Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

Image reconstruction in low-field MRI: A super-resolution approach

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

MEREL LISANNE DE LEEUW DEN BOUTER

Delft, the Netherlands
JUNE 2017

MSc THESIS APPLIED MATHEMATICS

"Image reconstruction in low-field MRI: A super-resolution approach"

MEREL LISANNE DE LEEUW DEN BOUTER

Delft University of Technology

Daily Supervisor

Dr. ir. M.B. van Gijzen

Other thesis committee members

Dr. ir. R.F. Remis

Prof. dr. ir. G. Jongbloed

Prof. dr. A. Webb

Prof. dr. ir. S. Schiff

June, 2017

Delft

Summary

The quality of magnetic resonance images produced by conventional MRI scanners is guaranteed by the strength and homogeneity of the magnetic field. However, the superconducting magnets required to produce such a field make MRI scanners large and expensive and hence inaccessible to a large number of people in developing countries. Our partners are developing low-cost, portable MRI scanners that do not depend on superconducting magnets. In these scanners, the signal-to-noise ratio will be significantly lower due to the lower magnetic field strength. Additionally, inhomogeneities will be present, which means that the traditional way of obtaining the image, by inverse Fourier Transform, is no longer feasible. In this research, image reconstruction is done using an ill-posed system of equations of the form $A\mathbf{x} = \mathbf{y}$, where A is the reconstruction matrix and $\mathbf{x}$ and $\mathbf{y}$ are vectors containing the image pixel values and the measured signals, respectively. Three different regularization techniques are considered, with total variation yielding the best results. Two methods for solving the regularized least-squares problem are considered: CGLS and CGNE. For the types of problems we are dealing with, CGNE is outperformed by CGLS: CGLS requires a lower number of iterations to converge and the computational cost per iteration is lower. The main focus of this research is on super-resolution: reconstructing a high resolution image from one or several low resolution images. Due to the low signal-to-noise ratios that are expected in the low-field MRI prototypes, it might be better to reconstruct images of a low resolution, and using these, create high resolution images, instead of opting for a direct high resolution reconstruction. In order to test this, the signal generation in a Halbach array based MRI scanner is simulated. Our simulations show that for very low ($< 1.5 - 2$) signal-to-noise ratios, super-resolution can yield better results than direct high resolution reconstruction. Data obtained in a 7 T MRI scanner is used to validate our reconstruction model. Due to the type of gradient used and the low number of measurements in this experiment, the amount of available information is very limited. This makes it challenging to produce an image of good quality. However, in our final image, out of the four water bottles in the phantom, the three largest ones are clearly visible.

Preface

Nine months ago I started this research in order to fulfill the final requirements of my Applied Mathematics Master's program. It was carried out in the Numerical Linear Algebra research group of the Delft University of Technology and in the MRI research group at the Leiden University Medical Center. While, at times, the research was stressful and frustrating, the journey has been amazing and it has brought me great things. The most important one is a job for the next four years in the form of a PhD project that, if successful, might save children's lives. I have always known that I want to use my analytical skill set to try to bring about positive change in the world, but I had no idea how. So I am very grateful that this project found its way to me in the form of Martin van Gijzen.

Martin, thank you for being a great daily supervisor. Your enthusiasm for this project is very contagious and made working on it even better. I would like to thank my other supervisors, Rob Remis and Andrew Webb, for their guidance and their willingness to answer my questions. Thank you, Steven Schiff, for allowing me to spend two weeks in your research group at Pennsylvania State University.

A big shout-out to the people I shared an office with for the duration of this project: Deborah, Hassan, Hrishikesh and Vin. We became a family over the past months and that made coming in to work every day a real joy.

Last but not least, I would like to thank my parents for their support and constant belief in me.

Merel Lisanne de Leeuw den Bouter
Delft, June 2017

Contents

Chapter 1

Introduction

Magnetic resonance imaging (MRI) is a powerful technique that reveals the internal structure and function of the human body in a non-invasive way. In clinical MRI scanners, superconducting magnets are used to generate very high magnetic fields. The size and strict infrastructure demands of these magnets make it impossible to use them in many environments. Additionally, these MRI scanners are costly to purchase, site and maintain [1]. Due to these limitations, many people in developing countries do not have access to MRI technology.

Teams at the Leiden University Medical Center (LUMC) and at Pennsylvania State University (PSU) are working on developing low-cost, portable MRI scanners with a magnetic field in the milliTesla range, aiming to diagnose children in developing countries with hydrocephalus ('water on the brain'), a debilitating disease that is usually left untreated in the third world. The PSU prototype is shown in Figure 1.1.

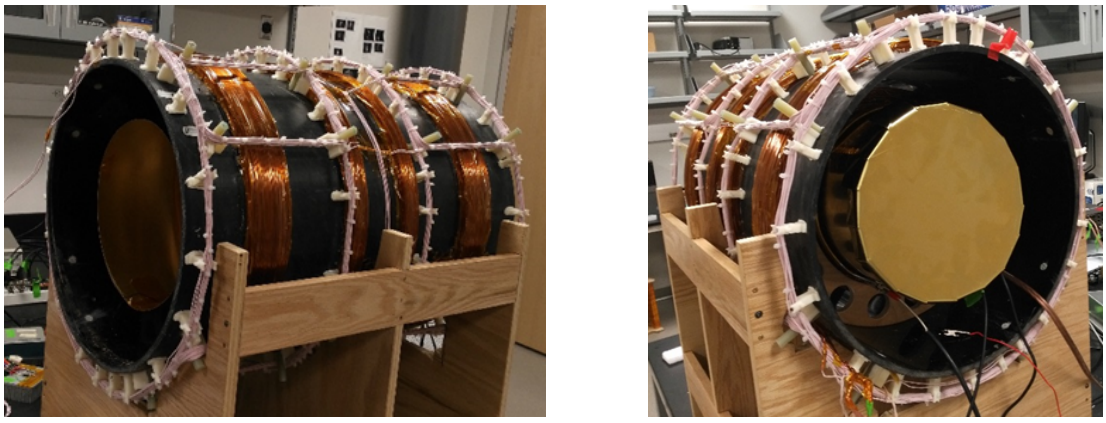
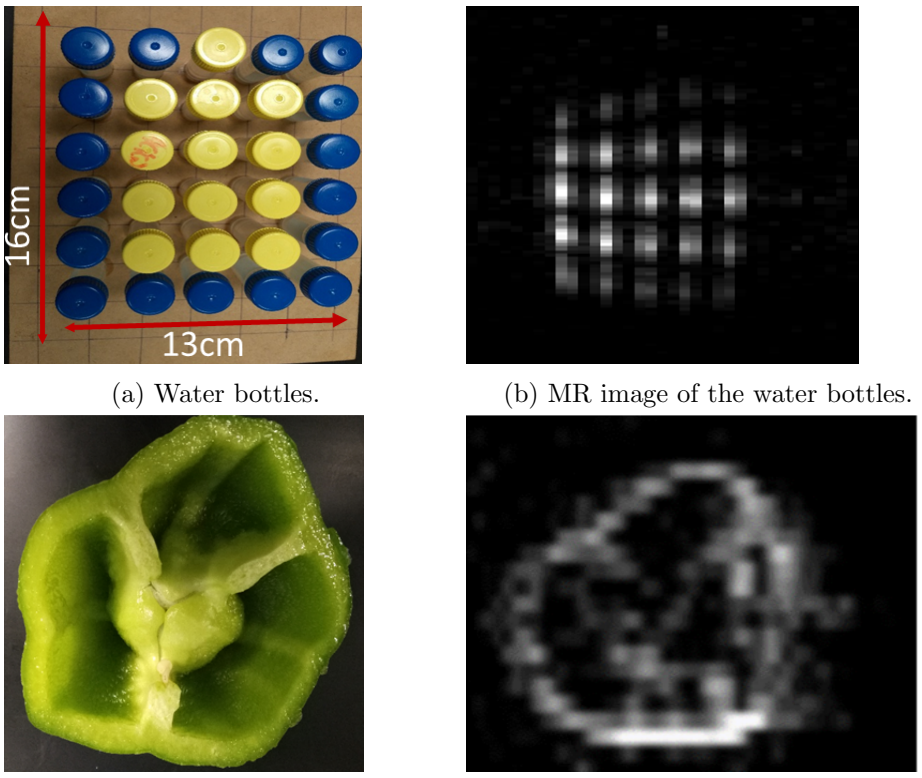


Figure 1.1: The PSU low-field MRI scanner prototype.

Some images produced using this prototype are shown in Figure 1.2. The main features of the imaged objects can be discerned, but in order to use this device in a clinical setting, clearer and more detailed images are required. The focus of this research is to improve the quality of images generated by low-field MRI scanners using super-resolution reconstruction (SRR). SRR enables us to use several low resolution images in order to produce one image with a higher resolution. The first super-resolution algorithms were introduced by Gerchberg [2] in 1974 and De Santis and Gori [3] in 1975. The application of super-resolution reconstruction to MRI was first mentioned in [4] in 1997. Since then, a large number of articles adapting the super-resolution concept from video-processing to MRI data have been reported and encouraging results have been demonstrated [5]. The approach taken in this research leads to a system of equations of

the form $A\mathbf{x} = \mathbf{y}$, where $\mathbf{x}$ is the unknown quantity. Due to the ill-posedness of this problem, regularization is needed. Another issue is that the higher the desired resolution of the target image, the more pixels, and the larger the matrix. Therefore, for high resolution problems, matrix inversion is computationally infeasible. Iterative methods have to be employed to get to the solution. The methods of choice are conjugate gradient methods.



(a) Water bottles.

(b) MR image of the water bottles.

(c) Bell pepper.

(d) MR image of the bell pepper.

Figure 1.2: The prototype was used to create MR images of an array of water bottles and a bell pepper.

1.1 Research goal

The main goal of this project is to research whether super-resolution reconstruction can be used to improve the quality of the images produced using low-field MRI to such an extent that this method can be used to detect hydrocephalus.

The following research questions were formulated in order to structure this research:

1. Can super-resolution reconstruction yield high resolution MR images when applied to simulated data?

2. Can super-resolution reconstruction yield high resolution MR images when applied to real data?

3. How should we formulate the matrix A in Equation (4.5), describing the transition from a high resolution object to low resolution images?

4. Which method(s) should be used to solve the minimization problem arising during the image reconstruction process?

5. Which type of regularization yields the best results?

1.2 Thesis structure

The structure of this report is as follows. Chapter 2 describes conventional MRI: the hardware components, how signals are generated and how they are used to generate an image. In chapter 3, the MRI scanners at LUMC and PSU are discussed and a more general model describing the signal is introduced. Chapter 4 describes the super-resolution reconstruction technique that will be used in this research. In chapter 5, the super-resolution reconstruction and the direct reconstruction models will be cast in the form of a minimization problem with different types of regularization. Chapter 6 contains a description of the conjugate gradient methods that will be used to solve the least-squares problem resulting from the super-resolution reconstruction model. A comparison of the performance of two different conjugate gradient methods (CGLS and CGNE) is given in chapter 7. Chapter 8 contains the results that were obtained when the super-resolution framework was applied to simulated data. In chapter 9, an MR image is formed out of measurements carried out in an inhomogeneous magnetic field. Chapter 10 contains conclusions and suggestions for future research.

Chapter 2

Conventional MRI

This chapter is based on [6]. All images were taken from [6] as well.

2.1 Hardware components

There are three main components in an MR scanner: a main magnet, a magnetic field gradient system and an RF system.

2.1.1 The main magnet

The purpose of the main magnet is to generate a strong and uniform static magnetic field, which is referred to as the B_0 field. In clinical MR scanners, superconducting magnets are used to produce magnetic field strengths between 1.5 and 3 T. However, this research focuses on low-field MRI, in which magnetic fields of less than 0.15 T are generated by a resistive magnet. Resistive magnets can be used at room-temperature, as opposed to superconducting magnets that have to be cooled with liquid helium. This makes low-field MRI less costly. However, one of the main issues with low-field MRI is the lower signal-to-noise ratio.

2.1.2 The magnetic field gradient system

Three orthogonal gradient coils make up the magnetic field gradient system. A schematic representation of two orthogonal gradient coils is shown in Figure 2.1. Gradient coils are used to generate time-varying magnetic fields which vary linearly in each of the spatial dimensions. One of the main specifications of the gradient system is the gradient strength. The higher the gradient strength, the better. Another important feature is the rate at which the maximum gradient strength can be obtained, known as the rise time. Better gradient systems tend to have shorter rise times. In most clinical imaging systems, the maximum gradient strength that can be attained is 10 mT/m and rise times of approximately 1 ms from 0 to 10 mT/m are considered good.

2.1.3 The RF system

The radio frequency (RF) system consists of two components: a transmitter coil and a receiver coil. The transmitter coil generates a rotating magnetic field, called the B_1 field, which excites the electron spins in the object that is to be imaged. The precessing magnetization is converted into an electrical signal by the receiver coil. Both coils are called RF coils because they resonate at radio frequency. The RF system is required to generate a uniform B_1 field and to have a high detection sensitivity.

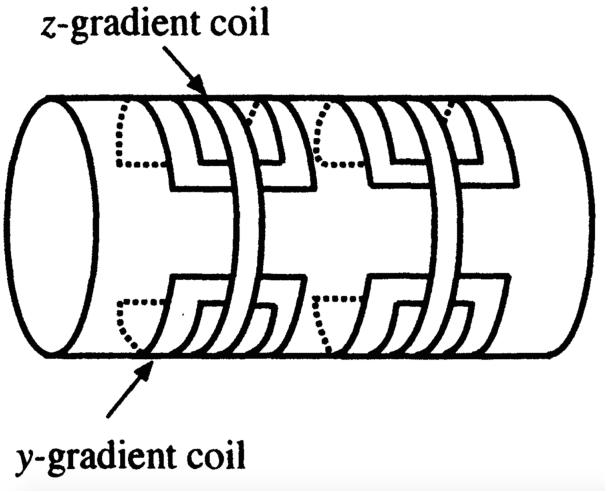


Figure 2.1: Schematic representation of the y-coil and the z-coil that are used to generate the y- and z-gradient.

2.2 Signal generation and detection

2.2.1 Spin

Any physical object consists of atoms, which in turn consist of nuclei and the electrons orbiting around them. Nuclei with odd atomic weights, such as the hydrogen atom, have an angular momentum $\vec{J}$ called spin. Spin can be visualized as a physical rotation, similar to the rotation of a top around its own axis. Because nuclei are electrically charged, each nuclear spin generates its own microscopic magnetic field. This can be represented by the magnetic moment $\vec{\mu}$. $\vec{J}$ and $\vec{\mu}$ are related to each other in the following way:

$$\vec{\mu} = \gamma \vec{J}, \tag{2.1}$$

where γ is the gyromagnetic ratio, the value of which depends on the nucleus in question. For hydrogen atoms, $\gamma = 2.675 \cdot 10^8$ rad(sT)$^{-1}$. In the absence of an external magnetic field, the direction of μ is random, as can be seen in Figure 2.2a. To generate a net magnetic field from the object to be imaged, the spin vectors have to be aligned. This can be done by applying a strong external magnetic field. This field is assumed to be applied in the z-direction:

$$\vec{B} = B_0 \vec{k}, \tag{2.2}$$

where $\vec{k}$ is the unit vector in the z-direction. For a spin-$\frac{1}{2}$ system, the spin will align either parallel (spin-up) or antiparallel (spin-down) to the magnetic field, as shown in Figure 2.2b. The angular frequency of nuclear precession is described by the Larmor equation

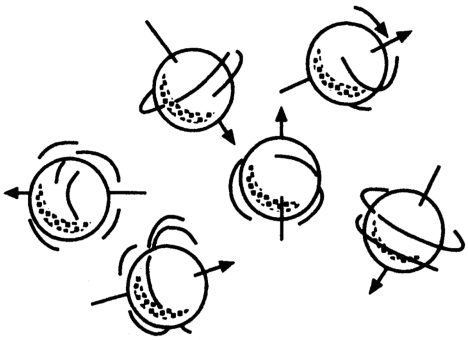
$$\omega_0 = \gamma B_0. \tag{2.3}$$

ω_0 is known as the Larmor frequency.

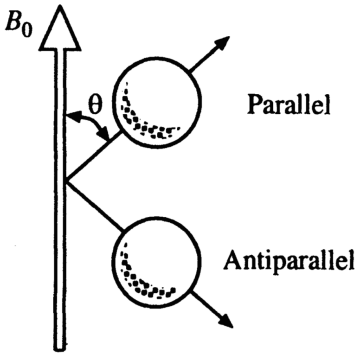
2.2.2 Bulk magnetization

According to quantum theory, spin-up states have an energy of $E_\uparrow = -\frac{1}{2}\gamma\hbar B_0$ while spin-down states have an energy of $E_\downarrow = \frac{1}{2}\gamma\hbar B_0$, where $\hbar = 1.055 \cdot 10^{-34}$ m^{2}kg/s is the reduced

Planck constant. So a parallel alignment constitutes a lower energy state than an antiparallel alignment. Therefore, spins will be more likely to assume the parallel alignment, yielding an observable macroscopic magnetization $\vec{M}$, which is pointed in the same direction as $\vec{B}_0$ and whose magnitude is directly proportional to B_0.



(a) When no external magnetic field is present, the nuclear magnetic moments point in random directions.



(b) In the presence of an external magnetic field, the nuclear magnetic moments align themselves parallel or antiparallel to $\vec{B}_0$.

Figure 2.2: Nuclear magnetic moments in the absence and presence of an external magnetic field.

2.2.3 RF excitations

Planck's law dictates that the energy carried by electromagnetic radiation with frequency ω_{rf} is

$$E_{rf} = \hbar\omega_{rf}. \tag{2.4}$$

In order to make the spins transition from one energy state to another, E_{rf} must be equal to the energy difference between the spin states. Therefore, the following must hold:

$$\hbar\omega_{rf} = E_{\downarrow} - E_{\uparrow} = \gamma\hbar B_0 = \hbar\omega_0, \tag{2.5}$$

which leads to

$$\omega_{rf} = \omega_0. \tag{2.6}$$

Equation (2.6) is the resonance condition.

B_1 field The B_1 field, also known as RF pulse, is a short-lived magnetic field that oscillates in the radio-frequency range. The B_1-field is much weaker than the B_0-field. The B_1-field typically has the following form:

$$\vec{B}_1(t) = 2B_1^e(t)\cos(\omega_{rf}t + \phi)\vec{i}, \tag{2.7}$$

where $B_1^e(t)$ is a the pulse envelope function, ω_{rf} is the excitation carrier frequency and ϕ is the initial phase angle. Two popular envelope functions are the rectangular pulse and the sinc pulse. The B_1-field is linearly polarized because the oscillations occur linearly along the x-axis. Equation (2.7) can be rewritten as

$$\begin{aligned}\vec{B}_1(t) = B_1^e(t)[\cos(\omega_{rf}t + \phi)\vec{i} - \sin(\omega_{rf}t + \phi)\vec{j}] + \\ B_1^e(t)[\cos(\omega_{rf}t + \phi)\vec{i} + \sin(\omega_{rf}t + \phi)\vec{j}].\end{aligned} \tag{2.8}$$

In (2.8), the first bracketed term rotates clockwise, while the second rotates counterclockwise. The spins rotate clockwise, which means that if the B_1-field has a frequency near the Larmor frequency, the effect of the counterclockwise component is negligible. So the effective B_1-field is described by

$$\vec{B}_1(t) = B_1^e(t)[\cos(\omega_{rf}t + \phi)\vec{i} - \sin(\omega_{rf}t + \phi)\vec{j}] \tag{2.9}$$

with an x-component

$$B_{1,x} = B_1^e(t)\cos(\omega_{rf}t + \phi) \tag{2.10}$$

and a y-component

$$B_{1,y} = -B_1^e(t)\sin(\omega_{rf}t + \phi). \tag{2.11}$$

For brevity, $B_1(t)$ can be described using complex notation:

$$B_1(t) = B_{1,x} + iB_{1,y} = B_1^e(t)e^{-i(\omega_{rf}t+\phi)}. \tag{2.12}$$

The initial phase ϕ will be assumed to be 0 from now on.

