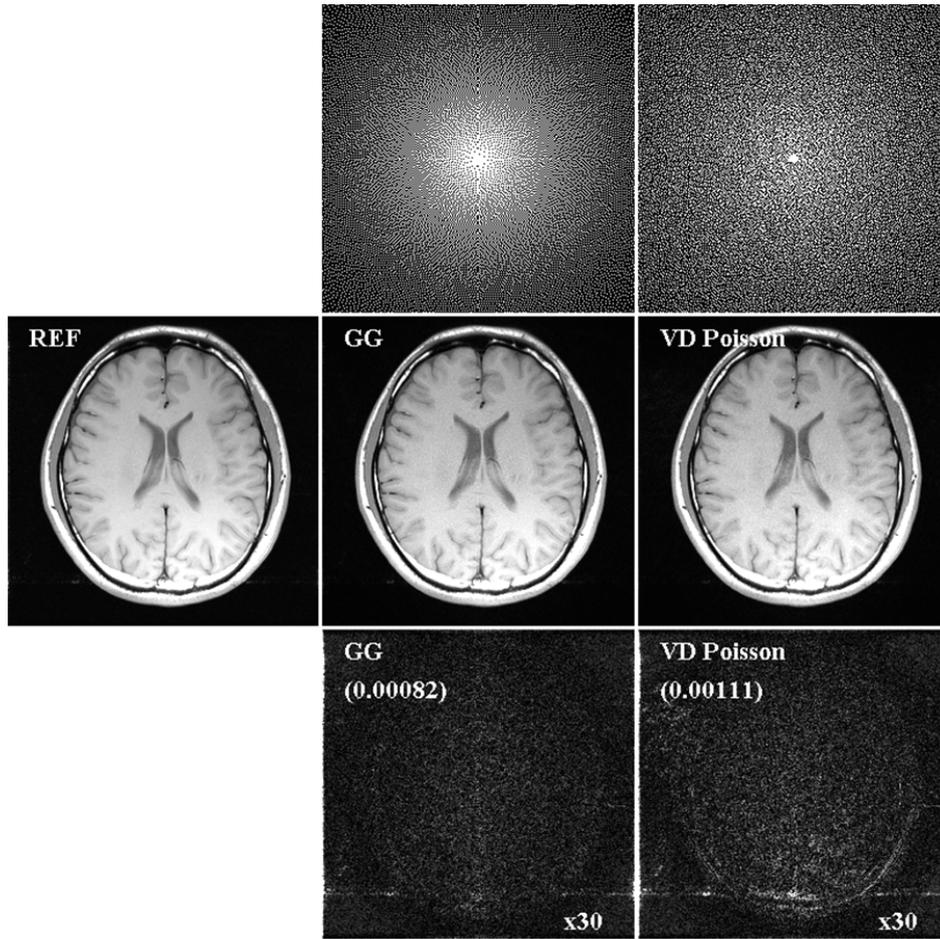


(a)

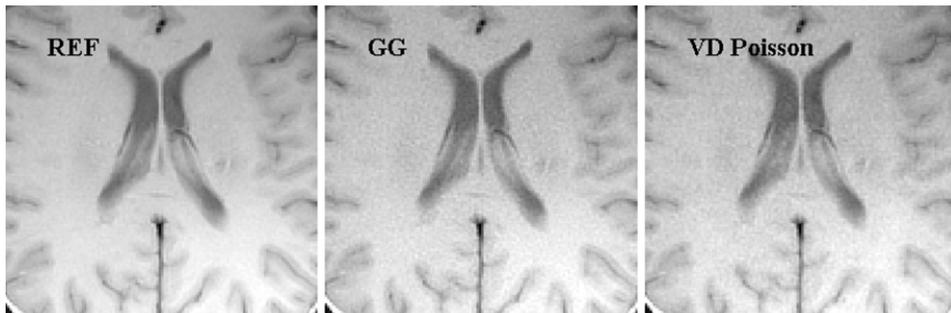


(b)

Figure 4.17 For the first data set at $R=3$, (a) sampling mask (top), reference and reconstructions (middle), and the corresponding error images (bottom) , (b) reconstructions in region of interest.