The Bloch equation The behavior of the net magnetization $\vec{M}$ is governed by the Bloch equation, which, in the context of MRI, has the following form:

$$\frac{d\vec{M}}{dt} = \gamma\vec{M} \times \vec{B} - \frac{M_x\vec{i} + M_y\vec{j}}{T_2} - \frac{(M_z - M_z^{eq})\vec{k}}{T_1}, \tag{2.13}$$

where M_z^{eq} is the thermal equilibrium value for $\vec{M}$ in the presence of $\vec{B}_0$ only. T_1 and T_2 are time constants that characterize the relaxation process of a spin system after it has been excited. In order to describe the excitation effect of an RF pulse, a reference frame that rotates with an angular frequency ω is introduced. In this case, $\omega = \omega_0$, the Larmor frequency. The axes are denoted by x', y' and z' and the unit vectors are $\vec{i}'$, $\vec{j}'$ and $\vec{k}'$. The following transformation relates the rotating frame to the stationary frame:

$$\begin{cases} \vec{i}' = \cos(\omega t)\vec{i} - \sin\omega t\vec{j} \\ \vec{j}' = \sin(\omega t)\vec{i} - \cos\omega t\vec{j} \\ \vec{k}' = \vec{k} \end{cases} \tag{2.14}$$

It can be shown that in this rotating frame, the Bloch equation is given by

$$\frac{d\vec{M}_{rot}}{dt} = \gamma\vec{M}_{rot} \times \vec{B}_{eff} - \frac{M_{x'}\vec{i}' + M_{y'}\vec{j}'}{T_2} - \frac{(M_z - M_z^{eq})\vec{k}'}{T_1}, \tag{2.15}$$

where $\vec{B}_{eff}$, the effective magnetic field that the $\vec{M}$ experiences, is given by

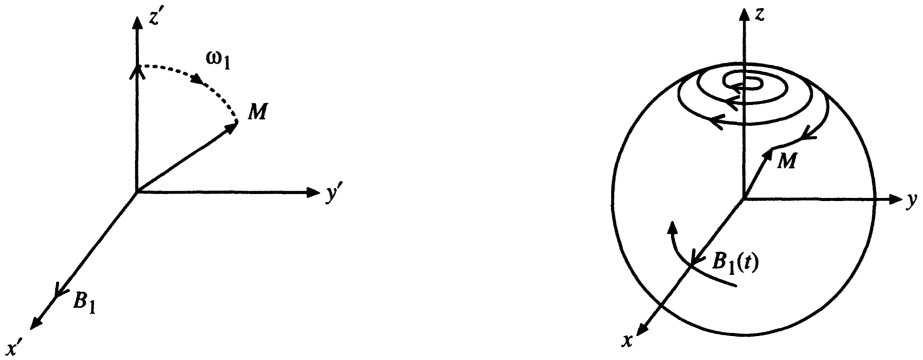
$$\vec{B}_{eff} = \vec{B}_{rot} + \frac{\vec{\omega}}{\gamma}. \tag{2.16}$$

2.2.4 Relaxation

From the Bloch equation it can be shown that the bulk magnetization vector $\vec{M}$ precesses about the x'-axis with angular velocity

$$\vec{\omega}_1 = -\gamma \vec{B}_1, \tag{2.17}$$

as shown in Figure 2.3a.



(a) Rotating frame. (b) Stationary frame.

Figure 2.3: Motion of $\vec{M}$ in the presence of a B_1-field.

The precession of $\vec{M}$ about the B_1-field is called forced precession, as opposed to the precession of $\vec{M}$ about B_0, which is known as free precession. Forced precession causes the bulk magnetization to tip away from the z'-axis, yielding a measurable transverse component $\vec{M}_{x'y'}$. After the RF pulse has been applied, $\vec{M}$ will return to precessing around the z'-axis. During this return, two relaxation processes occur: longitudinal relaxation and transverse relaxation. Longitudinal relaxation is the recovery of M_z, while transverse relaxation is the destruction of the transverse magnetization M_{xy}. After the RF pulse, $\vec{B}_{eff} = 0$. Therefore, the solution to the Bloch equation looks as follows:

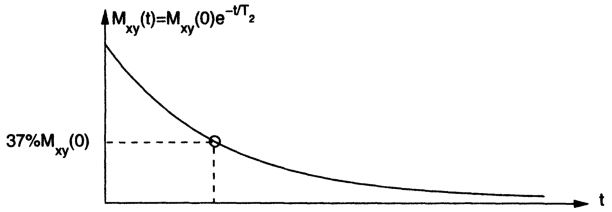
$$\begin{cases} M_{x'y'}(t) = M_{x'y'}(0_+)e^{-t/T_2} \\ M_{z'}(t) = M_z^{eq}\left(1 - e^{-t/T_1}\right) + M_{z'}(0_+)e^{-t/T_1} \end{cases} \tag{2.18}$$

where $M_{x'y'}(0^+)$ and $M_{z'}(0^+)$ are the magnetizations in the transverse plane and along the z'-axis, respectively, immediately after the RF pulse. If all magnetization has been tipped into the transverse plane, $M_{z'}(0^+) = 0$. Figure 2.4 shows plots of $M_{x'y'}$ and $M_{z'}$. $M_{x'y'}(t)$ is characterized by T_2, while $M_{z'}$ is characterized by T_1.

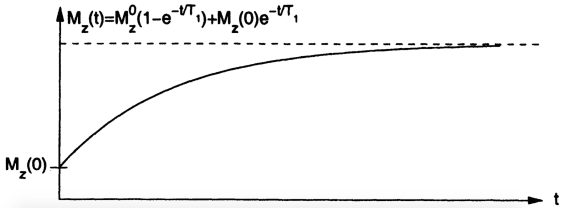
Transforming Equations (2.18) back to the laboratory system yields

$$\begin{cases} M_{xy}(t) = M_{xy}(0_+)e^{-t/T_2}e^{-i\omega_0 t} \\ M_z(t) = M_z^{eq}\left(1 - e^{-t/T_1}\right) + M_z(0_+)e^{-t/T_1} \end{cases} \tag{2.19}$$

where $M_{xy}(0_+) = M_{x'y'}(0_+)e^{-i\omega_0\tau_p}$, with τ_p the duration of the RF pulse. The trajectory of $\vec{M}$ can be seen in Figure 2.5. It should be noted that while $\vec{M}$ approaches the z-axis, its magnitude is not preserved, due to the relaxation process. This is different from the behavior of $\vec{M}$ as it moves away from the z-axis during the excitation period.



(a) $M_{x'y'}$ after an RF pulse as a function of time.



(b) M'_z after an RF pulse as a function of time.

Figure 2.4: Relaxation curves.

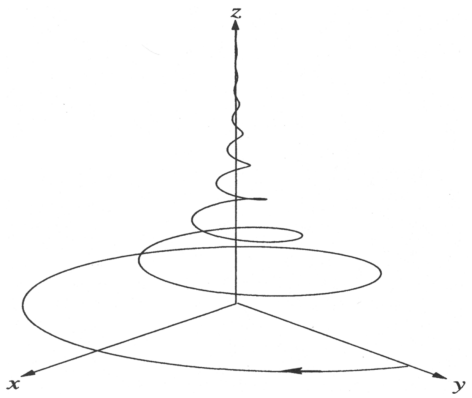


Figure 2.5: The trajectory of $\vec{M}$ in the laboratory frame during the relaxation period.

2.2.5 Signal detection

By Faraday's law of induction, the voltage induced in the receiver coil is described by

$$V(t) = -\frac{\partial}{\partial t} \int_{\text{object}} \vec{B}_r(\mathbf{r}) \cdot \vec{M}(\mathbf{r}, t)\ d\mathbf{r}, \tag{2.20}$$

where $\vec{B}_r(\mathbf{r})$ is referred to as the receive field, since it can be interpreted as the magnetic flux density generated by the receive coil carrying a unit current. Note that $\vec{B}_r(\mathbf{r})$ is a weighting vector. After some manipulations, Equation (2.20) can be written as

$$V(t) = \int_{\text{object}} \omega(\mathbf{r}) |\vec{B}_{r,xy}(\mathbf{r})| |M_{xy}(\mathbf{r}, 0)| e^{-t/T_2(\mathbf{r})} \cos\left[-\omega(\mathbf{r})t + \phi_e(\mathbf{r}) - \phi_r(\mathbf{r}) + \frac{\pi}{2}\right] d\mathbf{r}, \tag{2.21}$$

where $\phi_e(\mathbf{r})$ is the initial phase shift induced by RF excitation and $\phi_r(\mathbf{r})$ is the phase of the reception field. Because the Larmor frequency is high, the voltage $V(t)$ is a high-frequency signal, which can cause unnecessary problems in later processing stages. Therefore, in practice, $V(t)$ is moved to a low-frequency band using a phase-sensitive detection method: $V(t)$ is multiplied

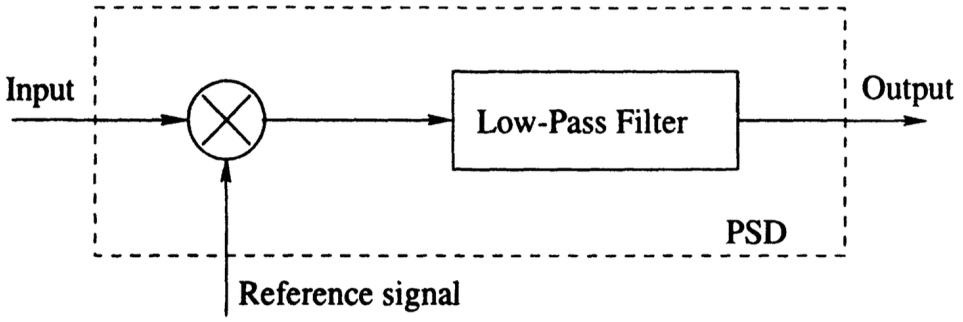
by the reference sinusoidal signal $2\cos(\omega_0 t)$ and then filtered using a low-pass-filter, effectively removing the high-frequency component. The result will be called $S_R(t)$. $S_R(t)$ can be shown to be described by the following expression:

$$S_R(t) = \int_{\text{object}} \omega(\mathbf{r})|\vec{B}_{r,xy}(\mathbf{r})||M_{xy}(\mathbf{r},0)|e^{-t/T_2(\mathbf{r})}\cos\left[-\Delta\omega(\mathbf{r})t + \phi_e(\mathbf{r}) - \phi_r(\mathbf{r}) + \frac{\pi}{2}\right] d\mathbf{r}, \quad (2.22)$$

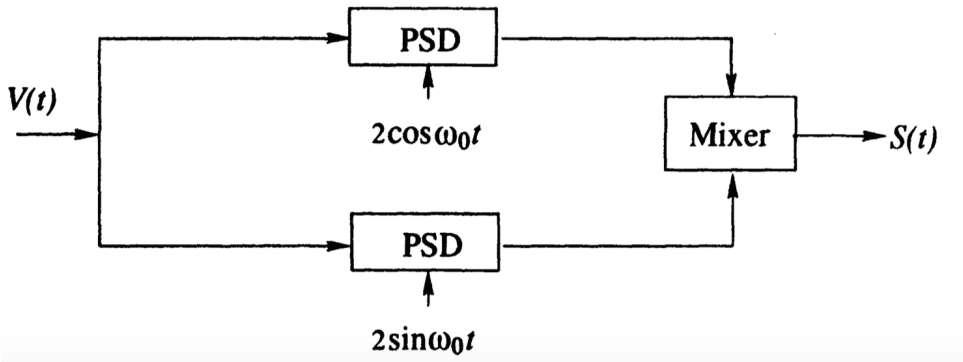
where $\Delta\omega(\mathbf{r}) = \omega(\mathbf{r}) - \omega_0$ is the spatially dependent resonance frequency in the rotating frame. However, now it is not clear whether the precession is clockwise or counterclockwise. Therefore, $V(t)$ is multiplied by $2\sin(\omega_0 t)$ and passed through a low-pass-filter again, yielding $S_I(t)$:

$$S_I(t) = \int_{\text{object}} \omega(\mathbf{r})|\vec{B}_{r,xy}(\mathbf{r})||M_{xy}(\mathbf{r},0)|e^{-t/T_2(\mathbf{r})}\sin\left[-\Delta\omega(\mathbf{r})t + \phi_e(\mathbf{r}) - \phi_r(\mathbf{r}) + \frac{\pi}{2}\right] d\mathbf{r}. \quad (2.23)$$

In this way, the rotating magnetization is detected with two orthogonal detectors. This procedure is known as quadrature detection and is shown schematically in Figure 2.6.



(a) Phase-sensitive detection



(b) Quadrature detection.

Figure 2.6: Schematic representation of quadrature detection.

The two outputs of the system, $S_R(t)$ and $S_I(t)$ are often put in complex form:

$$S(t) = S_R(t) + iS_I(t). \quad (2.24)$$

Then

$$S(t) = \int_{\text{object}} \omega(\mathbf{r})|\vec{B}_{r,xy}(\mathbf{r})||M_{xy}(\mathbf{r},0)|e^{-t/T_2(\mathbf{r})}e^{-i\left(\Delta\omega(\mathbf{r})t - \phi_e(\mathbf{r}) + \phi_r(\mathbf{r}) - \frac{\pi}{2}\right)} d\mathbf{r}. \quad (2.25)$$

Using

$$\begin{cases} |B_{r,xy}(\mathbf{r})|e^{-i\phi_r(\mathbf{r})} = B^*_{r,xy}(\mathbf{r}) \\ |M_{xy}(\mathbf{r},0)|e^{-i\phi_e(\mathbf{r})} = M_{xy}(\mathbf{r},0) \end{cases} \quad (2.26)$$

and omitting the scaling constant $e^{i\pi/2}$, Equation (2.25) can be rewritten as

$$S(t) = \int_{\text{object}} \omega(\mathbf{r}) \vec{B}^*_{r,xy}(\mathbf{r}) M_{xy}(\mathbf{r}, 0) e^{-t/T_2(\mathbf{r})} e^{-i\Delta\omega(\mathbf{r})t} \, d\mathbf{r}. \tag{2.27}$$

Now we assume that $\Delta\omega(\mathbf{r}) << \omega_0$ and that the reception field $B_{r,xy}$ is homogeneous. Then, using $\Delta\omega(\mathbf{r}) = \gamma\Delta B(\mathbf{r})$ and leaving out the constant terms, Equation (2.27) can be simplified to

$$S(t) = \int_{\text{object}} M_{xy}(\mathbf{r}, 0) e^{-t/T_2(\mathbf{r})} e^{-i\gamma\Delta B(\mathbf{r})t} \, d\mathbf{r}. \tag{2.28}$$

2.3 Signal characteristics

2.3.1 Free induction decays

A single RF pulse applied to a nuclear spin system leads to free induction decay (FID) in the spin system. When the magnetic field the sample is exposed to is perfectly homogeneous, the FID signal is characterized by a T_2 decay. However, when the magnetic field is inhomogeneous, differences in precessional frequency arise, causing the FID signal to decay at a much faster rate. A group of nuclear spins with the same precessional frequency is called an isochromat. The time constant T_2^* is used to characterize the signal decay in the presence of field inhomogeneity. The following relationship between T_2 and T_2^* is widely used in MRI literature:

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \gamma\Delta B_0. \tag{2.29}$$

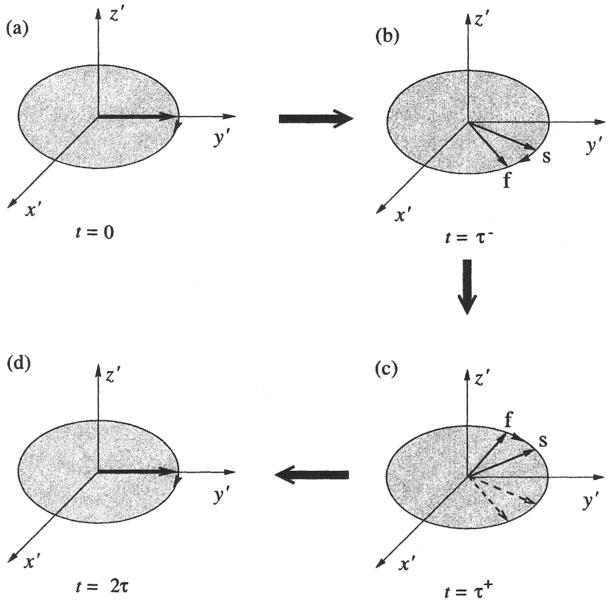


Figure 2.7: Refocusing the bulk magnetic moment of 2 isochromats in a spin-echo experiment.

2.3.2 RF echoes

An echo signal consists of a dephasing period and a refocusing phase. After the RF pulse is applied along the x'-direction (called the 90° pulse), inhomogeneities in the magnetic field will

lead to different isochromats. The isochromats will progressively lose phase coherence as the free precession continues. In order to regain phase coherence, after a time τ, a pulse is applied along the y'-direction that flips the different magnetic moments over to the other side of the transverse plane (the 180° pulse). This will cause the bulk magnetic moment of isochromats with higher precessional speed to lag behind the ones with lower precessional speed. A time τ after the y'-pulse, the vectors will all have the same phase again. This process is illustrated in Figure 2.7 for two isochromats with precessional frequencies ω_s and ω_f. The strength of the signal during this pulsing sequence is shown in Figure 2.8. The decay of the signal is characterized by T_2^*, while the decay in maximum amplitude is characterized by T_2.

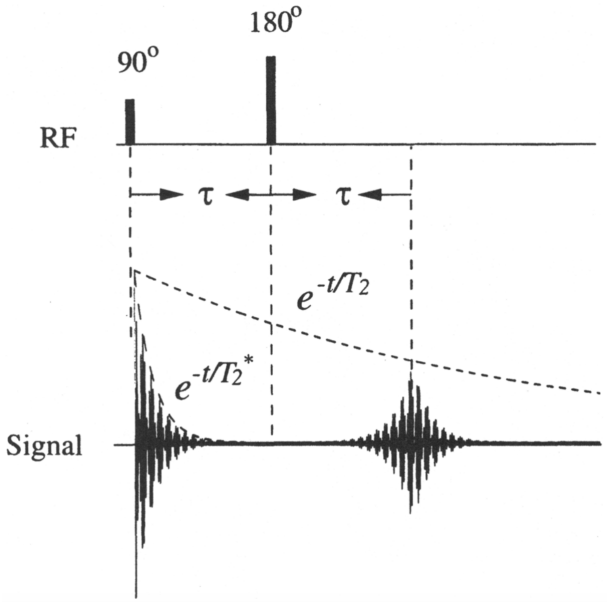


Figure 2.8: Formation of a spin echo signal.

Instead of just one spin echo, a sequence of spin echoes can be generated by repeatedly applying 180° pulses at times τ, 3τ, 5τ,...

2.4 Image contrast

Protons in different materials have different longitudinal and transverse relaxation times, T_1 and T_2. The differences between these parameters can be used to produce contrast in MR images [7]. This is done by tuning the repetition time T_R (the time between two successive B_1-pulses) and the echo time $T_E := 2\tau$ correctly. If T_R is long, all the transverse magnetization will have turned into longitudinal magnetization again. However, if T_R is shorter, a new B_1-pulse is applied before the longitudinal magnetization is restored in all materials, meaning that the transverse magnetization in materials with a long T_1 will be smaller after this pulse than in materials with a short T_1, leading to strong signals from materials with a short T_1 and weak signals from materials with a long T_1: T_1-weighting.

If T_E is short, no significant decay in transverse relaxation will have taken place. However, a long T_E causes materials with a short T_2 to lose their transverse magnetization, while materials with a long T_2 will maintain it. This causes T_2-weighting: materials with a long T_2 yield stronger signals than those with a short T_2.

If T_R is long and T_E is short, only the spin-density will determine the contrast in the image. An overview of how the different kinds of weighting can be achieved is given in Table 2.1.

Table 2.1: Image contrast for different choices of T_E and T_R.

Contrast	T_E	T_R
T_1-weighting	Short	Appropriate
T_2-weighting	Appropriate	Long
Spin density-weighting	Short	Long

2.5 Signal localization

2.5.1 Slice selection

In order to make a 2D image of a 3D object, a slice has to be selected. Then, only the spins in that slice of the object will be excited. To selectively excite spins, a gradient field and a shaped RF pulse are necessary. An RF pulse can only be frequency-selective, which means that the spin resonance frequency has to be made position-dependent. This is achieved by augmenting the B_0-field with a linear gradient field during the excitation period. This gradient field is called a slice-selection gradient. The images are made in the xy-plane, which means that the gradient will be applied in the z-direction, yielding the total magnetic field strength $B(z) = B_0 + G_z z$. Now, to select a slice of thickness Δz centered around z_0, the following spatial selection function is required:

$$p_s(z) = \begin{cases} 1, & |z - z_0| < \frac{\Delta z}{2} \\ 0, & \text{otherwise.} \end{cases} \tag{2.30}$$

It can be shown that the corresponding pulse envelope function has to satisfy

$$B_1^e(t) \propto \operatorname{sinc}\left[\pi \frac{\gamma}{2\pi} G_z \Delta z \left(t - \frac{\tau_p}{2}\right)\right], \tag{2.31}$$

where τ_p is the duration of the pulse.

2.5.2 Spatial information encoding

After the RF pulse, the free precession period allows for spatial information to be encoded into the signal. There are two ways to encode spatial information: frequency encoding and phase encoding.