(a)



(b)

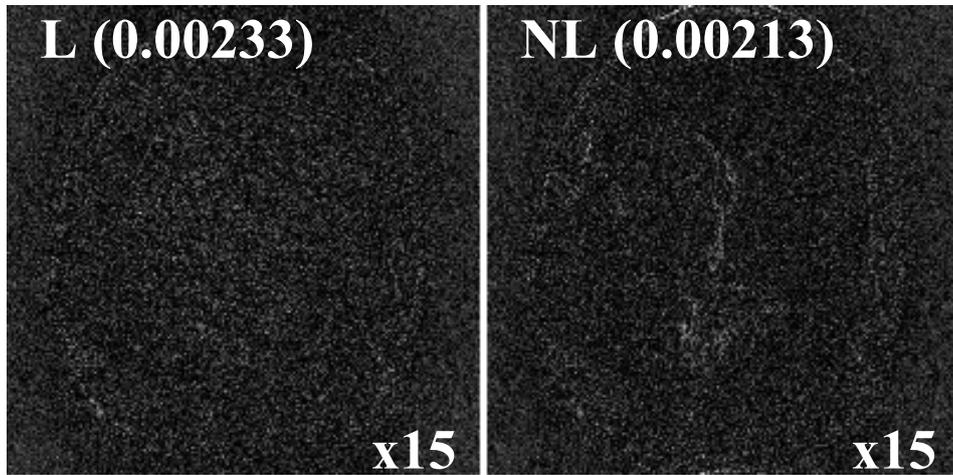
Figure 4.18 For the second data set at $R=3$, (a) sampling mask (top), reference and reconstructions (middle), and the corresponding error images (bottom), (b) reconstructions in region of interest.

4.3.2 Case of nonlinear SAKE

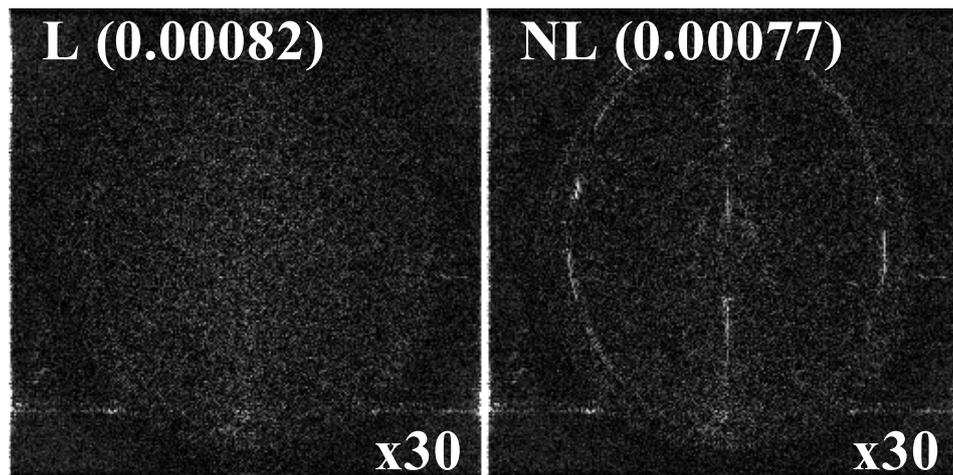
Figure 4.19 shows how nonlinear kernel has an effect on the performance of the reconstruction according to the reduction factor R . Using the nonlinear kernel reduces noise more effectively than the original method. However, the distribution of the noise is not uniformly random and the error image shows a distinctive pattern. For this reason, we will use the combination reconstruction method, which uses nonlinear kernel in the first third of entire iterations and then linear kernel in the later iterations, in order to reduce errors but remove the distinctive patterns. We choose the SAKE parameters as $w_{n\text{thresh}}=2.2$ for NL SAKE or combination of Linear and NL SAKES. In addition, we normalized k-space data for stabilization of nonlinear or combination SAKE so that all k-space data should be smaller than the maximum of sampled k-space data.