Frequency encoding When the precession frequency of an activated MR signal is made to be linearly dependent on its spatial origin, frequency encoding is used. This can be achieved by applying a linear gradient field to the magnetic field after the RF pulse. For instance, a gradient G_x in the x-direction leads to a magnetic field

$$\vec{B} = (B_0 + G_x x)\vec{k}, \tag{2.32}$$

which leads to the Larmor frequency being a linear function of x:

$$\omega(x) = \omega_0 + \gamma G_x x. \tag{2.33}$$

Assuming an object with spin distribution $\rho(x, y)$ and omitting the transverse relaxation effect, the FID signal generated locally from spins in an infinitesimal interval dx at point x is

$$dS(x, y, t) \propto \rho(x, y)\ dx\ dy\ e^{-i\gamma(B_0+G_x x)t}. \tag{2.34}$$

Then, the signal generated by the entire object, neglecting the proportionality constant, is

$$S(t) = \int\int_{\text{object}} dS(x, y, t) = \int\int_{\text{object}} \rho(x, y)e^{-i\gamma(B_0+G_x x)t}\ dx\ dy \tag{2.35}$$

$$= \left[\int\int_{\text{object}} \rho(x, y)e^{-i\gamma G_x xt}\ dx\ dy\right] e^{-i\omega_0 t}. \tag{2.36}$$

Removal of the carrier signal $e^{-i\omega_0 t}$ (demodulation) yields

$$S(t) = \int\int_{\text{object}} \rho(x, y)e^{-i\gamma G_x xt}\ dx\ dy. \tag{2.37}$$

Phase encoding After an RF pulse, a gradient G_y is turned on in the y-direction for a short time T_{pe}, and then it is turned off. Then the local signal $dS(x, y, t)$ is described by

$$dS(x, y, t) = \begin{cases} \rho(x, y)\ dx\ dy\ e^{-i\gamma(B_0+G_y y)t}, & 0 \le t \le T_{pe} \\ \rho(x, y)\ dx\ dy\ e^{-i\gamma G_y y T_{pe}} e^{-i\gamma B_0 t}, & T_{pe} \le t. \end{cases} \tag{2.38}$$

During the interval $0 \le t \le T_{pe}$, the preparatory period, the signal is frequency-encoded. Therefore, signals from different y-positions will have different phase angles after a time T_{pe}. At time T_{pe}, the signal will have an initial phase angle

$$\phi(y) = -\gamma G_y y T_{pe}. \tag{2.39}$$

$\phi(y)$ is linearly dependent on the position y, so the signal is phase-encoded. Now,

$$S(t) = \int\int_{\text{object}} dS(x, y, t) = \left[\int\int_{\text{object}} \rho(x, y)e^{-i\gamma G_y y T_{pe}}\ dx\ dy\right] e^{-i\omega_0 t}. \tag{2.40}$$

Again, the carrier signal $e^{-i\omega_0 t}$ will be removed after demodulation.

k-space representation Combining a frequency-encoding gradient G_x in the x-direction and a phase-encoding gradient G_y in the y-direction yields (after demodulation)

$$S(t) = \int\int_{object} \rho(x, y)e^{-i(\gamma G_x xt+\gamma G_y y T_{pe})}\ dx\ dy. \tag{2.41}$$

Using the following substitutions:

$$\begin{cases} k_x &= \gamma G_x t \\ k_y &= \gamma G_y T_{pe}, \end{cases} \tag{2.42}$$

the signal is described by

$$S(k_x, k_y) = \int\int_{object} \rho(x, y)e^{-i(k_x x+k_y y)}\ dx\ dy. \tag{2.43}$$

So the substitutions given by Equation (2.42) allow us to use k-space for signal representation. A conventional strategy to produce sufficient data to cover k-space is to generate a set of "identical" signals $\{S_n(t)\}$ and then encode each one properly so that k-space is covered by multiple lines. This can be done by changing the phase-encoding gradient, because a change in gradient strength leads to a different line in k-space.

2.6 Image reconstruction

In practice, the spin density ρ depends not only on x and y, but also on z: $\rho(x, y, z)$, which means that Equation (2.43) should be replaced by

$$S(k_x, k_y) = \int \int_{object} \int_{slice} \rho(x, y, z) e^{-i(k_x x + k_y y)} \, dz \, dx \, dy. \tag{2.44}$$

The desired image function will be denoted by $I(x, y)$. $I(x, y)$ is related to the spin density $\rho(x, y, z)$ in the following way:

$$I(x, y) = \int_{-\frac{\Delta z}{2}}^{\frac{\Delta z}{2}} \rho(x, y, z) \, dz. \tag{2.45}$$

Therefore, $\rho(x, y)$ is replaced by $I(x, y)$ in Equation (2.43):

$$S(k_x, k_y) = \int \int_{object} I(x, y) e^{-i2\pi(k_x x + k_y y)} \, dx \, dy. \tag{2.46}$$

Now k_x and k_y are described by

$$k_x = \frac{\gamma}{2\pi} G_x t \tag{2.47}$$

$$k_y = \frac{\gamma}{2\pi} G_y T_{pe}, \tag{2.48}$$

in order to cast Equation (2.46) as a Fourier transform. The aim is to obtain $I(x, y)$ given $S(k_x, k_y)$. Now, assuming that k-space is sampled uniformly, the set of k-space points at which measured data is collected is given by

$$\mathcal{D} = \{(k_{n,x}, k_{m,y}) : k_{n,x} = n\Delta k_x, k_{m,y} = m\Delta k_y, \ n, m \in \mathbb{Z}\}. \tag{2.49}$$

The field of view (FOV) is defined as the region $|x| < \frac{W_x}{2}$, $|y| < \frac{W_y}{2}$ where W_x and W_y are finite numbers such that

$$I(x, y) = 0, \quad |x| > \frac{W_x}{2}, |y| > \frac{W_y}{2}. \tag{2.50}$$

Then, if the following inequalities hold:

$$\Delta k_x < \frac{1}{W_x}, \quad \Delta k_y < \frac{1}{W_y}, \tag{2.51}$$

$I(x, y)$ can be shown to be related to $S(x, y)$ in the following way:

$$I(x, y) = \Delta k_x \Delta k_y \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} S(n\Delta k_x, m\Delta k_y) e^{i2\pi(n\Delta k_x x + m\Delta k_y y)}, \tag{2.52}$$
$$|x| < \frac{1}{\Delta k_x}, |y| < \frac{1}{\Delta k_y}.$$

In practice, k-space is sampled a finite number of times. So there are $N, M \in \mathbb{N}$ such that $\mathcal{D}$ is described by

$$\mathcal{D} = \left\{ (k_{n,x}, k_{m,y}) : k_{n,x} = n\Delta k_x, k_{m,y} = m\Delta k_y, \ -\frac{N}{2} \leq n < \frac{N}{2}, -\frac{M}{2} \leq m < \frac{M}{2} \right\}. \tag{2.53}$$

Then, the Fourier reconstruction formula is

$$I(x,y) = \Delta k_x \Delta k_y \sum_{n=-N/2}^{N/2-1} \sum_{m=-M/2}^{M/2-1} S(n\Delta k_x, m\Delta k_y) e^{i2\pi(n\Delta k_x x + m\Delta k_y y)}, \tag{2.54}$$
$$|x| < \frac{1}{\Delta k_x}, |y| < \frac{1}{\Delta k_y}.$$

$I(x,y)$ can be obtained efficiently using a Fast Fourier transform algorithm. Due to the truncation of the Fourier series, the Fourier reconstruction is not identical to the true image function.

Chapter 3

Low-field MRI

As opposed to conventional MRI, this research focuses on low-field MRI, in which magnetic fields in the order of milliTeslas are generated by a resistive magnet. Resistive magnets can be used at room-temperature, as opposed to superconducting magnets that have to be cooled with liquid helium. This makes low-field MRI less costly. However, one of the main issues with low-field MRI is the lower signal-to-noise ratio [6]. This is due to the linear dependence of the SNR on B_0 [8].

3.1 The PSU prototype

3.1.1 Magnetic field

Instead of a static and large $\vec{B}_0$, the prototype MRI scanner uses a pulsed magnetic field $\vec{B}_p$ combined with a very small static field $\vec{B}_0$. One reason for this choice is that generating a larger static B_0 field generates more heat in the coils, so the higher B_0, the more cooling is necessary, and the more expensive the MRI scanner. The B_p-pulses are called prepolarizing pulses. During a B_p-pulse, the nuclear magnetization will tend towards an enhanced equilibrium value, aligned with the higher effective magnetic field $\vec{B}_{eff} = \vec{B}_0 + \vec{B}_p$. Then, the pulse is ramped down slowly, such that the enhanced magnetization will realign with $\vec{B}_0$ without precessing. This enhanced magnetization will decay to the B_0 equilibrium value according to the T_1mvalue of the sample. If a pulsing sequence is conducted before the magnetization has completely died down, the signal will be enhanced. Therefore, prepolarizing leads to a higher signal-to-noise ratio. The pulse sequence that is employed in the low-field MR scanner is shown in Figure 3.1.

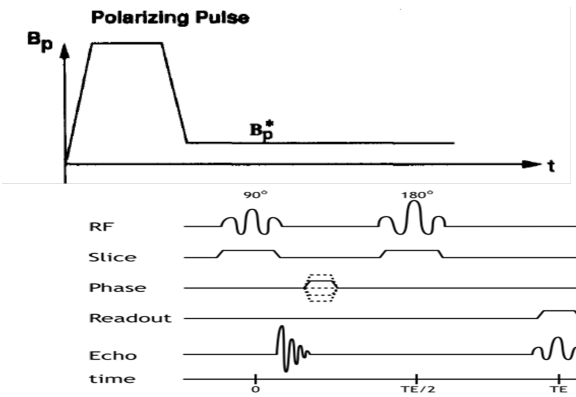


Figure 3.1: The pulse sequence that is used in the low-field MR scanner. Source: PSU.

3.1.2 Field inhomogeneities

In conventional MRI, a Fourier Transform is used to obtain the image given the signal. However, this approach is only possible because of the homogeneity of the B_0-field. In low-field MRI, there is much more inhomogeneity in the magnetic field. Figure 3.2 shows a plot of very coarse measurements of B_0 in the prototype as a function of x and z, where $y = 0$. x and z are defined as in Figure 3.3.

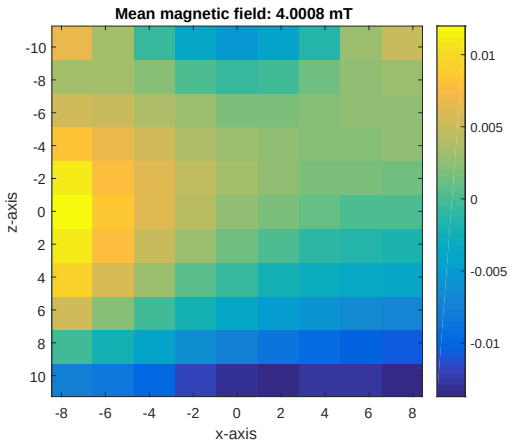


Figure 3.2: Measurements of the B_0-field in the xz-plane, at $y = 0$. The units on the axes are centimeters.

From Figure 3.2, it is clear that the maximum deviation from the mean field is about 0.3%. This does not seem like much, but in conventional MRI the maximum deviation is about a thousand times as small. The PSU team does use the Fourier Transform to obtain the final image.

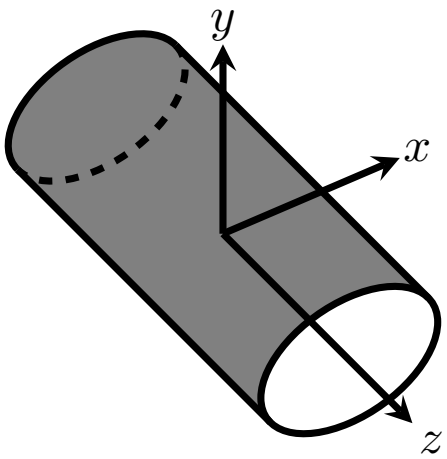


Figure 3.3: x, y and z-direction in the MR scanner.

3.2 The LUMC prototype

A picture of the magnet created by the team at LUMC is shown in Figure 3.4.

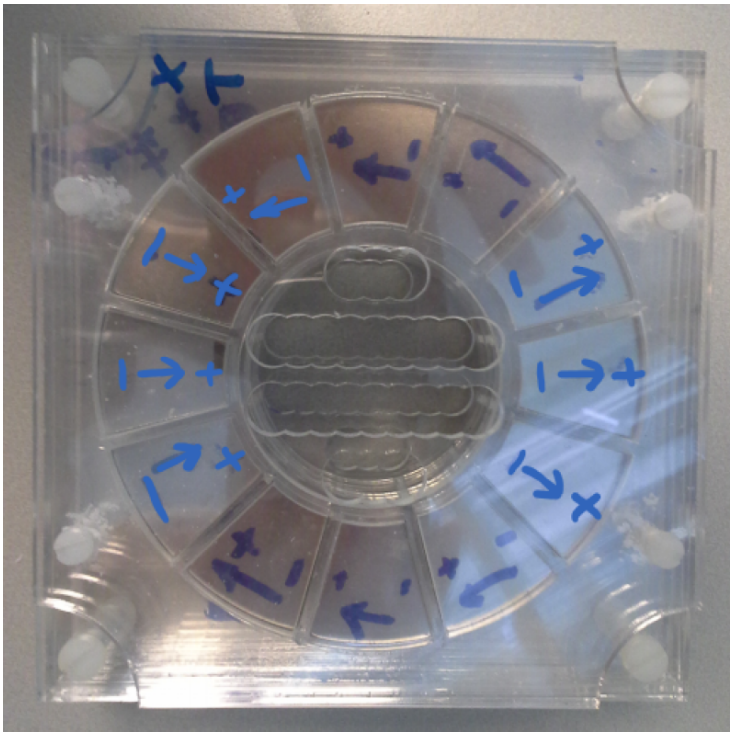
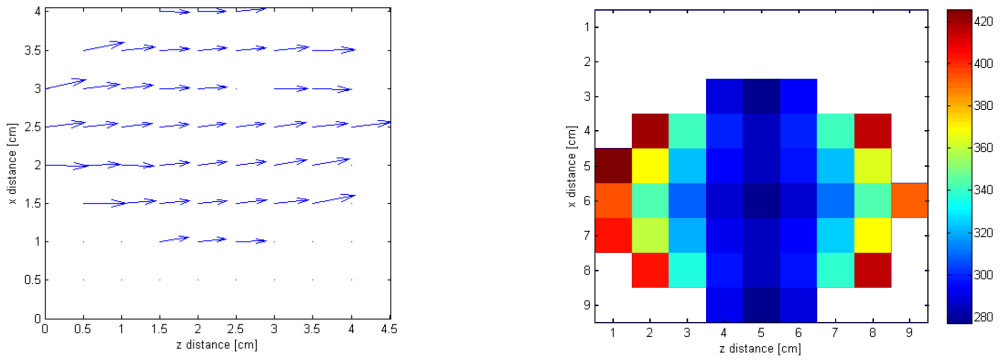


Figure 3.4: The magnet constructed by the team at LUMC. The direction of the magnetization in each magnet is shown using arrows. Source: [9].

They use a configuration of twelve permanent magnets, arranged in a circle, such that the resulting magnetic field (the B_0-field) is oriented mainly in the z-direction. Here, too, the z-axis is oriented along the axis of the cylinder. In [9], the B_0-field was measured at $y = 0$. The result is shown in Figure 3.5.



(a) Direction of the B_0-field.

(b) Magnitude of the B_0-field.

Figure 3.5: B_0-field measurements.

Additionally, the magnetic field was simulated using COMSOL. The result is shown in Figure 3.6.

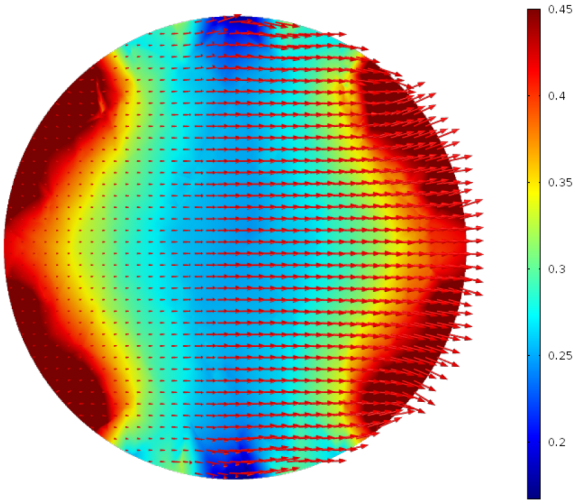


Figure 3.6: Simulated field based on the Halbach array in Figure 3.4

In this prototype, the inhomogeneities in the B_0-field are used to allow for spatial encoding, instead of gradient coils. To obtain a sufficient amount of data, the idea is to rotate either the object or the device and measure the signal after each rotation. To excite the spins, an RF pulse has to be sent out. COMSOL simulations and measurements of the magnetic field inside the magnet show that variations in the field strength are in the order of $10 - 100$ mT. In order to excite all the spins in the phantom, a bandwidth of more than 1 MHz is required. However, a bandwidth of more than 20 kHz is infeasible, forcing us to use a sequence of pulses of different frequencies. Each pulse excites the spins within a frequence band of 20 kHz, until all frequencies have been covered at least once.

3.3 General measurement model

Taking away the assumption of homogeneity, the signal is described by (2.27), but T_2 has to be replaced by T_2^*. Then the following expression is obtained:

$$S(t) = \int_{\text{object}} \omega(\mathbf{r})|\vec{B}_{r,xy}(\mathbf{r})|M_{xy}(\mathbf{r},0)e^{-t/T_2^*(\mathbf{r})}e^{-i\phi_r(\mathbf{r})}e^{-i\gamma\Delta B(\mathbf{r})t}\, d\mathbf{r}. \tag{3.1}$$

Now, denoting the signal sensitivity response pattern of the coil by $c(\mathbf{r})$, the general forward model for the signal is:

$$S(t) = \int_{\text{object}} \omega(\mathbf{r})c(\mathbf{r})|\vec{B}_{r,xy}(\mathbf{r})|M_{xy}(\mathbf{r},0)e^{-t/T_2^*(\mathbf{r})}e^{-i\phi_r(\mathbf{r})}e^{-i\phi(\mathbf{r},t)}\, d\mathbf{r}, \tag{3.2}$$

where

$$\phi(\mathbf{r},t) = \gamma\Delta B(\mathbf{r})t. \tag{3.3}$$

Here, $M_{xy}(\mathbf{r},0)$ is the desired image. For ease of notation, we define

$$x(\mathbf{r}) := M_{xy}(\mathbf{r},0). \tag{3.4}$$

The measurements recorded in an MRI scan consist of noisy samples of the MRI signal described by Equation (3.2):

$$s_i = S(t_i) + e_i, \quad i = 1, ..., L, \tag{3.5}$$

where s_i denotes the ith sample of the signal, measured at time t_i. e_i is the measurement error in the ith sample. The e_i are modeled as additive, zero-mean, white gaussian noise. Using Equation (3.2) to describe the measurement model, the image reconstruction problem is to estimate $x(\mathbf{r})$ from a measurement vector $\mathbf{s}$. $x(\mathbf{r})$ is a continuous function. In order to estimate it using a finite set of measurements, $x(\mathbf{r})$ is approximated using a finite series expansion:

$$x(\mathbf{r}) = \sum_{j=1}^{M} x_j b(\mathbf{r} - \mathbf{r}_j), \tag{3.6}$$

where $b(\cdot)$ denotes the object basis function and $\mathbf{r}_j$ is the center of the jth basis function. Usually, rectangular basis functions are used. In that case, M is the number of pixels. Substituting Equation (3.7) into Equation (3.2) yields

$$S(t_i) = \sum_{j=1}^{M} w_{ij} x_j, \tag{3.7}$$

where

$$w_{ij} = \int_{\text{object}} b(\mathbf{r} - \mathbf{r}_j) c(\mathbf{r}) \omega(\mathbf{r}) |\vec{B}_{r,xy}(\mathbf{r})| e^{-t_i/T_2^*(\mathbf{r})} e^{-\iota \phi_r(\mathbf{r})} e^{-\iota \Delta\phi(\mathbf{r},t)} \, d\mathbf{r}. \tag{3.8}$$

Usually, the basis functions are highly localized, allowing 'center of pixel' approximations to be used:

$$w_{ij} = c(\mathbf{r}_j)\omega(\mathbf{r}_j)|\vec{B}_{r,xy}(\mathbf{r}_j)| e^{-t_i/T_2^*(\mathbf{r}_j)} e^{\iota\zeta(\mathbf{r}_j)} e^{-\iota\Delta\phi(\mathbf{r}_j,t_i)}, \quad \Delta\phi(\mathbf{r}_j, t_i) = \gamma \Delta B(\mathbf{r}_j) t_i. \tag{3.9}$$

Combining Equations (3.5), (3.7) and (3.9) allows us to represent the system of equations in matrix-vector form:

$$\mathbf{s} = W\mathbf{x} + \mathbf{e}, \tag{3.10}$$

where the elements of W are described by Equation (3.9).

3.4 Signal-to-noise ratio

3.4.1 Signal

The principle of reciprocity, introduced by Hoult and Richards in 1976 [10], states that at any spatial location, the size of the voltage induced in the coil by a rotating magnet corresponds to the size of the field ($B_{r,xy}$) that would be produced at that location if a unit current were to flow in the coil [11]. Using this, it can be shown that the amplitude of the free induction decay from a volume ΔV of sample with an equilibrium magnetization M_0 precessing in the xy-plane at the Larmor frequency ω_0, is

$$S_{voxel} = \omega_0 B_{r,xy} M_0 \Delta V. \tag{3.11}$$

For protons at human body temperature, the equilibrium magnetization can be calculated using the following equation [11]:

$$M_0 \approx \rho_0 \frac{\gamma^2 \hbar^2}{4 k_B T} B_0, \tag{3.12}$$

where $k_B = 1.38 \cdot 10^{-23}$ m^2 kg s^{-2} K^{-1} is the Boltzmann constant and T is the temperature. In tissue water, the spin density is $\rho_0 = 6.69 \cdot 10^{28}$ m^{-3}, leading to an equilibrium magnetization of

$$M_0 \approx 0.0031 B_0. \tag{3.13}$$

If we take a solenoidal coil that fits inside the Halbach magnet, the magnitude of the receiver field $B_{r,xy}$ is given by [10]

$$B_{r,xy} = \frac{\mu_0 n}{\sqrt{4a^2 + h^2}}, \tag{3.14}$$

where $\mu_0 = 4\pi \cdot 10^{-7}$ N/A^2 is the permeability of free space, n is the number of turns, a is the radius and h is the length of the coil. Assuming a five-turn coil with length 0.02 m and radius 0.05 m yields a receptor field of 60 mT/A. Assuming the image has 32×32 voxels, the slice thickness is 5 mm and the FoV is 10 cm yields a signal of $2.3 \cdot 10^{-8}$ V.

3.4.2 Noise

The electrons in the receiving coil are subject to random Brownian motion, which causes a small randomly varying voltage to be measured, in addition to the voltage induced by the object being imaged [11]. The root mean square of the random voltage is given by the Nyquist equation [11]:

$$N_{rms} = \sqrt{4 k_B T R \Delta f}, \tag{3.15}$$

where R is the resistance of the receiver coil and Δf is the bandwidth. This thermal or Johnson noise is approximately white noise, which means that its mean is zero and its variance is finite. Because the mean is zero, the standard deviation of the noise is equal to the root mean square value given by Equation (3.15). As mentioned before, the noise will be assumed to be Gaussian. Assuming a coil resistance of 1 Ω and temperature of 293 K (room temperature), the standard deviation is approximately 0.14 nV Hz$^{-\frac{1}{2}}$. The LUMC prototype will have a maximum bandwidth of 20 kHz, leading to a standard deviation of 20 nV.

3.4.3 SNR

Combining Equations (3.11) and (3.15) yields the following expression for the SNR per voxel:

$$SNR = \frac{S_{voxel}}{N_{rms}} = \frac{\omega_0 B_{r,xy} M_0 \Delta V}{\sqrt{4 k_B T R \Delta f}}. \tag{3.16}$$

Using the values found for S_{voxel} and N_{rms}, the SNR per voxel is about 1.6, which shows that the noise has a significant influence on the measurements. Equation (3.16) shows that increasing the voxel size (by decreasing the number of voxels) increases the SNR per voxel. This makes it interesting to investigate whether super-resolution reconstruction can generate better results than direct high-resolution reconstruction.

Chapter 4

Super-resolution reconstruction

SRR uses multiple low resolution (LR) images of the same object to form a high resolution (HR) image. Typically, the different LR images represent different "looks" at the same object. That is, LR images are down-sampled and shifted versions of the HR image. The shift has to be a subpixel shift, because integer shifts yield no new information, as shown in Figure 4.1. Another way of acquiring different LR images would be to rotate the object.

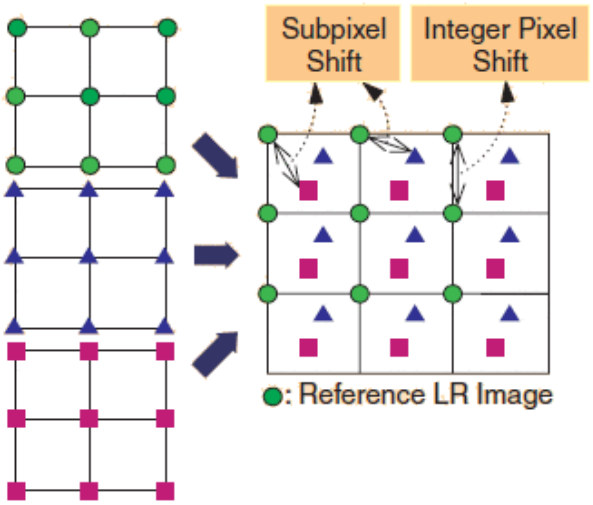


Figure 4.1: Integer shifts do not provide any new information. Subpixel shifts are needed to make SRR possible. Source: [12].

4.1 Data acquisition model

Let X be the desired HR image with dimensions $m \times n$. Then X is an $m \times n$ matrix consisting of the grayscale values of each pixel. In order to apply super-resolution reconstruction, X must be put in vector form:

$$\mathbf{x} = \text{vec}(X) = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_n \end{bmatrix}, \tag{4.1}$$

where $\mathbf{x}_i$ is the ith column of X. This notation will be used throughout the whole report, so images in matrix form will be denoted by capital letters and images in vector form will be denoted by the same letter, but in bold lowercase.

Let $\{Y_k\}_{k=1}^{M}$ be a sequence of LR images. The data acquisition model used in SRR is as follows [12], [5]. First, the HR image X undergoes a geometric deformation, such as a subpixel shift or a rotation. This geometric deformation is described by an operator D_k. Then, during the imaging process, each of the geometrically deformed images is subject to a blurring operator B_k and a down-sampling operator denoted by D_k. Also, noise is added to the system: V_k. The noise is usually assumed to be independent, identically distributed Gaussian noise. A pictorial overview of the acquisition model is shown in Figure 4.2.

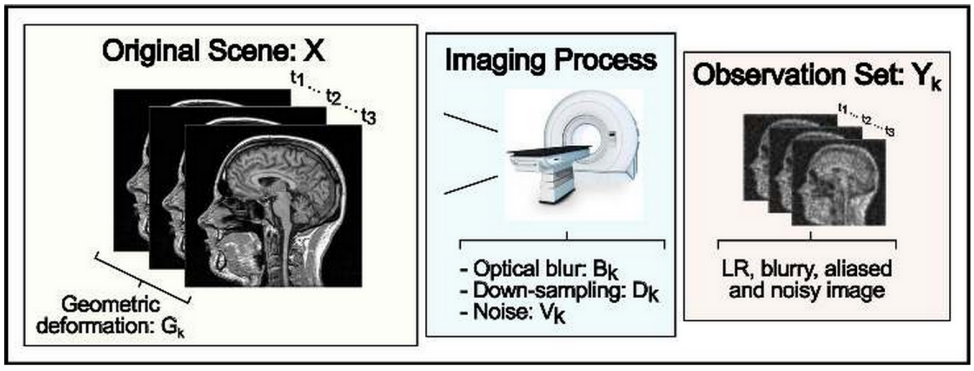


Figure 4.2: The general data acquisition model used in super-resolution reconstruction. Source: [5].

G_k, B_k and D_k can be cast in matrix-form when $\mathbf{x}$, $\mathbf{y}_k$ and $\mathbf{v}_k$ are used instead of X, Y_k and V_k, yielding the following system of equations:

$$\mathbf{y}_k = D_k B_k G_k \mathbf{x} + \mathbf{v}_k, \quad k = 1, ..., M. \tag{4.2}$$

Clearly, the term $D_k B_k G_k$ can be replaced by a single matrix A_k:

$$\mathbf{y}_k = A_k \mathbf{x} + \mathbf{v}_k, \quad k = 1, ..., M. \tag{4.3}$$

By vertically concatenating the vectors $\mathbf{y}_k$, the matrices A_k and the vectors $\mathbf{v}_k$ as follows:

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_M \end{pmatrix}, \quad A = \begin{pmatrix} A_1 \\ \vdots \\ A_M \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_M \end{pmatrix}, \tag{4.4}$$

a single system of equations is obtained:

$$\mathbf{y} = A\mathbf{x} + \mathbf{v}. \tag{4.5}$$

4.1.1 Geometric deformation

Geometric deformation, represented by G_k, is of fundamental importance in SRR, because it yields different views of the same object, providing additional information. As mentioned before, the deformation has to be of subpixel nature, which requires a very accurate estimation of the deformation. It can be difficult to attain such a high precision, which makes registration a bottleneck in reconstruction process. By introducing predetermined motion, this difficulty can be eliminated. When imaging a static object, the deformation is artificially created by shifting or rotating the field of view by a known value [5].

4.1.2 Blurring

The operator B_k represents the blurring caused by the imaging process. Commonly, blurring is assumed to be spatially-invariant, so B_k simplifies to B. The point-spread function (PSF) P is a function that describes the blurring of one pixel over its surrounding pixels. Usually, blurring is a local phenomenon, so one pixel influences only a very limited amount of other pixels around it. Outside a certain radius, its influence is essentially zero. Therefore, P is a very small matrix compared to the image matrices. Let Z be an image and let Z_b be the blurred version of Z. If the PSF is assumed to be spatially invariant, the blurring of Z can be represented by a two-dimensional convolution:

$$Z_b = P \ast\ast Z. \tag{4.6}$$

Most articles concerning SRR in MRI suggest that the PSF is well approximated by a Gaussian function. In the 2D case, the (unscaled) elements of the PSF are described by

$$p_{ij} = \exp\left(-\frac{1}{2}\begin{bmatrix} i-k \\ j-l \end{bmatrix}^T \begin{bmatrix} s_1^2 & \rho^2 \\ \rho^2 & s_2^2 \end{bmatrix} \begin{bmatrix} i-k \\ j-l \end{bmatrix}\right), \tag{4.7}$$

where s_1 and s_2 determine the width of the PSF and ρ determines its orientation. (k, l) is the central element of P. All elements of a PSF have to sum to 1, so P has to be scaled accordingly. The Gaussian function decays exponentially away from the center and it is reasonable to truncate the values of the PSF when they have decayed by a sufficiently large factor $(10^4 - 10^8)$ [13]. If $\rho = 0$, the PSF is symmetrical along the horizontal and vertical axes and the PSF takes the simpler form

$$p_{ij} = \exp\left(-\frac{1}{2}\left(\frac{i-k}{s_1}\right)^2 - \frac{1}{2}\left(\frac{j-l}{s_2}\right)^2\right). \tag{4.8}$$

The blurring matrix B_k can be obtained from P. The blurred image is equal to the convolution of P and Z, so pixel (i, j) of Z_b can be computed by rotating P by 180 degrees and matching it with pixels in the image Z by placing the center of P over pixel (i, j) of Z. Corresponding components are multiplied and the results are summed to compute element (i, j) of Z_b. This can be carried out for all elements in Z_b, leading to the blurring matrix B.

Chapter 5

Minimization problem

5.1 Least-squares solution

When reconstructing the image from the obtained signal, Equation (3.10) describes the dependence of the signal on the image $\mathbf{x}$. SRR aims to solve Equation (4.5) for $\mathbf{x}$. So both of these frameworks lead to a system of equations of the form $\mathbf{y} = A\mathbf{x} + \mathbf{v}$ (but in the direct reconstruction case, A is replaced by W, $\mathbf{y}$ by $\mathbf{s}$ and $\mathbf{v}$ by $\mathbf{e}$). However, this problem is ill-posed. In order to obtain an $\mathbf{x}$ that describes the data well, $||\mathbf{y} - A\mathbf{x}||_2^2$ has to be small. $||\cdot||_2$ denotes the ℓ_2-norm. If A^*A is nonsingular, the least-squares solution, so the solution that minimizes $||\mathbf{y} - A\mathbf{x}||_2^2$, is

$$\mathbf{x}_{\text{ls}} = (A^*A)^{-1}A^*\mathbf{y}. \tag{5.1}$$

$\mathbf{x}_{\text{ls}}$ is equal to

$$\mathbf{x}_{\text{ls}} = (A^*A)^{-1}A^*\mathbf{y} = \mathbf{x} + (A^*A)^{-1}A^*\mathbf{v}. \tag{5.2}$$

The last term is called the inverted noise. We can rewrite A^*A using its singular value decomposition:

$$A^*A = U\Sigma G^*, \tag{5.3}$$

where U and G are orthogonal matrices and $\Sigma = \text{diag}(\sigma_i)$ is a diagonal matrix whose elements σ_i are nonnegative and appear in nonincreasing order. The σ_i are the singular values of A^*A. If A^*A is nonsingular, its inverse can be written as

$$(A^*A)^{-1} = G\Sigma^{-1}U^*. \tag{5.4}$$

So the inverted noise is equal to

$$(A^*A)^{-1}A^*\mathbf{v} = G\Sigma^{-1}U^*A^*\mathbf{y} = \sum_i \frac{\mathbf{u}_i^*A^*\mathbf{v}}{\sigma_i}\mathbf{g}_i, \tag{5.5}$$

where $\mathbf{u}_i$ and $\mathbf{g}_i$ are the ith column vectors of U and G respectively. If A^*A is almost singular, its smallest singular values will be very close to 0, causing the inverted noise to become large, contaminating the reconstructed image.

5.2 Tikhonov regularization

In order to obtain a solution that is not corrupted by noise, a Tikhonov regularization term can be added to the problem, which leads to the following minimization problem:

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2 + \frac{1}{2}\lambda||F\mathbf{x}||_2^2, \tag{5.6}$$

where λ is the regularization parameter and F is an operator. The term $||F\mathbf{x}||_2^2$ allows us to enforce prior information that is available about $\mathbf{x}$. For instance, F can be chosen to be a first-order difference matrix, because in MR images it is very likely that neighboring pixels have the same value. The value of λ determines to what extent this regularization term is taken into account. A small λ will cause more emphasis to be placed on obtaining a solution that fits the data well, i.e. make sure that the term $\frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2$ is small, while a large λ will ensure that the prior information is enforced, so $\frac{1}{2}||F\mathbf{x}||_2^2$ will be small.

It is not immediately clear which value of λ leads to the best solution. One way of choosing λ is by employing the L-curve criterion [14]. The L-curve is a log-log plot of $||\mathbf{y} - A\mathbf{x}||_2$ versus $||F\mathbf{x}||_2$ for a number of values of λ. This plot often has an L-shape and the best regularization parameter is supposed to lie in the corner of the L, balancing out the two terms.

Equation (5.6) is a convex problem, which means that taking the gradient and setting it equal to 0 yields a condition for the global optimal solution:

$$(A^*A + \lambda F^*F)\mathbf{x} = A^*\mathbf{y}, \tag{5.7}$$

leading to

$$\mathbf{x} = (A^*A + \lambda F^*F)^{-1}A^*\mathbf{y}. \tag{5.8}$$

5.3 Total variation regularization

Tikhonov regularization is not the only possible way of regularizing the problem. Another popular choice is the ℓ_1 term where F is a first-order difference matrix, defined such that

$$||F\mathbf{x}||_1 = \sum_{k=1}^{n}\sum_{l=2}^{m}|X_{l,k} - X_{l-1,k}| + \sum_{l=1}^{m}\sum_{k=2}^{n}|X_{l,k} - X_{l,k-1}|. \tag{5.9}$$

So the minimization problem becomes

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2 + \frac{1}{2}\lambda||F\mathbf{x}||_1. \tag{5.10}$$

The $||F\mathbf{x}||_1$ term penalizes jumps between neighboring pixels. However, jumps are not penalized as severely as in the Tikhonov case, because the ℓ_2 norm makes the regularization term grow quadratically with the difference in value between neighboring pixels, while using the ℓ_1 norm ensures that the penalization grows only linearly. Penalizing large jumps very harshly is undesirable, because MR images tend to contain large jumps between different tissues. The form of regularization used in Equation (5.10) (with F as defined in Equation (5.9)) is called total variation regularization.

5.4 Edge-preserving regularization

When Tikhonov regularization is used, the penalty term grows quadratically with the difference between pixels. The good thing about this is that neighboring pixels are very likely to have the same value, which is desirable in MR images. However, jumps between pixels of different tissues will be severely penalized, leading to overly smooth images. In that case, total variation minimization yields a better result. Edge-preserving regularization combines the advantages of both types of regularization, such that up to a certain value, discontinuities are penalized in a quadratic way, while larger jumps are penalized in a linear way. The rest of this chapter is based on [15].

5.4.1 MAP estimation

Maximum a posteriori (MAP) estimation is another technique commonly used to estimate the original image X given a degraded image Y (or in our case, several degraded images). A MAP technique maximizes the conditional probability of $\mathbf{x}$ when $\mathbf{y}$ is given:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} [\log P(\mathbf{x}|\mathbf{y})]. \tag{5.11}$$

Using Bayes' formula and eliminating constant terms, this can be rewritten as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} [-\log P(\mathbf{y}|\mathbf{x}) - \log P(\mathbf{x})]. \tag{5.12}$$

Because the noise is assumed to be independent, identically distributed Gaussian, the probability density of $\mathbf{y}$ given $\mathbf{x}$ is

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{mn/2}} \exp\left(\frac{-||\mathbf{y} - A\mathbf{x}||^2}{2\sigma^2}\right) \tag{5.13}$$

where σ is the standard deviation of the noise.

5.4.2 Huber-Markov random field model

We suppose that our image X can be modeled as a Markov random field, meaning that the conditional distribution of a pixel value, given all the other pixel values, is only dependent on its neighbors. So, defining the set of neighbors of pixel s by ∂s, we have [16]

$$P(x_s|x_{r\neq s}) = P(x_s|x_{\partial s}). \tag{5.14}$$

Next, we assume that $P(\mathbf{x})$ is a Gibbs distribution:

$$P(\mathbf{x}) = \frac{1}{Z} \exp\left(-\frac{1}{\mu} \sum_{c\in\mathcal{C}} \rho(\mathbf{h}_c^* \mathbf{x})\right), \tag{5.15}$$

where Z is a normalizing constant, μ is the temperature parameter and $\mathbf{h}_c$ is the coefficient vector for the group of pixels c. $\rho(\cdot)$ is a function that has to satisfy the following properties: convexity, symmetry and $\rho(t) << t^2$ for large $|t|$. The Huber function, defined as

$$\rho_T(t) = \begin{cases} t^2, & |t| \leq T \\ T^2 + 2T(|t| - T), & |t| > T \end{cases} \tag{5.16}$$

satisfies these properties. In Figure 5.1, the Huber function is plotted as a function of t:

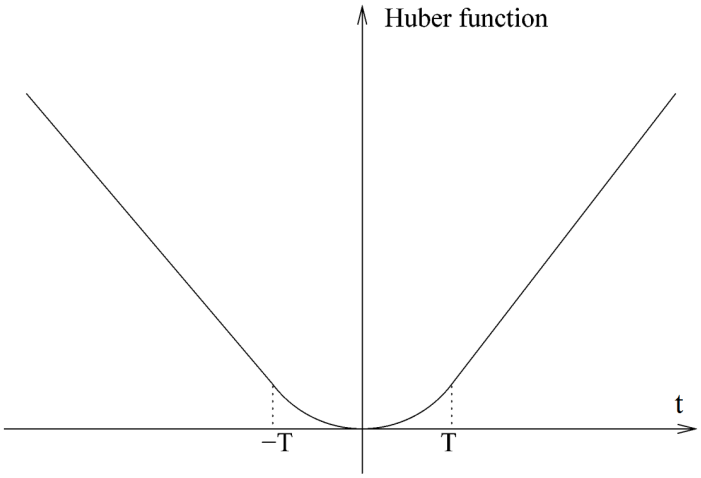


Figure 5.1: The Huber function.