Figure 4.20(a) shows the normalized mean square errors (NMSEs) of various types of masks applied to SAKE according to $R=3.0$. The NMSE curve for each method shows the mean curve with minimum and maximum bounds for fifty masks. We see in the figure that the proposed random mask shows the best performance and least deviations all over the reduction factors. Figure 4.20(b) shows that the MCCs of our proposed masks are less than half of those of VD Poisson disk all over the reduction factors. The correlation coefficients for our proposed method are also shown to be less varied than those of VD Poisson disk. It means that our masks produce more incoherent noise against VD Poisson masks.

The experiments on the brain data [14] are shown in Figure 4.21. Our method using the proposed masks pattern reduces the reconstruction errors compared to the original method using the VD Poisson disks. Figure 4.22 shows the results for the second brain data. The NMSE using our sampling mask is below the mask used in the original SAKE paper. The reconstructed images of the brain using our method are more accurate than those using the VD Poisson disk.



(a)



(b)

Figure 4.19 Error images by L and NL SAKEs using proposed mask at $R=3$: (a) One for the first brain data and (b) one for the second brain data.

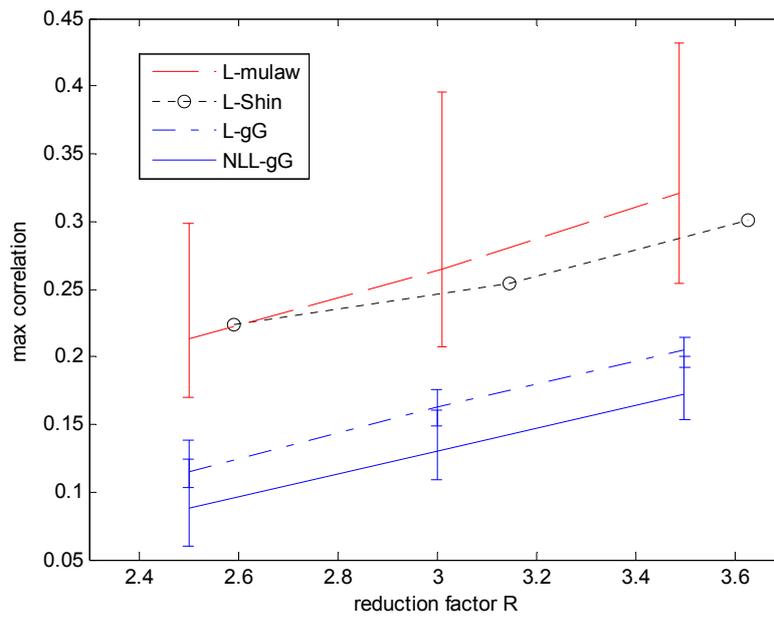
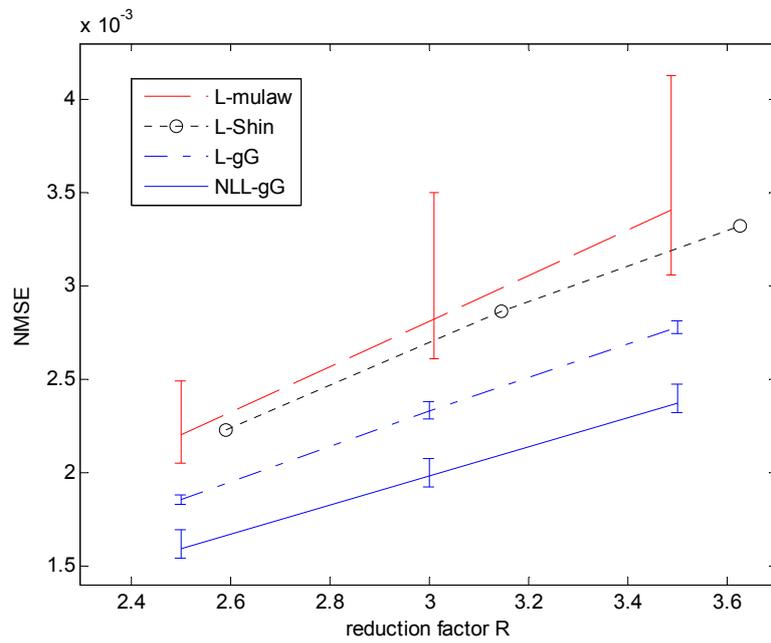
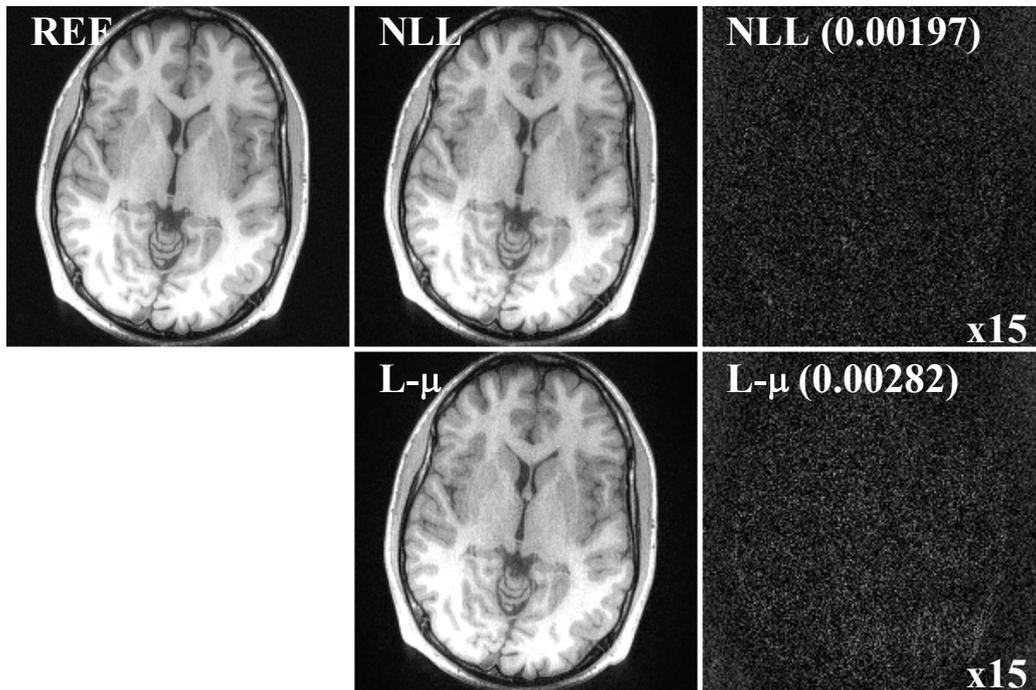
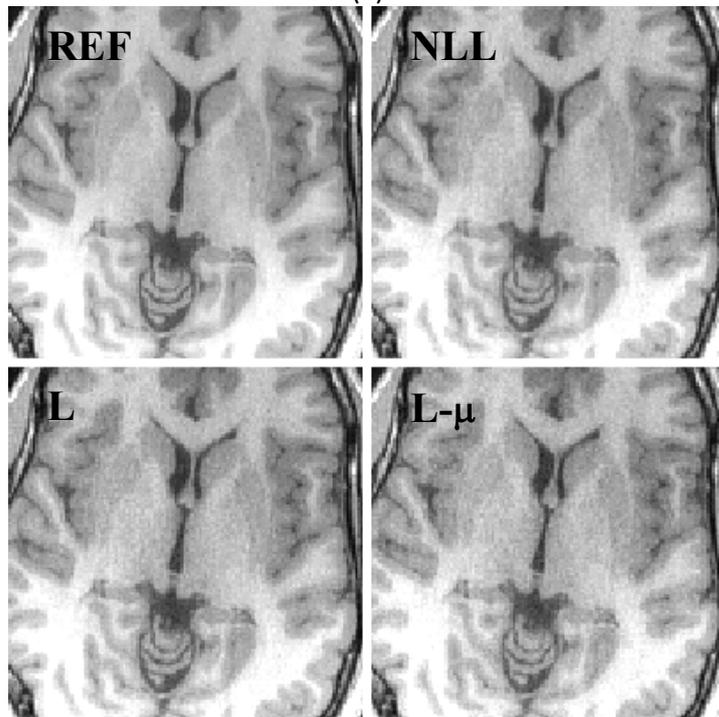


Figure 4.20 Performance comparison of SAKE reconstruction method using Shin masks and one using μ -law mask, one using proposed masks, and nonlinear SAKE on using proposed masks according to the reduction factor R: (a) NMSEs and (b) maximum correlations.



(a)



(b)

Figure 4.21 Reference image, reconstructed, magnified reconstruction, and error images by various SAKEs at $R=3$: (a) Reference image, reconstructed, and error images (b) magnified reconstructions.

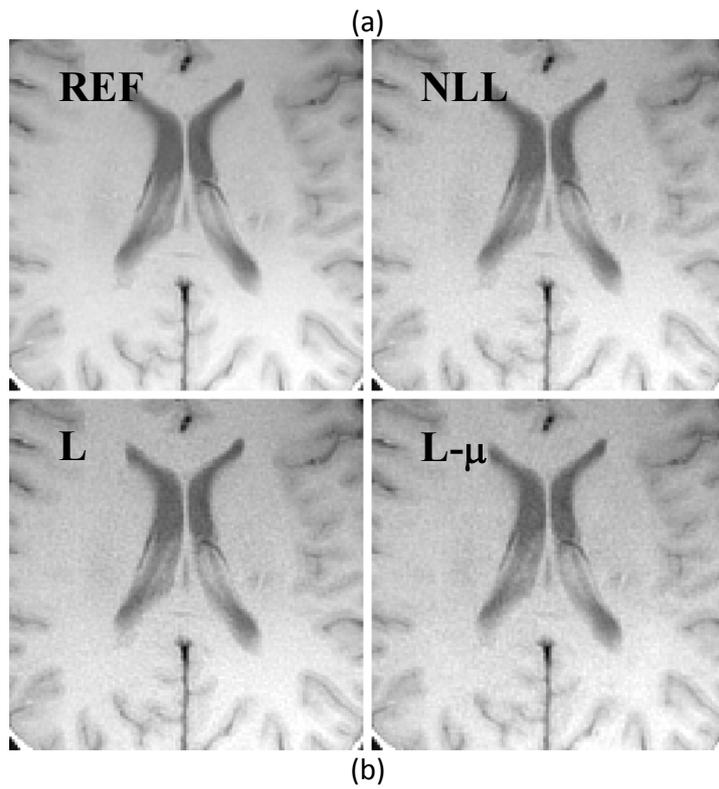
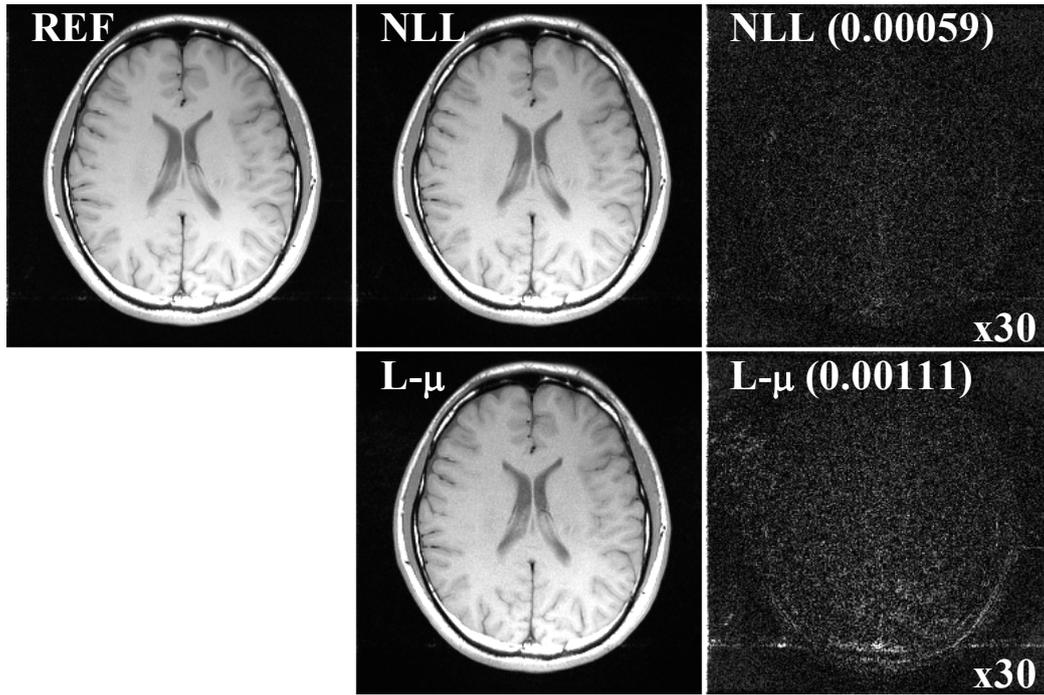