In [15], second-order approximations are used to measure image roughness. However, because we aim to discourage jumps between neighboring pixels as in total variation regularization, first-order approximations will be used in this research:

$$\mathbf{h}^*_{k,l,0}\mathbf{x} = -\frac{1}{2}X_{k,l-1} + \frac{1}{2}X_{k,l} \tag{5.17}$$

$$\mathbf{h}^*_{k,l,1}\mathbf{x} = -\frac{1}{2}X_{k-1,l} + \frac{1}{2}X_{k,l}, \tag{5.18}$$

leading to the following expression for $-\log P(\mathbf{x})$:

$$-\log P(\mathbf{x}) = -\log Z + \frac{1}{\mu}\sum_k \sum_l \sum_{m=0}^{1} \rho_T(\mathbf{h}^*_{k,l,m}\mathbf{x}). \tag{5.19}$$

Now, the functional to be minimized is given by

$$M_T(\mathbf{x}) = ||\mathbf{y} - A\mathbf{x}||^2 + \frac{2\sigma^2}{\mu}\sum_k \sum_l \sum_{m=0}^{1} \rho_T(\mathbf{h}^*_{k,l,m}\mathbf{x}). \tag{5.20}$$

5.4.3 Majorization

Majorization is an iterative technique that minimizes a function by minimizing its majorizing function. A function $g(\theta)$ is said to majorize $f(\theta)$ at θ_i if

$$f(\theta_i) = g(\theta_i) \tag{5.21}$$

$$f(\theta) \leq g(\theta) \text{ for all } \theta. \tag{5.22}$$

In order to minimize Equation (5.20), a series of functionals $N^i_T(\mathbf{x})$ is defined such that

$$M_T(\mathbf{x}^i) = N^{i+1}_T(\mathbf{x}^i) \tag{5.23}$$

$$M_T(\mathbf{x}) \leq N^{i+1}_T(\mathbf{x}) \quad \text{for all } \mathbf{x}, \tag{5.24}$$

where $\mathbf{x}^i$ is the value at which $N^i_T(\mathbf{x})$ attains its minimum. This algorithm can be shown to converge. To majorize the Huber function in Equation (5.16), we define

$$\tilde{N}^{i+1}_T(t) = \begin{cases} t^2, & |t| \leq T \\ \frac{T}{|t^i|}t^2 + T|t^i| - T^2, & |t| > T. \end{cases} \tag{5.25}$$

Here, t^i is the minimizing point of $\tilde{N}_T^i(t)$. Note that $\tilde{N}_T^i(t)$ is a quadratic function. Now let H_m denote the operator corresponding to $\sum_k \sum_l \mathbf{h}_{k,l,m}^* \mathbf{x}$. The constant term $T|t^i| - T^2$ can be ignored, leaving the modified function

$$\hat{N}_T^{i+1}(\mathbf{x}) = \sum_{m=0}^{1} (H_m \mathbf{x})^* \Gamma_m H_m \mathbf{x}, \tag{5.26}$$

where Γ_m are diagonal matrices whose elements are equal to either 1 (if $|\mathbf{h}_{k,l,m}^* \mathbf{x}| \leq T$) or $\frac{T}{|\mathbf{h}_{k,l,m}^* \mathbf{x}^\mathbf{i}|}$ (if $|\mathbf{h}_{k,l,m}^* \mathbf{x}| > T$). Then, the functional $N_T^{i+1}(\mathbf{x})$ is defined as

$$N_T^{i+1}(\mathbf{x}) = ||\mathbf{y} - A\mathbf{x}||^2 + \lambda \sum_{m=0}^{1} (H_m \mathbf{x})^* \Gamma_m H_m \mathbf{x}, \tag{5.27}$$

where $\lambda = \frac{2\sigma^2}{\mu}$ can be seen as a regularizing constant. Taking the gradient of Equation (5.27) yields an optimality criterion for $\mathbf{x}^{i+1}$:

$$\left(A^* A + \lambda \sum_{m=0}^{1} H_m^* \Gamma_m H_m \right) \mathbf{x}^{i+1} = A^* \mathbf{y}. \tag{5.28}$$

Of course, we do not know the values contained in Γ_m beforehand. In order to obtain the correct matrix, fixed-point iteration is used.

Chapter 6

Conjugate gradient methods

As seen in the previous chapter,

$$(A^*A + \lambda F^*F)\mathbf{x} = A^*\mathbf{y}, \tag{6.1}$$

provides us with an optimality criterion for minimization problem (5.6). The solution to (6.1) is

$$\mathbf{x} = (A^*A + \lambda F^*F)^{-1}A^*\mathbf{y}. \tag{6.2}$$

When X is an image with $n \times m$ pixels, $A^*A + \lambda F^*F$ is an $n^2m^2 \times n^2m^2$ matrix. That means that when X is a high resolution image, $A^*A + \lambda F^*F$ becomes very large, making it computationally infeasible to execute a matrix inversion. We will rely on iterative solvers to obtain the solution to Equation (6.1). The methods of choice will be conjugate gradient methods. Sections 6.1, 6.2, 6.3 and 6.4 are based on [17] and [18].

6.1 General problem statement

6.1.1 Tikhonov regularization

The minimization problem (5.6) is a special form of

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - A\mathbf{x}||^2_{C^{-1}} + \frac{1}{2}\lambda||\mathbf{x}||^2_R, \tag{6.3}$$

where $R = F^*F$ and C is the covariance matrix of the noise. Now, the optimality criterion is represented by the normal equations:

$$(A^*C^{-1}A + \lambda R)\mathbf{x} = A^*C^{-1}\mathbf{y}. \tag{6.4}$$

Note that minimization problem (6.3) can be formulated as the following constrained minimization problem:

$$\begin{aligned} \min_{\mathbf{r},\mathbf{x}} \frac{1}{2}||\mathbf{r}||^2_C + \frac{1}{2}\lambda||\mathbf{x}||^2_R \\ \text{subject to } \mathbf{r} = C^{-1}(\mathbf{y} - A\mathbf{x}). \end{aligned} \tag{6.5}$$

Using the technique of Lagrange multipliers, we find that

$$\mathbf{r} = C^{-1}(\mathbf{y} - A\mathbf{x}), \quad \lambda R\mathbf{x} = A^*\mathbf{r}. \tag{6.6}$$

If λR is invertible, $\mathbf{x}$ can be eliminated, yielding

$$\left(\frac{1}{\lambda}AR^{-1}A^* + C\right)\mathbf{r} = \mathbf{y} \tag{6.7}$$

and $\mathbf{x}$ can be obtained from $\mathbf{r}$:

$$\mathbf{x} = \frac{1}{\lambda}R^{-1}A^*\mathbf{r}. \tag{6.8}$$

In order to be able to work with $\lambda = 0$, we can define $\mathbf{z} := \frac{1}{\lambda}\mathbf{r}$, yielding the system

$$(AR^{-1}A^* + \lambda C)\mathbf{z} = \mathbf{y}, \quad \mathbf{x} = R^{-1}A^*\mathbf{z}. \tag{6.9}$$

Since minimization problems (6.3) and (6.5) are equivalent, Equations (6.4) and (6.9) are equivalent too.

6.2 Standard conjugate gradient

The conjugate gradient method was developed by Hestenes and Stiefel [19] in 1952. It is a Krylov subspace method for solving systems of the form $K\mathbf{u} = \mathbf{f}$, where K is a square Hermitian positive definite matrix. Starting with an initial estimate $\mathbf{u}_0$ of the solution $\mathbf{u}$, new estimates $\mathbf{u}_1$, $\mathbf{u}_2$, ... of $\mathbf{u}$ are determined, with $\mathbf{u}_{k+1}$ being closer to $\mathbf{u}$ than $\mathbf{u}_k$. The search directions are denoted by $\mathbf{p}_k$. At each step, the residual $\mathbf{s}_k$ is computed:

$$\mathbf{s}_k = \mathbf{f} - K\mathbf{u}_k. \tag{6.10}$$

In each iteration, the conjugate gradient method computes $\mathbf{u}_k$ such that

$$||\mathbf{u} - \mathbf{u}_k||_K = \min_{\substack{\mathbf{v}\in\mathbf{u}_0+ \\ \text{span}\{\mathbf{p}_0,,\mathbf{p}_{k-1}\}}} ||\mathbf{u} - \mathbf{v}||_K. \tag{6.11}$$

Given an iterate $\mathbf{u}_k$, the gradient descent method would use $\mathbf{s}_k$ as a search vector. The conjugate gradient method uses search vectors that are conjugate with respect to K (so $\mathbf{p}_k K\mathbf{p}_l = 0$ for $k \neq l$). If we have n vectors satisfying this condition, $P := \{\mathbf{p}_0, ..., \mathbf{p}_{n-1}\}$ forms a basis for $\mathbb{C}^n$, which means that $\mathbf{u}$ can be written as a weighted sum of the $\mathbf{p}_k$:

$$\mathbf{u} = \sum_{i=0}^{n-1} \alpha_i \mathbf{p}_i. \tag{6.12}$$

Reasoning from Equation (6.12), $\mathbf{u}_{k+1}$ is calculated as follows:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \alpha_k \mathbf{p}_k. \tag{6.13}$$

Then the residual $\mathbf{s}_{k+1}$ is

$$\mathbf{s}_{k+1} = \mathbf{f} - K\mathbf{u}_{k+1} = \mathbf{f} - K(\mathbf{u}_k + \alpha_k\mathbf{p}_k) = \mathbf{s}_k - \alpha_k K\mathbf{p}_k. \tag{6.14}$$

Given $\{\mathbf{p}_0, ..., \mathbf{p}_k\}$, $\mathbf{p}_{k+1}$ can be calculated using a Gram-Schmidt process:

$$\mathbf{p}_{k+1} = \mathbf{s}_{k+1} - \sum_{i\leq k} \frac{\mathbf{p}_i^* K\mathbf{s}_k}{\mathbf{p}_i^* K\mathbf{p}_i}\mathbf{p}_i. \tag{6.15}$$

However, this would involve the storage of all previous search directions. In [20], it is shown that $\mathbf{s}_{k+1}$ is orthogonal to $\mathbf{p}_i$ for all $i \leq k$, which enables us to store only $\mathbf{s}_k$, $\mathbf{p}_k$ and $\mathbf{u}_k$ and still calculate $\mathbf{s}_{k+1}$, $\mathbf{p}_{k+1}$ and $\mathbf{u}_{k+1}$. $\mathbf{p}_{k+1}$ is calculated as follows:

$$\mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k \mathbf{p}_k \tag{6.16}$$

where β_k is chosen such that $\mathbf{p}_{k+1}$ is conjugate to $\mathbf{p}_k$. Multiplying Equation (6.12) on the left by $\mathbf{p}_k^* K$ yields

$$\alpha_k = \frac{\mathbf{p}_k^* \mathbf{f}}{\mathbf{p}_k^* K \mathbf{p}_k}. \tag{6.17}$$

Substituting Equations (6.10) and (6.16) into Equation (6.17) leads to an expression for α_k:

$$\alpha_k = \frac{\mathbf{p}_k^* \mathbf{f}}{\mathbf{p}_k^* K \mathbf{p}_k} = \frac{\mathbf{p}_k^* (\mathbf{s}_k + K\mathbf{u}_k)}{\mathbf{p}_k^* K \mathbf{p}_k} = \frac{\mathbf{p}_k^* \mathbf{s}_k}{\mathbf{p}_k^* K \mathbf{p}_k} = \frac{\mathbf{s}_k^* \mathbf{s}_k}{\mathbf{p}_k^* K \mathbf{p}_k}. \tag{6.18}$$

In order to calculate β_k, Equation (6.16) is multiplied on the right by $K\mathbf{p}_k$, yielding

$$\beta_k = -\frac{\mathbf{s}_{k+1}^* K \mathbf{p}_k}{\mathbf{p}_k^* K \mathbf{p}_k}. \tag{6.19}$$

From Equation (6.14), we see that $K\mathbf{p}_k = \frac{1}{\alpha_k}(\mathbf{s}_k - \mathbf{s}_{k+1})$, so

$$\beta_k = -\frac{\mathbf{s}_{k+1}^* K \mathbf{p}_k}{\mathbf{p}_k^* K \mathbf{p}_k} = -\frac{\mathbf{s}_{k+1}^* \frac{1}{\alpha_k}(\mathbf{s}_k - \mathbf{s}_{k+1})}{\mathbf{p}_k^* \frac{1}{\alpha_k}(\mathbf{s}_k - \mathbf{s}_{k+1})} = \frac{\mathbf{s}_{k+1}^* \mathbf{s}_{k+1}}{\mathbf{s}_k^* \mathbf{s}_k}. \tag{6.20}$$

Algorithm 1 is the complete conjugate gradient algorithm.

Algorithm 1 STANDARD CG

Require: $K \in \mathbb{C}^{N \times N}, \mathbf{u}_0, \mathbf{f} \in \mathbb{C}^N$;
Ensure: Approximate solution $\mathbf{u}_k$ such that $\|\mathbf{f} - K\mathbf{u}_k\| \leqslant TOL$.
$\mathbf{s}_0 = \mathbf{f} - K\mathbf{u}_0$; $\mathbf{p}_0 = \mathbf{s}_0$; $\gamma_0 = \mathbf{s}_0^* \mathbf{s}_0$;
while $\sqrt{\gamma_k} > TOL$ **and** $k < k_{max}$ **do**
  $\xi_k = \mathbf{p}_k^* K \mathbf{p}_k$
  $\alpha_k = \frac{\gamma_k}{\xi_k}$
  $\mathbf{u}_{k+1} = \mathbf{u}_k + \alpha_k \mathbf{p}_k$
  $\mathbf{s}_{k+1} = \mathbf{s}_k - \alpha_k K \mathbf{p}_k$
  $\gamma_{k+1} = \mathbf{s}_{k+1}^* \mathbf{s}_{k+1}$
  $\beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
  $\mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k \mathbf{p}_k$
  $k = k + 1$
end while

From Equations (6.14) and (6.16), it follows that $\text{span}\{\mathbf{p}_0, \mathbf{p}_1, ..., \mathbf{p_{k-1}}\}$ is equal to the Krylov subspace $\mathcal{K}_k(K, \mathbf{p}_0) := \text{span}\{\mathbf{p}_0, K\mathbf{p}_0, ..., K^{k-1}\mathbf{p}_0\}$. So the conjugate gradient method minimizes $||\mathbf{u} - \mathbf{u}_k||_K$ over $\mathbf{u}_0 + \mathcal{K}_k(K, \mathbf{p}_0)$.

For the convergence behavior of the conjugate gradient method, the following holds:

$$||\mathbf{u} - \mathbf{u}_k||_A \leq 2 \left(\frac{\sqrt{\kappa_2(K)} - 1}{\sqrt{\kappa_2(K)} + 1} \right)^k ||\mathbf{u} - \mathbf{u}_0||_A, \tag{6.21}$$

where $\kappa(K)$ is the condition number of the matrix K. So the smaller the condition number, the faster the convergence.

6.3 CGLS

The conjugate gradient method for least squares (CGLS) is obtained by applying the standard conjugate gradient method to the normal equations $A^*A\mathbf{x} = A^*\mathbf{y}$. Additionally, some modifications are made to enhance stability. This method minimizes the residual in every step, because

$$||\mathbf{x} - \mathbf{x}_k||_{A^*A} = (\mathbf{x} - \mathbf{x}_k)^* A^* A(\mathbf{x} - \mathbf{x}_k) = (\mathbf{y} - A\mathbf{x}_k)^*(\mathbf{y} - A\mathbf{x}_k). \tag{6.22}$$

Something similar can be done using the normal Equations (6.4). By replacing K by $A^*C^{-1}A + \lambda R$, $\mathbf{u}$ by $\mathbf{x}$ and $\mathbf{f}$ by $A^*C^{-1}\mathbf{y}$, the standard conjugate gradient method in Algorithm 1 can be applied to this problem. By defining

$$\mathbf{r}_k := C^{-1}(\mathbf{y} - A\mathbf{x}_k), \tag{6.23}$$

a recursion for the residual is introduced:

$$\begin{aligned}\mathbf{s}_{k+1} &= A^*C^{-1}\mathbf{y} - (A^*C^{-1}A + \lambda R)\mathbf{x}_{k+1} = A^*C^{-1}(\mathbf{y} - A\mathbf{x}_{k+1}) - \lambda R\mathbf{x}_{k+1} \\ &= A^*\mathbf{r}_{k+1} - \lambda R\mathbf{x}_{k+1}.\end{aligned} \tag{6.24}$$

Defining $\mathbf{q}_k = A\mathbf{p}_k$, ξ_k is calculated as

$$\xi_k = \mathbf{q}_k^* C^{-1}\mathbf{q}_k + \lambda \mathbf{p}_k^* R\mathbf{p}_k. \tag{6.25}$$

Algorithm 2 shows CGLS tailored specifically to Equation (6.4).

Algorithm 2 CGLS

Require: $A \in \mathbb{C}^{M\times N}, C \in \mathbb{C}^{M\times M}, R \in \mathbb{C}^{N\times N}, \mathbf{x}_0 \in \mathbb{C}^N, \mathbf{y} \in \mathbb{C}^M, \lambda \in \mathbb{R} \geq 0$;
Ensure: Approximate solution $\mathbf{x}_k$ such that $\|A^*\mathbf{r}_k - \lambda R\mathbf{x}_k\| \leqslant TOL$.
1: $\mathbf{r}_0 = C^{-1}(\mathbf{y} - A\mathbf{x}_0)$; $\mathbf{s}_0 = A^*\mathbf{r}_0 - \lambda R\mathbf{x}_0$; $\mathbf{p}_0 = \mathbf{s}_0$; $\mathbf{q}_0 = A\mathbf{p}_0$; $\gamma_0 = \mathbf{s}_0^*\mathbf{s}_0$; $k = 0$;
2: **while** $\sqrt{\gamma_k} > TOL$ **and** $k < k_{max}$ **do**
3: $\quad \xi_k = \mathbf{q}_k^* C^{-1}\mathbf{q}_k + \lambda \mathbf{p}_k^* R\mathbf{p}_k$
4: $\quad \alpha_k = \frac{\gamma_k}{\xi_k}$
5: $\quad \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k\mathbf{p}_k$; $R\mathbf{x}_{k+1} = R\mathbf{x}_k + \alpha_k R\mathbf{p}_k$
6: $\quad \mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k C^{-1}\mathbf{q}_k$
7: $\quad \mathbf{s}_{k+1} = A^*\mathbf{r}_{k+1} - \lambda R\mathbf{x}_{k+1}$
8: $\quad \gamma_{k+1} = \mathbf{s}_{k+1}^*\mathbf{s}_{k+1}$
9: $\quad \beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
10: $\quad \mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k\mathbf{p}_k$
11: $\quad \mathbf{q}_{k+1} = A\mathbf{p}_{k+1}$
12: $\quad k = k + 1$
13: **end while**

In each iteration, $\mathbf{e}_k := \mathbf{x} - \mathbf{x}_k$ is minimized over the Krylov subspace $\mathbf{x}_0 + \mathcal{K}_k(A^*C^{-1}A + \lambda R, \mathbf{p}_0)$ in the $(A^*C^{-1}A + \lambda R)$-norm. So

$$||\mathbf{e}_k||^2_{A^*C^{-1}A+\lambda R} = (\mathbf{x} - \mathbf{x}_k)^*(A^*C^{-1}A + \lambda R)(\mathbf{x} - \mathbf{x}_k) \tag{6.26}$$

$$= (\mathbf{r} - \mathbf{r}_k)^* C(\mathbf{r} - \mathbf{r}_k) + \lambda \mathbf{e}_k^* R\mathbf{e}_k = ||\mathbf{r} - \mathbf{r}_k||^2_C + \lambda||\mathbf{e}_k||^2_R \tag{6.27}$$

is minimized in every iteration. Note the correspondence with the original constrained minimization problem (6.5).

6.4 CGNE

CGNE is obtained by setting $\mathbf{x} = A^*\mathbf{z}$ and applying standard CG to the normal equations

$$AA^*\mathbf{z} = \mathbf{y}. \tag{6.28}$$

For underdetermined problems, solving this equation is more natural than solving $A^*A\mathbf{x} = K^*\mathbf{y}$, because in that case AA^* is of lower dimension than A^*A. Another important advantage is that CGNE minimizes the error in the 2-norm:

$$||\mathbf{x} - \mathbf{x}_k||^2_{AA^*} = (A(\mathbf{z} - \mathbf{z}_k))^*(A(\mathbf{z} - \mathbf{z}_k)) = (\mathbf{x} - \mathbf{x}_k)^*(\mathbf{x} - \mathbf{x}_k). \tag{6.29}$$

Unfortunately, this method only works for consistent problems, rendering it useless for most problems in practice due to the presence of noise. However, this issue is solved by regularizing, because regularization leads to a well-posed problem. The standard CG method can be applied to Equation (6.9) by replacing K by $AR^{-1}A^* + \lambda C$, $\mathbf{u}$ by $\mathbf{z}$ and $\mathbf{f}$ by $\mathbf{y}$. Next, we set $\mathbf{x}_k = R^{-1}A^*\mathbf{z}_k$ and $\mathbf{r}_k = \lambda\mathbf{z}_k$ and we define $\mathbf{q}_k := A^*\mathbf{p}_k$, yielding the CGNE algorithm, given as Algorithm 3. In every iteration, the CGNE method minimizes

$$\begin{aligned}(\mathbf{z} - \mathbf{z}_k)^*(AR^{-1}A^* + \lambda C)(\mathbf{z} - \mathbf{z}_k) &= (\mathbf{x} - \mathbf{x}_k)^*R(\mathbf{x} - \mathbf{x}_k) + \frac{1}{\lambda}(\mathbf{r} - \mathbf{r}_k)^*C(\mathbf{r} - \mathbf{r}_k) \\ &= ||\mathbf{e}_k||^2_R + \frac{1}{\lambda}||\mathbf{r} - \mathbf{r}_k||^2_C \end{aligned} \tag{6.30}$$

over the Krylov subspace $\mathbf{z}_0 + \mathcal{K}_k(AR^{-1}A^* + \lambda C, \mathbf{p}_0)$. So CGNE minimizes the same expression as CGLS, but the Krylov subspace is different. As far as we know, the CGNE method has not been applied to minimization problem (6.3) before in this way.