Figure 4.22 Reference image, reconstructed, magnified reconstruction, and error images by various SAKEs at $R=3$: (a) Reference image, reconstructed, and error images (b) magnified reconstructions.

4.4 Conclusion

We have proposed a new method to generate VD random sampling. Our masks have the following advantages over VD Poisson discs: Our method can always generate masks with a constant reduction factor, which means it is easy to control the number of samples with a parameter. Furthermore, our method is conceptually easy to implement using probability model and cost function satisfying the incoherence requirement. In addition, our proposed masks yield better image quality and lower NMSEs over VD Poisson disks. Future work will apply our proposed method to 3D MRI reconstruction or reconstruction of dynamic MR images in k-t domain.

Chapter 5

Conclusions and Future works

The work described in this thesis has been concerned with the development of image processing algorithm for MRI within filter design and probability. A number of interesting features of the proposed algorithm have been described and the method was shown to be effective in reconstruction of acquired data. These may be summarized as:

- Iterative GRAPPA using adaptive Wiener filter
- Mask sampling design using Generalized Gaussian probability and conflict cost function

5.1 Thesis Summary

A general introduction of the motivation and challenges of medical image processing was first presented in Chapter 1. Medical imaging processing plays an important role in reducing the MRI scanning time, the problem of extracting clinically useful information has become important from little amount of scanned data. For example, brain imaging helps to define the character and extent of brain disease, aiding diagnosis and treatment. To facilitate an effective and efficient analysis of this brain information, techniques for performing sampling or removing noise become a key challenge, among other image processing analysis such as visualization and quantitative comparison, which led to the main focus of this thesis:

A review of common techniques for MR imaging algorithms was given in Chapter 2.

In Chapter 3, we proposed an improved GRAPPA using adaptive Wiener filter. We tested it on brain images and get better noise reduction in reconstructing the data.

In Chapter 4, we proposed a new sampling pattern generation using generalized probability and conflict cost function. Changing many parameters to produce the variations

of our proposed pattern and applying different brain data, we showed how our method is effective.

5.2 Recommendations for Future Work

Although the results presented here have demonstrated the effectiveness of the image processing approach in MRI, it could be further developed in a number of ways:

Extending the mask applying to different domain

The proposed algorithm takes into account both and to describe features. Limitations on the existing model have been discussed in Chapter 4 and it was shown that such might, in some ways, not be well adapted to implement on 3D images or k-t domain to improve dynamic images. Perhaps by doing some experiments on other images, such as bones, and more data by including more random results than 50 cases, we could obtain more reliable results.

Using more advance image processing techniques

We discussed compressed sensing in Chapter 2. Nowadays, there are many solutions or reconstruction algorithm to compressed sensing such as total variation. Furthermore, we can change the domain into wavelet or curvelet domain to reconstruct the original image effectively. This would lead to a better optimization.

References

- [1] Donoho, David L. "Compressed sensing." *Information Theory, IEEE Transactions on* 52.4 (2006): 1289-1306.
- [2] Lustig, Michael, et al. "Compressed sensing MRI." *Signal Processing Magazine, IEEE* 25.2 (2008): 72-82.
- [3] Hashemi, Ray Hashman, William G. Bradley, and Christopher J. Lisanti. *MRI: the basics*. Lippincott Williams & Wilkins, 2012.
- [4] Carlos Fernandez Granda, "Nonlinear reconstruction in parallel magnetic resonance imaging" Master's thesis, the Technical University of Madrid, 2008
- [5] Pruessmann, Klaas P., et al. "SENSE: sensitivity encoding for fast MRI." *Magnetic resonance in medicine* 42.5 (1999): 952-962.
- [6] Griswold, Mark A., et al. "Generalized autocalibrating partially parallel acquisitions (GRAPPA)." *Magnetic Resonance in Medicine* 47.6 (2002): 1202-1210.
- [7] Ying, Leslie, and Jinhua Sheng. "Joint image reconstruction and sensitivity estimation in SENSE (JSENSE)." *Magnetic Resonance in Medicine* 57.6 (2007): 1196-1202.
- [8] Blaimer, Martin, et al. "SMASH, SENSE, PILS, GRAPPA: how to choose the optimal method." *Topics in Magnetic Resonance Imaging* 15.4 (2004): 223-236.
- [9] Wang, Haifeng, et al. "Improving GRAPPA using cross-sampled autocalibration data." *Magnetic Resonance in Medicine* 67.4 (2012): 1042-1053.
- [10] Wang, Ze, Jiongjiong Wang, and John A. Detre. "Improved data reconstruction method for GRAPPA." *Magnetic resonance in medicine* 54.3 (2005): 738-742.
- [11] Roemer, P. B., et al. "The NMR phased array." *Magnetic resonance in medicine* 16.2 (1990): 192-225.

- [12] Qu, Peng, Chunsheng Wang, and Gary X. Shen. "Discrepancy based adaptive regularization for GRAPPA reconstruction." *Journal of Magnetic Resonance Imaging* 24.1 (2006): 248-255.
- [13] Zhao, Tiejun, and Xiaoping Hu. "Iterative GRAPPA (iGRAPPA) for improved parallel imaging reconstruction." *Magnetic Resonance in Medicine* 59.4 (2008): 903-907.
- [14] Chang, Yuchou, Dong Liang, and Leslie Ying. "Nonlinear GRAPPA: A kernel approach to parallel MRI reconstruction." *Magnetic Resonance in Medicine* 68.3 (2012): 730-740.
- [15] Breuer, Felix A., et al. "Dynamic autocalibrated parallel imaging using temporal GRAPPA (TGRAPPA)." *Magnetic Resonance in Medicine* 53.4 (2005): 981-985.
- [16] Huang, Feng. "Technique for parallel MRI imaging (kt grappa)." U.S. Patent Application 11/221,334.
- [17] Liang, Dong, Bo Liu, and Leslie Ying. "Accelerating sensitivity encoding using compressed sensing." *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE. IEEE, 2008.*
- [18] Candès, Emmanuel J., and Michael B. Wakin. "An introduction to compressive sampling." *Signal Processing Magazine, IEEE* 25.2 (2008): 21-30.
- [19] Candes, Emmanuel, and Justin Romberg. "Sparsity and incoherence in compressive sampling." *Inverse problems* 23.3 (2007): 969.
- [20] Jung, Hong, et al. "k-t FOCUSS: A general compressed sensing framework for high resolution dynamic MRI." *Magnetic Resonance in Medicine* 61.1 (2009): 103-116.
- [21] Liang, Dong, et al. "k-t ISD: Dynamic cardiac MR imaging using compressed sensing with iterative support detection." *Magnetic Resonance in Medicine* 68.1 (2012): 41-53.
- [22] Lustig, M., and J. M. Pauly. "Iterative GRAPPA: a general solution for the GRAPPA reconstruction from arbitrary k-space sampling." *Proc. ISMRM 15th Scientific Meeting. 2007.*