Algorithm 3 CGNE

Require: $A \in \mathbb{C}^{M\times N}, C \in \mathbb{C}^{M\times M}, R \in \mathbb{C}^{N\times N}, \mathbf{x}_0 \in \mathbb{C}^N, \mathbf{y} \in \mathbb{C}^M, \lambda \in \mathbb{R} > 0$;
Ensure: Approximate solution $\mathbf{x}_k$ such that $\|\mathbf{y} - A\mathbf{x}_k - C\mathbf{r}_k\| \leqslant TOL$.

if $\lambda > 0$ **then**
  $\mathbf{x}_0 = \frac{1}{\lambda}R^{-1}A^*\mathbf{r}_0$
else
  $\mathbf{x}_0 = \mathbf{0}$, $\mathbf{r}_0 = \mathbf{0}$
end if
$\mathbf{s}_0 = \mathbf{y} - A\mathbf{x}_0 - C\mathbf{r}_0$, $\mathbf{p}_0 = \mathbf{s}_0$; $\mathbf{q}_0 = A^*\mathbf{p}_0$, $\gamma_0 = \mathbf{s}_0^*\mathbf{s}_0$, $k = 0$
while $\sqrt{\gamma_k} > TOL$ **and** $k < k_{max}$ **do**
  $\xi_k = \mathbf{q}_k^*R^{-1}\mathbf{q}_k + \lambda\mathbf{p}_k^*C\mathbf{p}_k$
  $\alpha_k = \frac{\gamma_k}{\xi_k}$
  $\mathbf{r}_{k+1} = \mathbf{r}_k + \lambda\alpha_k\mathbf{p}_k$;
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k R^{-1}\mathbf{q}_k$;
  $\mathbf{s}_{k+1} = \mathbf{s}_k - \alpha_k(AR^{-1}\mathbf{q}_k + \lambda C\mathbf{p}_k)$
  $\gamma_{k+1} = \mathbf{s}_{k+1}^*\mathbf{s}_{k+1}$
  $\beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
  $\mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k\mathbf{p}_k$
  $\mathbf{q}_{k+1} = A^*\mathbf{p}_{k+1}$
  $k = k + 1$
end while

6.5 Alternating Directions Method of Multipliers

When total variation regularization is used, the CG methods described above cannot be used directly, due to the use of an ℓ_1-penalty. We will resort to the Alternating Directions Method of Multipliers (ADMM) to solve minimization problems of the form

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2 + \frac{\lambda}{2}||F\mathbf{x}||_1. \tag{6.31}$$

This section is based on [16]. By defining

$$h(\mathbf{x}) := \frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2, \quad g(F\mathbf{x}) := \frac{\lambda}{2}||F\mathbf{x}||_1, \tag{6.32}$$

minimization problem (6.31) can be written as

$$\min_{\mathbf{x}} h(\mathbf{x}) + g(F\mathbf{x}), \tag{6.33}$$

which is in turn equivalent to

$$\min_{\mathbf{x},\mathbf{v}} h(\mathbf{x}) + g(\mathbf{v}), \tag{6.34}$$

$$\text{subject to } F\mathbf{x} = \mathbf{v}. \tag{6.35}$$

The reason for splitting the objective function into two separate functions is that the two resulting functions are straightforward to minimize on their own, as opposed to when they are combined. In order to enforce the constraint $F\mathbf{x} = \mathbf{v}$, a penalty term is added to the objective function:

$$\min_{\mathbf{x},\mathbf{v}} h(\mathbf{x}) + g(\mathbf{v}) + \frac{a}{2}||F\mathbf{x} - \mathbf{v}||^2, \tag{6.36}$$

where a is a constant that determines the gain of the penalty term. When a is very large, the constraint is strictly enforced. However, no matter how large a is, it is very unlikely that $F\mathbf{x}$ and $\mathbf{v}$ are exactly equal, so the minimizer of (6.36) will not be exactly equal to the minimizer of the objective function. In order to fix this, an additional variable $\mathbf{u}$ is introduced:

$$\min_{\mathbf{x},\mathbf{v}} h(\mathbf{x}) + g(\mathbf{v}) + \frac{a}{2}||F\mathbf{x} - \mathbf{v} + \mathbf{u}||^2. \tag{6.37}$$

So $\mathbf{u}$ must be chosen such that the constraint $F\mathbf{x} = \mathbf{v}$ is met. Equation (6.37) is known as the augmented Lagrangian for our constrained optimization problem. The correct value of $\mathbf{u}$ can be determined using the following simple augmented Lagrangian algorithm:

Algorithm 4 Split-Variable Augmented Lagrangian Algorithm

1: $\mathbf{u} = \mathbf{0}$
2: $\hat{\mathbf{v}} = \mathbf{0}$
3: **while** convergence has not been reached **do**
4: $\quad (\hat{\mathbf{x}}, \hat{\mathbf{v}}) = \arg\min_{\mathbf{x},\mathbf{v}} h(\mathbf{x}) + g(\mathbf{v}) + \frac{a}{2}||F\mathbf{x} - \mathbf{v} + \mathbf{u}||_2^2$
5: $\quad \mathbf{u} = \mathbf{u} + (F\hat{\mathbf{x}} - \hat{\mathbf{v}})$
6: **end while**

Instead of performing the optimization over $\mathbf{x}$ and $\mathbf{v}$ at the same time, it will be carried out in two steps, as shown in Algorithm 5.

Algorithm 5 Split-Variable ADMM

1: $\mathbf{u} = \mathbf{0}$
2: $\hat{\mathbf{v}} = \mathbf{0}$
3: **while** convergence has not been reached **do**
4: $\quad \hat{\mathbf{x}} = \arg\min_{\mathbf{x}} h(\mathbf{x}) + \frac{a}{2}||F\mathbf{x} - \hat{\mathbf{v}} + \mathbf{u}||_2^2$
5: $\quad \hat{\mathbf{v}} = \arg\min_{\mathbf{v}} g(\mathbf{v}) + \frac{a}{2}||F\hat{\mathbf{x}} - \mathbf{v} + \mathbf{u}||_2^2$
6: $\quad \mathbf{u} = \mathbf{u} + (F\hat{\mathbf{x}} - \hat{\mathbf{v}})$
7: **end while**

In Algorithm 5, the second minimization problem is:

$$\min_{\mathbf{v}} \frac{\lambda}{2}||\mathbf{v}||_1 + \frac{a}{2}||F\hat{\mathbf{x}} - \mathbf{v} + \mathbf{u}||^2. \tag{6.38}$$

The shrinkage function is defined as

$$S_\tau(z) = \text{sign}(z)\max\{|z| - \tau, 0\}. \tag{6.39}$$

It can be shown [16] that Equation (6.38) is minimized by

$$\hat{\mathbf{v}} = S_{\frac{\lambda}{a}}(F\hat{\mathbf{x}} + \mathbf{u}). \tag{6.40}$$

The first minimization problem is

$$\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - A\mathbf{x}||_2^2 + \frac{a}{2}||F\mathbf{x} - \hat{\mathbf{v}} + \mathbf{u}||_2^2. \tag{6.41}$$

Taking the derivative with respect to $\mathbf{x}$ yields a sufficient condition for optimality:

$$(A^*A + aF^*F)\mathbf{x} = A^*\mathbf{y} + aF^*(\mathbf{v} - \mathbf{u}), \tag{6.42}$$

which can be solved using a conjugate gradient method. The CGLS algorithm described in section 6.3 can be altered a bit to accommodate this kind of problem, as shown in Appendix A.

Chapter 7

CGLS vs CGNE: a performance comparison

In order to compare the performances of CGLS and CGNE, both methods were applied to several problems. Due to its simplicity, Tikhonov regularization was used. The Shepp-Logan phantom was chosen as the object to be imaged. This phantom was created by Shepp and Logan in 1974 [21] as a tool for image reconstruction simulations of the head and brain for 2D computerized tomography image reconstruction. Ten ellipses of varying size and signal intensity are used to mimic the geometric and x-ray attenuation properties of the head. In [22], the phantom was adapted to MR physics. They created T_1-weighted, T_2-weighted and spin density-weighted phantoms. The approach outlined in their paper was used to generate our phantom (which is spin density-weighted). For the overdetermined system, a 64×64 phantom was used, as shown in Figure 8.7. Equation (3.10) was used to generate the signal we would expect to obtain from the phantom. These simulations were carried out using the simulated magnetic field shown in Figure 3.6.

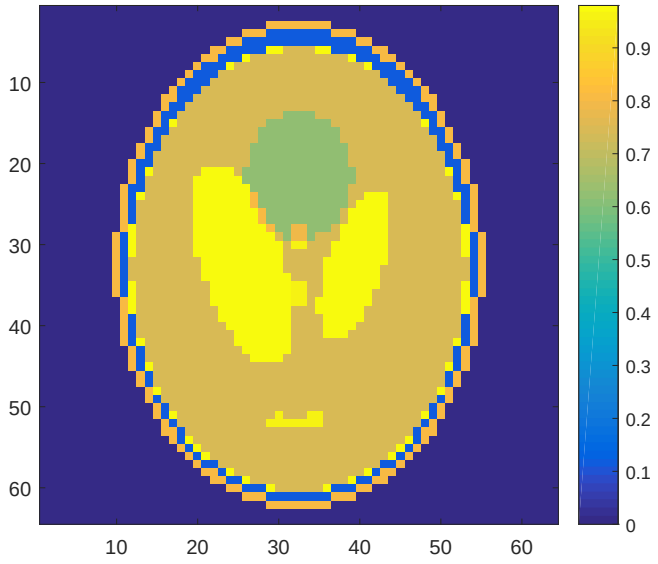


Figure 7.1: The HR image.

The Field of View was set to 10 cm. Due to the large inhomogeneities in the field, we

assumed that T_2^* does not depend on T_2 and is inversely proportional to the field inhomogeneity in each voxel:

$$\frac{1}{T_2^*} = \gamma \Delta B_0. \tag{7.1}$$

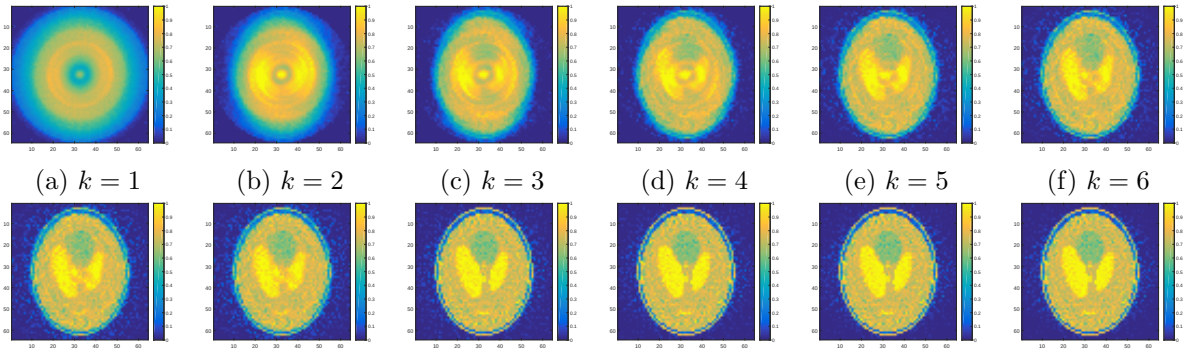
The noise was generated using Equation (3.15). The resistance was set to 1Ω, the temperature to room temperature (293 K) and the bandwidth to 20 kHz, leading to a standard deviation of 20 nV.

7.1 Noiseless scenario

The first scenario that was evaluated was a noiseless one, which means that the system $W\mathbf{x} = \mathbf{s}$ is consistent. 130 pulses were necessary to excite all spins and 36 angles were examined. For each pulse, only one measurement was done. The number of pixels was set to 64×64, leading to a system of 4680 equations and 4096 unknowns.

7.1.1 Without regularization

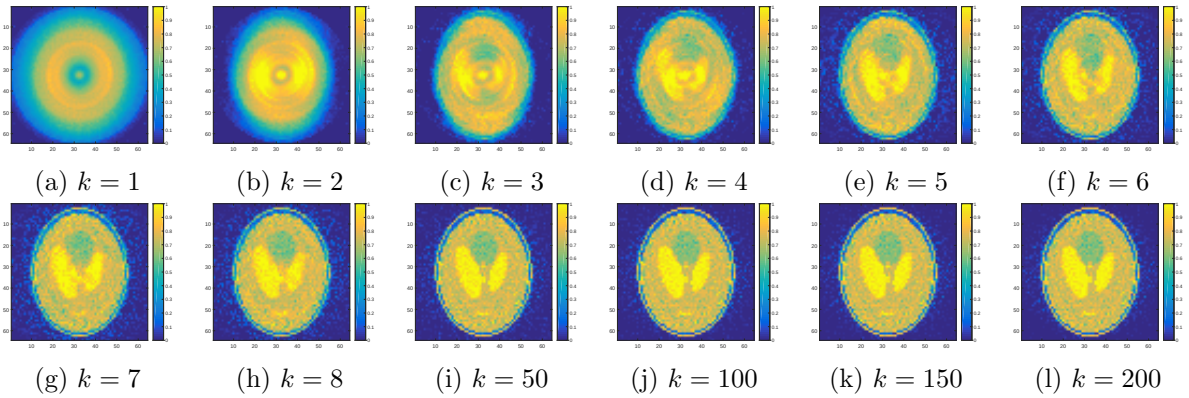
For CGLS, $\mathbf{x}_0 = \mathbf{0}$ was chosen as the initial guess. For CGNE, $\mathbf{r}_0$ was also set to $\mathbf{0}$. The iterates are shown in Figures 7.2 and 7.3. We see that in this case, the CGLS and CGNE iterates look very much alike. In CGLS, the Krylov subspace employed is $\mathcal{K}(W^*W, W^*\mathbf{s})$. In CGNE it is $\mathcal{K}(WW^*, \mathbf{s})$. Multiplying on the left by W^* shows that this is the same subspace as in the CGLS case. Because CGNE is an error-minimizing method, whereas CGLS minimizes the residual, and they operate in the same subspace, all CGNE iterates are closer to the solution than the corresponding CGLS iterates. This is consistent with the plot of the errors shown in Figure 7.4: we see that CGNE yields slightly smaller errors in each iteration than CGLS. For both methods, the average time per iteration was 0.3 ms.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 50$ (j) $k = 100$ (k) $k = 150$ (l) $k = 200$

Figure 7.2: CGLS iterates in a noiseless scenario.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 50$ (j) $k = 100$ (k) $k = 150$ (l) $k = 200$

Figure 7.3: CGNE iterates in a noiseless scenario.

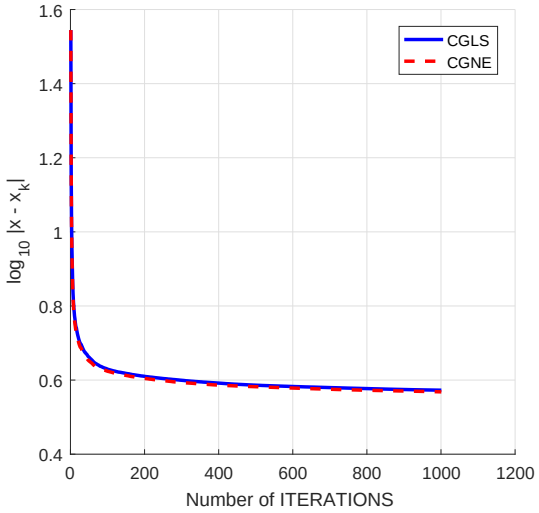
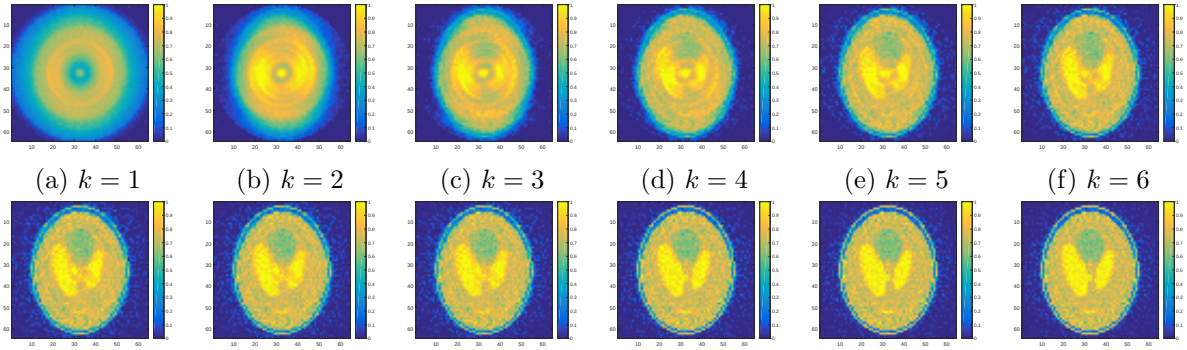


Figure 7.4: Errors of the CGLS and CGNE iterates as a function of the iteration number.

7.1.2 $R = I$

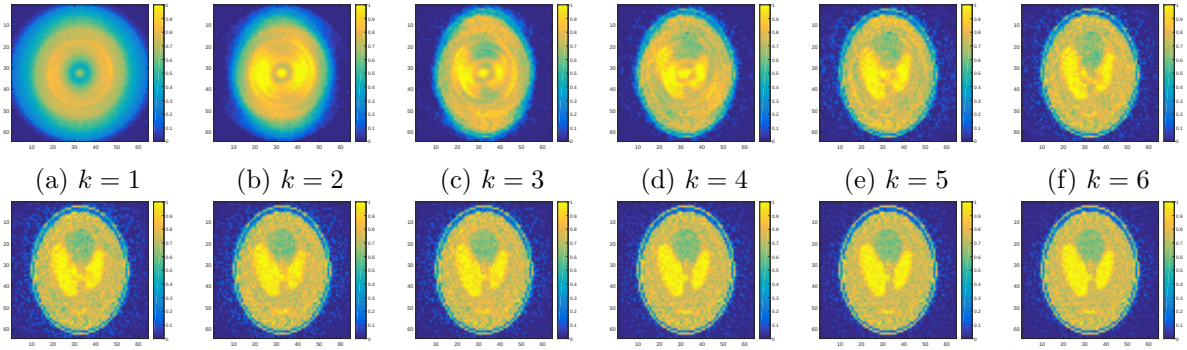
First, we will use $R = I$ as our regularization matrix. We also have $C = I$. This allows for a fair comparison between the two methods, because CGLS uses R and C^{-1}, while CGNE uses R^{-1} and C. When regularization (a small value was chosen: $\lambda = 1 \cdot 10^{-15}$) is applied in this case, the behavior of the two methods is similar to what we have seen before. CGNE yields the iterates with the smallest error, as can be seen in Figure 7.4. This is consistent with the fact that both methods are operating in the same Krylov subspace. Again, the average time per iteration was 0.3 ms for both methods.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure 7.5: CGLS iterates in a noiseless scenario, with regularization ($R = C = I$, $\lambda = 1 \cdot 10^{-15}$).



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure 7.6: CGNE iterates in a noiseless scenario, with regularization ($R = C = I$, $\lambda = 1 \cdot 10^{-15}$).

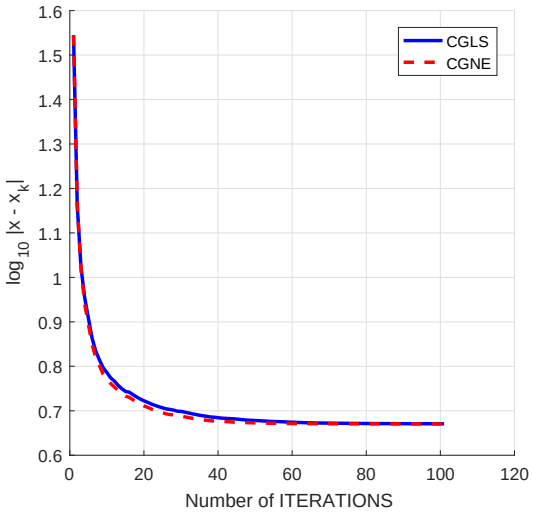
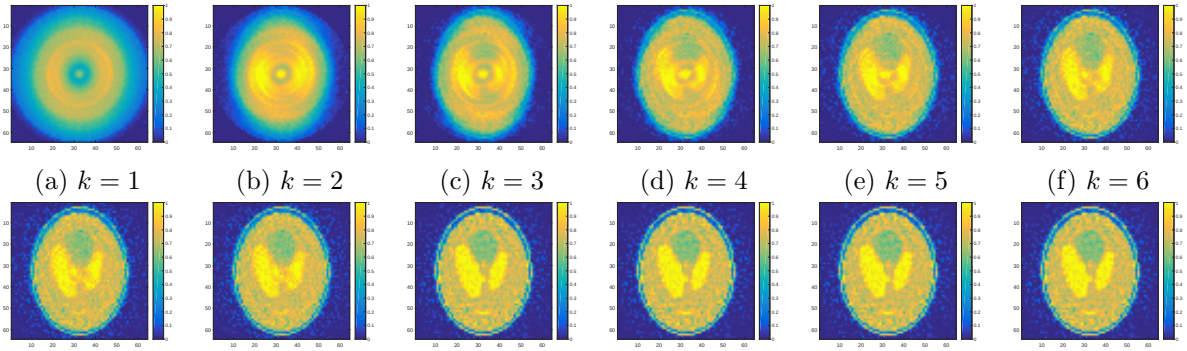


Figure 7.7: Errors of the CGLS and CGNE iterates in the case of a consistent system that is regularized with $R = C = I$ and $\lambda = 1 \cdot 10^{-15}$.

7.1.3 $R = F^*F$

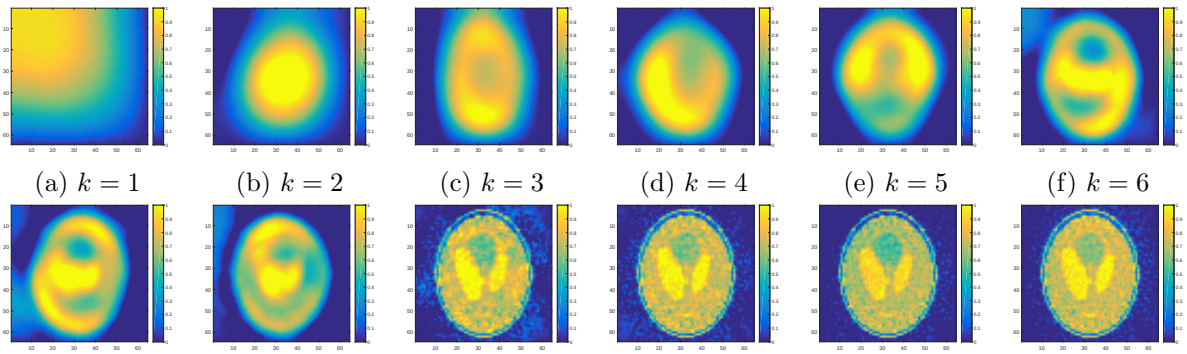
Next, the regularization matrix was set to $R = F^*F$, where F is a first-order difference matrix that calculates the difference between all neighboring pixels. In chapters 8 and 9, when our sole purpose is to produce images of the best possible quality, we will only consider this matrix. Figures 7.8 and 7.9 show the iterates in this case. The errors are plotted in Figure 7.10. In this case, CGLS clearly performs better than CGNE because it converges much more rapidly than CGNE. For CGLS, no clear visible differences can be seen between iterates 8 and 60, while CGNE needs more than 200 iterations to obtain an acceptable solution. This can be explained by the condition numbers of their respective matrices. CGLS operates on the matrix $A^*C^{-1}A + \lambda R$, which has a condition number of $2.6 \cdot 10^3$, while the matrix used by CGNE, $AR^{-1}A^* + \lambda C$, has a condition number of $2.8 \cdot 10^6$. A CGLS iteration took 0.6 ms on average, a CGNE iteration 4.4 ms. The difference can be explained by the fact that in CGNE, the calculations involve R^{-1}. Due to the sparsity structure of R, using a Cholesky decomposition for calculations with R^{-1} did not speed up the algorithm.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 50$ (j) $k = 100$ (k) $k = 150$ (l) $k = 200$

Figure 7.8: CGLS iterates in a noiseless scenario, with regularization ($R = F^*F$, $C = I$, $\lambda = 1 \cdot 10^{-15}$).



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 50$ (j) $k = 100$ (k) $k = 150$ (l) $k = 200$

Figure 7.9: CGNE iterates in a noiseless scenario, with regularization ($R = F^*F$, $C = I$, $\lambda = 1 \cdot 10^{-15}$).

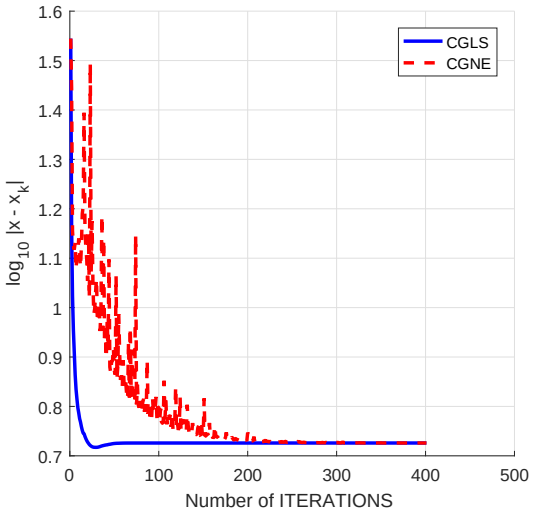


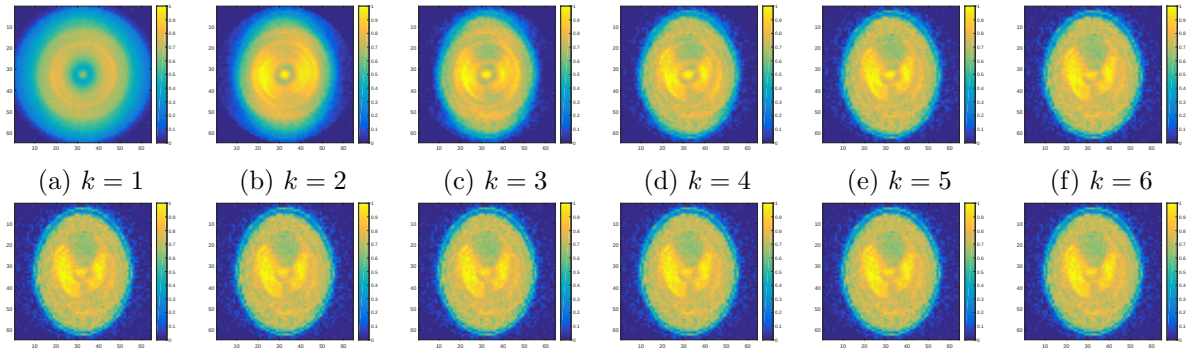
Figure 7.10: Errors of the CGLS and CGNE iterates in case $R = F^*F$. λ was set to $1 \cdot 10^{-15}$.

7.2 Overdetermined system with noise

The recording time was taken to be 1 ms, with a sampling rate of 20 kHz. Again, 130 pulses were necessary to excite all spins and 36 angles were examined. The number of pixels was set to 64×64, leading to a system of 93600 equations and 4096 unknowns.

7.2.1 $R = I$

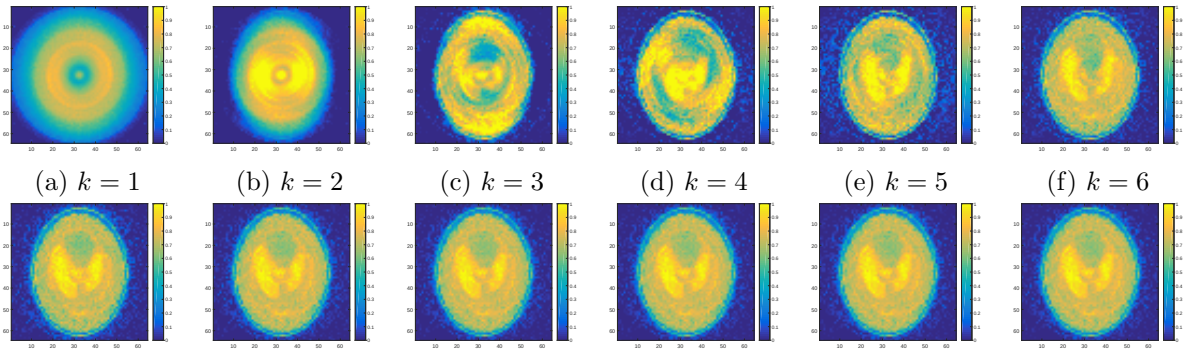
First, the regularization matrix was set to $R = I$. Figures 7.11 and 7.12 show the CGLS and CGNE iterates for different values of the iteration number k, respectively. The regularization parameter was set to $7 \cdot 10^{-14}$. For a smaller value of the regularization parameter, $\lambda = 1 \cdot 10^{-15}$, the iterates are shown in Figures 7.13 and 7.14.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

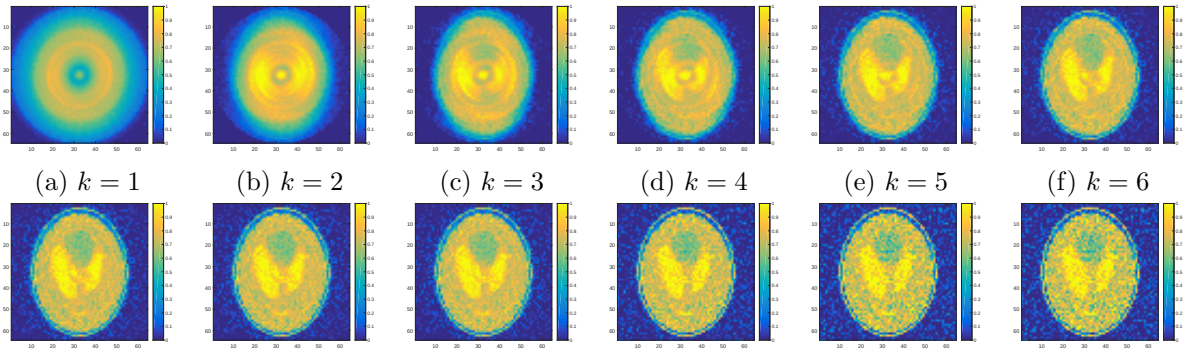
Figure 7.11: CGLS iterates for $\lambda = 7 \cdot 10^{-14}$ in the case of an overdetermined system with $R = I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

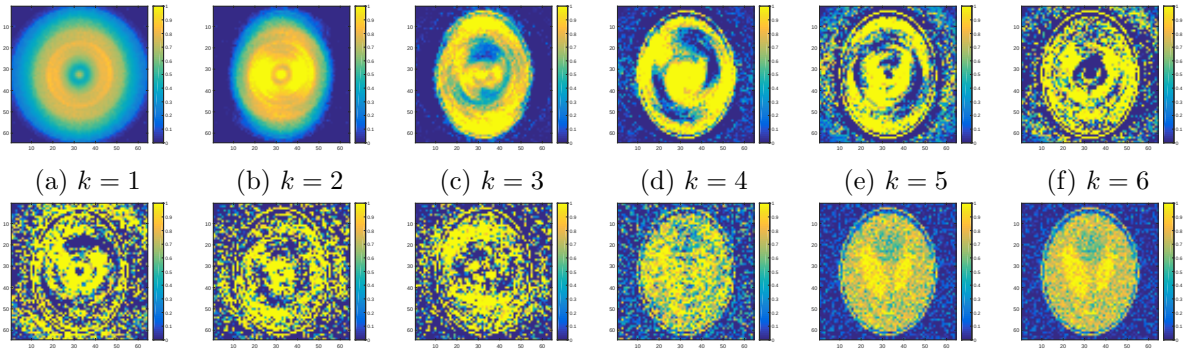
Figure 7.12: CGNE iterates for $\lambda = 7 \cdot 10^{-14}$ in the case of an overdetermined system with $R = I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure 7.13: CGLS iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an overdetermined system with $R = I$.



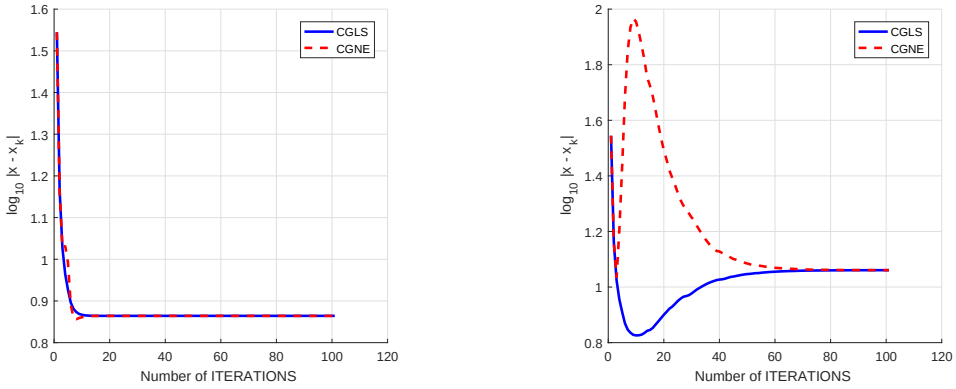
(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure 7.14: CGNE iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an overdetermined system with $R = I$.

As expected, CGLS and CGNE yield the same final image. In the case of a strong regularizing term, it is clear that both methods converge very rapidly, only a few iterations are needed. It is interesting to note that the iterate with the smallest error is obtained by CGNE. For $\lambda = 1 \cdot 10^{-15}$, at first sight it seems that CGLS converges faster than CGNE. However, when we look closely at

the iterates in Figure 7.13, the background in the image formed by iterate 20 is different from the background in iterate 40. So both methods need about the same number of iterations to reach convergence. This is consistent with the fact that $A^*A+\lambda I$ (used by CGLS) and $AA^*+\lambda I$ (used by CGNE) have the same condition number. It is interesting to see that when λ is relatively large, the two methods obtain iterates that are very similar, but this is not the case when λ is small. In that case, the CGLS iterates begin to resemble the final solution very quickly, and after that, the noise starts to contribute more and more to the image, until convergence is reached. For CGNE that is not true, the first iterates are severely impacted by the noise, but they look progressively better as the iteration number increases. These observations are consistent with what we see in Figure 7.15: the CGLS solution is better after 10 iterations than after 50, while this does not hold for the CGNE case. CGLS and CGNE needed 0.013 and 0.014 seconds per iteration respectively.

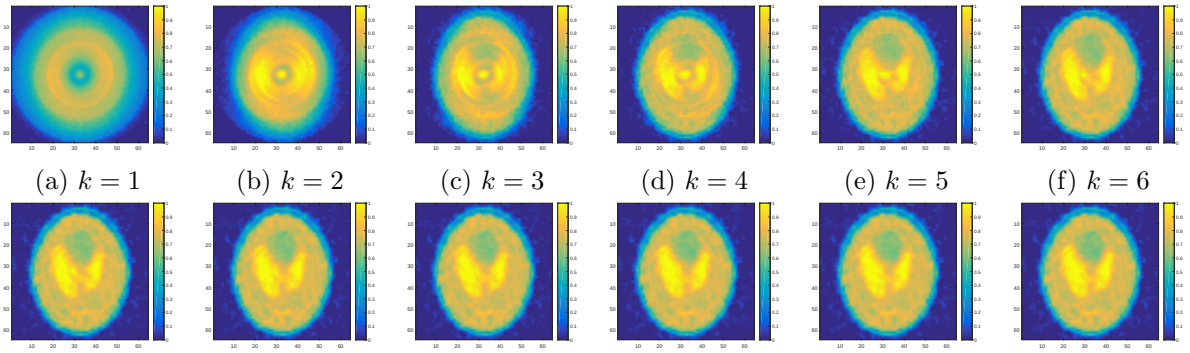


(a) $\lambda = 7 \cdot 10^{-14}$ (b) $\lambda = 1 \cdot 10^{-15}$

Figure 7.15: Plots of the errors of the CGLS and CGNE iterates for two different values of the regularization parameter. Here, $R = I$.

7.2.2 $R = F^*F$

The regularization matrix was set to $R = F^*F$, where F is a first-order difference matrix that calculates the difference between all neighboring pixels. Two different regularization parameters were considered, $\lambda = 7 \cdot 10^{-14}$ and $\lambda = 1 \cdot 10^{-15}$. The former has a strong regularizing effect, the latter does not.

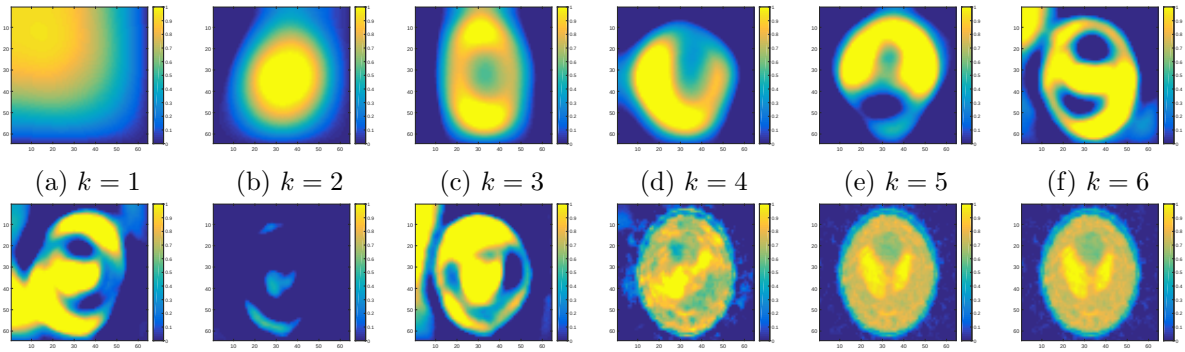


(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure 7.16: CGLS iterates for $\lambda = 7 \cdot 10^{-14}$ in the case of an overdetermined system with $R = F^*F$.

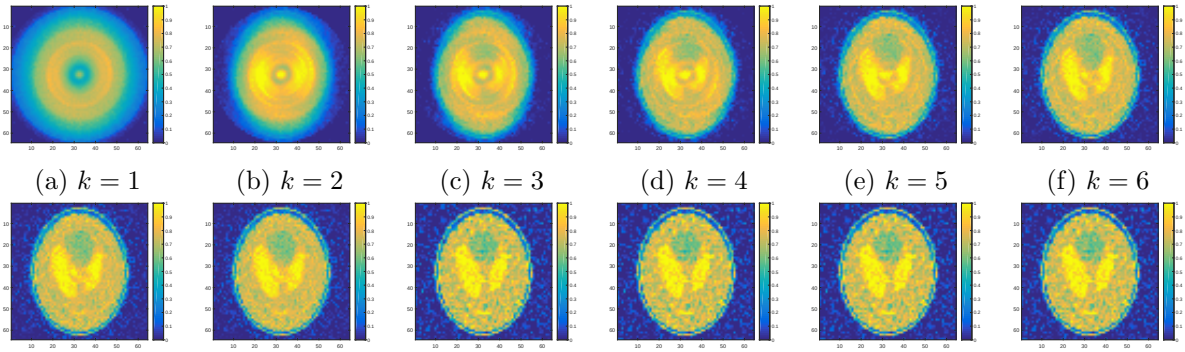
Figures 7.16 and 7.17 show the iterates $\mathbf{x}_k$ for different values of the iteration number k for $\lambda = 7 \cdot 10^{-14}$ in the CGLS and the CGNE algorithm, respectively. It is clear that the CGLS algorithm converges much more rapidly than CGNE. For CGLS, no clear visible differences can be seen from iterate 8 on, while CGNE needs more than 20 iterations to obtain an acceptable solution. For the smaller regularization parameter $\lambda = 1 \cdot 10^{-15}$, the iterates are shown in Figures 7.18 and 7.19. It looks like CGLS needs only a few iterations to converge to an acceptable solution.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

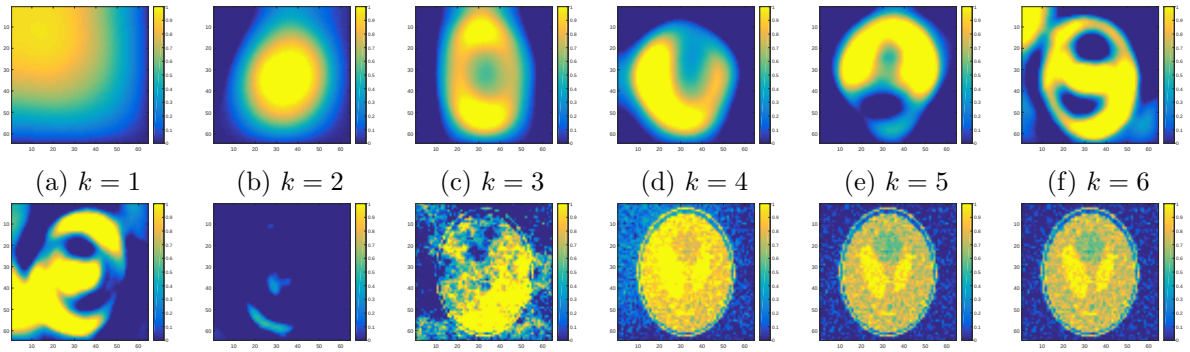
Figure 7.17: CGNE iterates for $\lambda = 7 \cdot 10^{-14}$ in the case of an overdetermined system with $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 100$ (j) $k = 200$ (k) $k = 300$ (l) $k = 400$

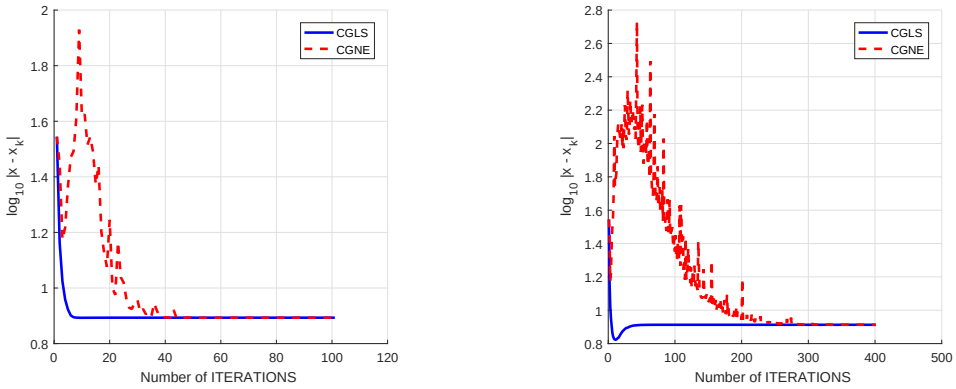
Figure 7.18: CGLS iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an overdetermined system with $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 100$ (j) $k = 200$ (k) $k = 300$ (l) $k = 400$

Figure 7.19: CGNE iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an overdetermined system with $R = F^*F$.