- [23] Park, Suhyung, and Jaeseok Park. "Adaptive self-calibrating iterative GRAPPA reconstruction." *Magnetic Resonance in Medicine* 67.6 (2012): 1721-1729.
- [24] Lustig, Michael, and John M. Pauly. "SPIRiT: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space." *Magnetic Resonance in Medicine* 64.2 (2010): 457-471.
- [25] Pruessmann, Klaas P. "Encoding and reconstruction in parallel MRI." *NMR in Biomedicine* 19.3 (2006): 288-299.
- [26] R.E. Walpole and R. H. Myers, *Probability and Statistics for Engineers and Scientists*, 5th Ed., Macmillan Publishing Co., 1993
- [27] J.S. Lim, *Two-dimensional Signal and Image Processing*, Prentice-Hall, 1990.
- [28] P.J. Shin, P.E.Z. Larson, M. A. Ohliger, M. Elad, J.M. Pauly, D.B. Vigneron, M. Lustig, "Calibrationless parallel imaging reconstruction based on structured low-rank matrix completion," *Magn Reson Med.* online 2013.
- [29] V.K. Ingle and J.G. Proakis, *Essentials of Digital Signal Processing using MATLAB*, 3rd, CENGAGE Learning, 2011.
- [30] Lustig, Michael, David Donoho, and John M. Pauly. "Sparse MRI: The application of compressed sensing for rapid MR imaging." *Magnetic resonance in medicine* 58.6 (2007): 1182-1195.
- [31] Nayak, KRISHNA S., and Dwight G. Nishimura. "Randomized trajectories for reduced aliasing artifact." *Proceedings of the 6th Annual Meeting of ISMRM, Sydney, Australia.* 1998.
- [32] Lustig, M., et al. "L1 SPIR-iT: Autocalibrating parallel imaging compressed sensing." *Proc Intl Soc Mag Reson Med.* Vol. 17. 2009.
- [33] Vasanawala, S. S., et al. "Practical parallel imaging compressed sensing MRI: Summary of two years of experience in accelerating body MRI of pediatric patients." *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on.* IEEE, 2011.

- [34] Ying, Xiang, et al. "An intrinsic algorithm for parallel Poisson disk sampling on arbitrary surfaces." *Visualization and Computer Graphics, IEEE Transactions on* 19.9 (2013): 1425-1437.
- [35] Knoll, Florian, et al. "Adapted random sampling patterns for accelerated MRI." *Magnetic resonance materials in physics, biology and medicine* 24.1 (2011): 43-50.
- [36] M.N. Do and M.Vetterli, "Wavelet-based texture retrieval using generalized Gaussian density and Kullback–Leibler distance," *IEEE Trans. Image Processing*, pp. 146-158, vol. 11, no. 2, Feb. 2002.
- [37] W. Kim, J. Lyu, Y. Zhou and L. Ying, "Conflict-cost based random sampling design for parallel MRI with low rank constraints," *Proc. SPIE*, vol. 9484, Baltimore, Apr. 2015.
- [38] N.S. Jayant and P. Noll, *Digital Coding of Waveforms – Principles and Applications to Speech and Video*, Prentice-Hall, 1984.
- [39] D. Dunbar and G. Humphreys, "A spatial data structure for fast Poisson-disk sample generation," presented at the SIGGRAPH, Boston, MA, Jul.-Aug. 30–3, 2006.
- [40] Candès, Emmanuel J., and Michael B. Wakin. "An introduction to compressive sampling." *Signal Processing Magazine, IEEE* 25.2 (2008): 21-30.
- [41] W. Kim and Y. Zhou, "Iterative GRAPPA using Wiener filter," *Proc. ISMRM*, Toronto, June 2015.