(a) $\lambda = 7 \cdot 10^{-14}$ (b) $\lambda = 1 \cdot 10^{-15}$

Figure 7.20: Plots of the errors of the CGLS and CGNE iterates for two different values of the regularization parameter. Here, $R = FF^*$.

We have seen that CGNE and CGLS yield the same final image, but it takes much longer for CGNE to converge to the optimal solution, especially when the regularization parameter is small. On average, a CGLS iteration took 0.013 seconds, while a CGNE iteration took 0.018 seconds. Figure 7.20 shows the value of the error for each iteration. We see that for $\lambda = 7 \cdot 10^{-14}$, CGNE needs about 30 iterations to reach the same error as CGLS, while for $\lambda = 1 \cdot 10^{-15}$, about 300 iterations are needed. For CGLS, only a few iterations are needed to converge when $\lambda = 7 \cdot 10^{-14}$. For the smaller regularization parameter, the solution has converged after approximately 30 iterations. However, after the first initial iterations, the difference between iterates are very small, so the solution already looks acceptable after a small number of iterations. This is not the case for CGNE. There, the iterates start looking acceptable only when the solution is nearly converged.

7.3 Underdetermined system with noise

In order to obtain an underdetermined system, the recording time was set to $5 \cdot 10^{-5}$ s and the number of pixels to 256×256. All other parameters remained the same, leading to a system of 4680 equations and 65536 unknowns. Again, two different regularization parameters were

considered: $\lambda = 5 \cdot 10^{-14}$ and $\lambda = 1 \cdot 10^{-15}$. The iterates and other results can be found in Appendix B. The results are very similar to the overdetermined case: for $R = F^*F$, CGLS converges to the optimal solution in just a few iterations and the number of iterations is not influenced by λ much, while CGNE converges more slowly and needs a lot more iterations when the regularization parameter is smaller. A CGLS iteration and a CGNE iteration took 0.008 and 0.11 seconds, respectively. The results for $R = I$ are also consistent with the findings in the overdetermined case. A CGLS iteration took 0.008 seconds, while a CGNE iteration took slightly less: 0.007 seconds.

7.4 Analysis of the results

7.4.1 $R = F^*F$

When $R = F^*F$, CGLS requires a smaller number of iterations to reach convergence. For both CGLS and CGNE, a smaller regularization parameter implies a larger number of iterations. However, this effect is much stronger in CGNE than in CGLS. These findings, combined with the fact that working with R^{-1} and C is much more cumbersome than C^{-1} and R, make CGLS the method of choice in this scenario. Because $R = F^*F$ is the regularization matrix that will be used to reconstruct the images in the next chapters, we will use CGLS for the remainder of this research.

7.4.2 $R = I$

For $R = I$, the number of iterations needed for convergence is approximately the same. In this case, CGLS is working with the matrix $W^*W + \lambda I$, while CGNE is using the matrix $WW^* + \lambda I$. These two matrices have the same condition number. Hence, the two methods should have the same convergence speed, which is consistent with what we have seen. For a consistent system, CGLS is outperformed by CGNE, due to the latter being an error-minimizing method, whereas CGLS minimizes the residual. In the case of an inconsistent system $W\mathbf{x} = \mathbf{s}$, where $\mathbf{s}$ includes noise, CGNE and CGLS demonstrated comparable performances in the case of a strong regularizing term, with CGNE obtaining iterates with smaller errors after the first few iterations. For a small regularization parameter, CGNE was outperformed by CGLS. In that case, the CGLS iterates start approximating the final solution after only a few iterations, while this does not hold for CGNE. The difference in behavior can be explained by different relationships between $\mathbf{x}_k$ and $\mathbf{r}_k$. In CGLS, $\mathbf{x}_k$ is obtained and, using this value, $\mathbf{r}_k$ is calculated as $\mathbf{r}_k = C^{-1}(\mathbf{s} - W\mathbf{x}_k)$. In CGNE, $\mathbf{r}_k$ is obtained first and then $\mathbf{x}_k = \frac{1}{\lambda}R^{-1}W^*\mathbf{r}_k$ is calculated. (Once convergence is reached, both relationships hold.)

7.4.3 CGNE as a viable alternative to CGLS

For consistent problems $W\mathbf{x} = \mathbf{s}$, CGNE performs slightly better than CGLS when there is no regularization and when $R = I$. For inconsistent problems, CGNE and CGLS behave similarly when $R = I$ and the regularization parameter is sufficiently large. When $R = F^*F$, the convergence of CGNE is slow compared to CGLS. However, in cases where the computational cost of doing calculations with C^{-1} is much higher than with R^{-1}, CGNE is a very promising method. Even if CGNE were to need more iterations, the lower computational cost per iteration could potentially compensate for this. This is especially promising for overdetermined systems, because in that case, R is smaller than C, which means that having to execute calculations with C and R^{-1} instead of C^{-1} and R could prove more efficient. And of course, when C is singular or nearly singular, CGLS cannot be used, making CGNE the method of choice.

Chapter 8

Super-resolution: simulation results

Simulations were carried out in order to test the super-resolution reconstruction method described in chapter 4. Two different forms of geometric deformation were examined: translation and rotation. In the translation case, the image was shifted, blurred and then down-sampled. This was done for four different shifts. Then the combined information in these four low resolution images was used to generate one high resolution image. The rotation case is more interesting to us, because we incorporated MRI physics into the model to generate the electrical signals we would expect from the object. Appropriate noise levels were incorporated, in order to test whether super-resolution can yield better results than direct high resolution reconstruction in low-field MRI. Again, the Shepp-Logan phantom was chosen as the object to be imaged. The values of the regularization parameters used in this chapter were chosen such that the error was minimized in the ℓ_1 norm. In this case, that is possible, because we are working with simulated data so the model solution is known. However, this is clearly not an option when the data is not simulated. Employing the L-curve criterion did not yield an optimal regularization parameter, because no L-shape could be detected. All results were obtained using CGLS with a maximum of 10 iterations. When ADMM was used to obtain a total variation regularized solution, a maximum of 10 outer iterations was used.

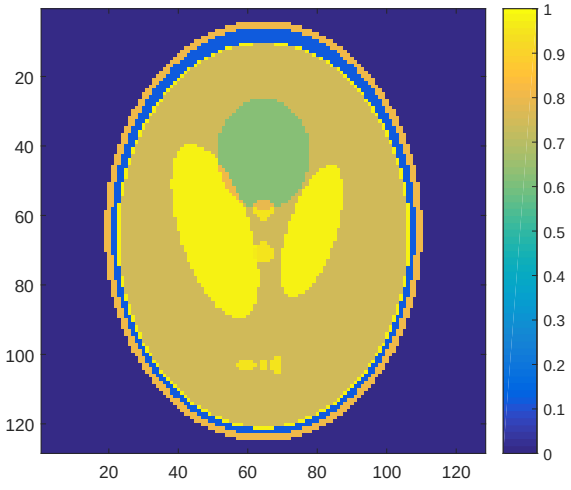


Figure 8.1: Desired HR image (128 × 128 pixels) of the MATLAB Shepp-Logan phantom.

8.1 Translation

The high resolution image and the low resolution images were chosen to be 128 × 128 pixels and 32 × 32 pixels respectively. The desired HR image is shown in Figure 8.1. Following the data acquisition model described in chapter 4, the LR images are obtained by first shifting the HR image by a subpixel amount, subjecting it to blurring, downsampling it and finally, adding noise. The steps in this process are shown in Figure 8.2.

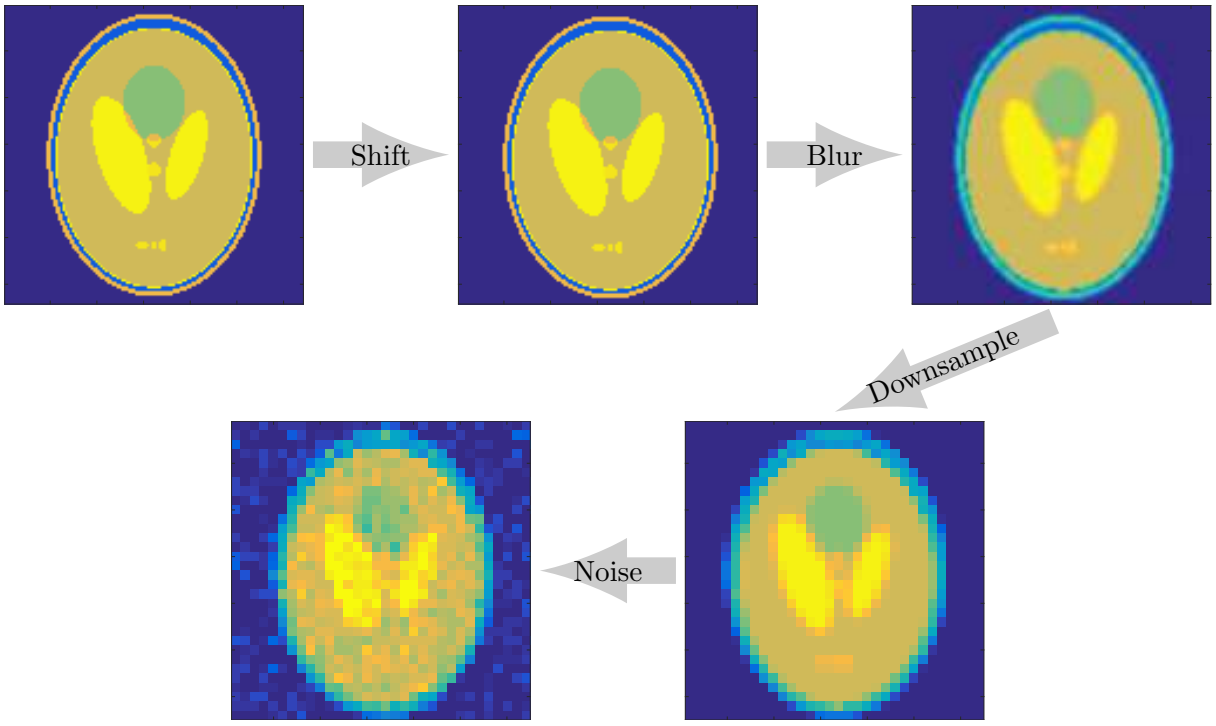
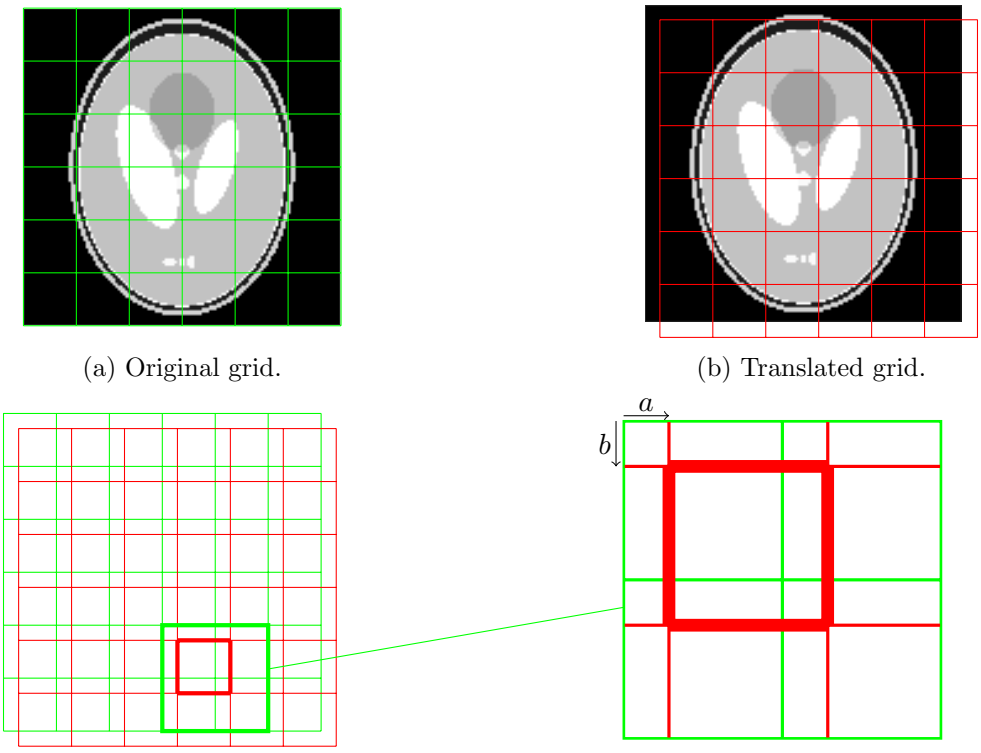


Figure 8.2: From HR to LR.

The grayscale values of the HR image were stored in a matrix X. The first LR image was created without shifting the object. The other 3 LR images were created using three different subpixel shifts: $(0.25, 0.25)$, $(0.5, 0.3)$ and $(0.8, 0.6)$. When $\bar{a} > 0, \bar{b} > 0$, a shift of $(\bar{a}, \bar{b})$ means that the LR grid is shifted to the right by $\bar{a}$ times the size of the LR pixel in the x-direction and towards the bottom by $\bar{b}$ times the size of the LR pixel in the y-direction. So the object itself is shifted to the left and towards the top. Denoting the down-sampling factor in the x-direction by L_1 and in the y-direction by L_2, the image is shifted by $(\bar{a}L_1, \bar{b}L_2)$ HR pixels. The translation of the grid is shown in Figure 8.3. For clarity, the phantom is shown in black and white. It is clear that some pixels are (partially) shifted beyond the borders of the image. Therefore, the original image has to be padded with zeros. In MR images, the areas close to the edges usually have a pixel intensity of zero, so this does not influence the final result. Defining $a := \bar{a}L_1$ and $b := \bar{b}L_2$, we can decompose a and b into an integer part and a remainder part:

$$a = \lfloor a \rfloor + a_{rem}, \quad b = \lfloor b \rfloor + b_{rem}. \tag{8.1}$$



(a) Original grid.

(b) Translated grid.

(c) The translated grid superimposed on the original grid.

(d) One pixel in the translated grid partially overlaps multiple pixels in the original grid.

Figure 8.3: Translation of the pixel grid.

Figures 8.3c and 8.3d show that the pixel value of the shifted image is a weighted average of the value of 4 pixels in the original image. The weights are determined by the area of the overlap between the shifted pixel and the original pixel. Figure 8.3d shows that the value of pixel (i, j) in the shifted image is equal to

$$\begin{aligned}
&(1 - a_{rem})(1 - b_{rem})X_{i-\lfloor a \rfloor, j-\lfloor b \rfloor} \\
&+ a_{rem}(1 - b_{rem})X_{i-\lfloor a \rfloor - 1, j-\lfloor b \rfloor} \\
&+ (1 - a_{rem})b_{rem}X_{i-\lfloor a \rfloor, j-\lfloor b \rfloor - 1} \\
&+ a_{rem}b_{rem}X_{i-\lfloor a \rfloor - 1, j-\lfloor b \rfloor - 1}.
\end{aligned} \tag{8.2}$$

For each of the shifts, this information was stored in matrix G_k such that $G_k\mathbf{x}$ yielded the desired shifted version of the HR image. A Gaussian function with $s_1 = s_2 = 1$ and $\rho = 0$ was used as the point spread function, leading to the blurring matrix B. The down-sampling operator D_k simply takes the average over $L_1L_2 = 16$ pixels. The noise was independent, with standard deviation $\sigma = 0.05$. The LR images are shown in Figure 8.4.

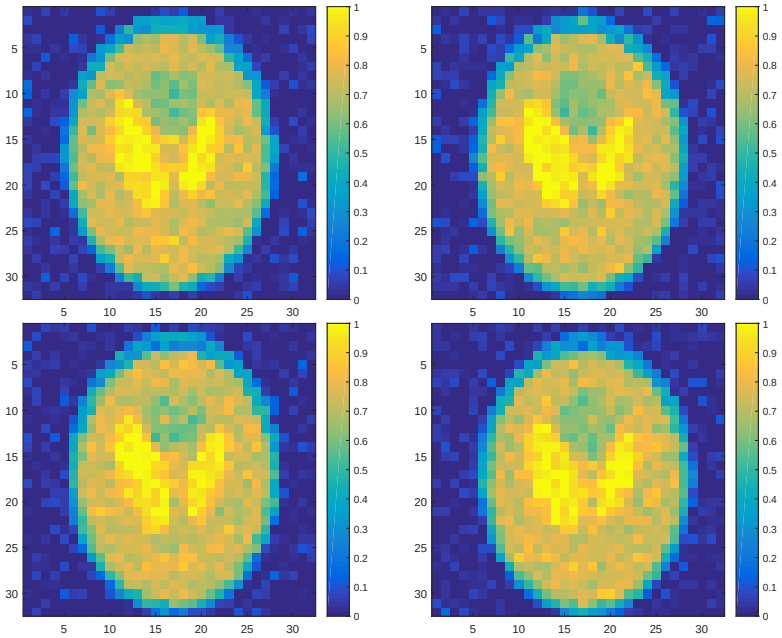


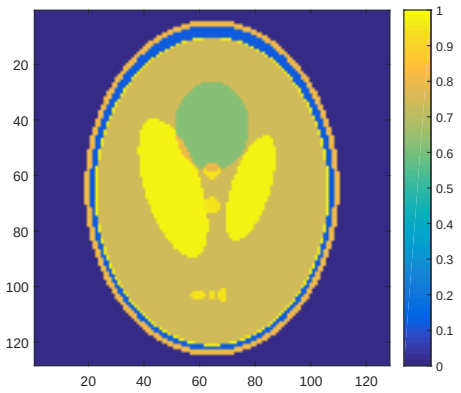
Figure 8.4: Four low resolution images (32 × 32 pixels).

Using these four $\mathbf{y}_k$, minimization problem (5.6) was solved for the three different types of regularization. The matrix A had dimensions 4096×16384. Figure 8.5 shows the phantom image and the three images resulting from the three different kinds of regularization. They all look better than the LR images. However, total variation regularization yields a better result than the other two, it looks less noisy. The image obtained when edge-preserving regularization is used looks better than the Tikhonov image. Obviously, the choice of regularization parameter (as well as a and T) influences the outcome.

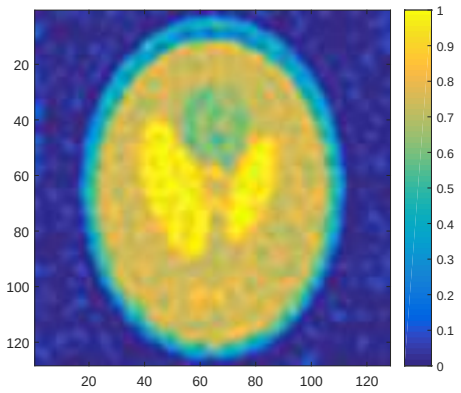
The pixels must have values between 0 and 1, but all the resulting minimizers had values outside of this range. The final images were obtained by projecting onto $[0, 1]$. The ℓ_1-norm and the ℓ_2-norm of the error of each of the three solutions are shown in Table 8.1. As the images suggest, total variation yields the smallest errors and Tikhonov the largest.

Table 8.1: ℓ_1 and ℓ_2 norm of the errors of the three obtained solutions.

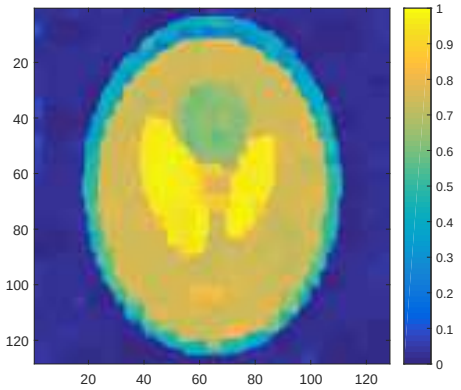
Regularization	Error in ℓ_1 norm	Error in ℓ_2 norm
Tikhonov	1148.9	16.4735
Total variation	964.4	15.8980
Edge-preserving	1074.2	16.1373



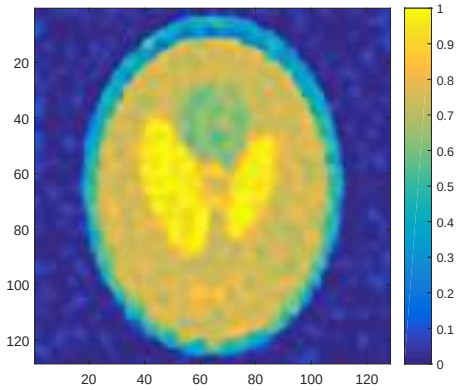
(a) Original image.



(b) Tikhonov regularization.



(c) Total variation regularization.

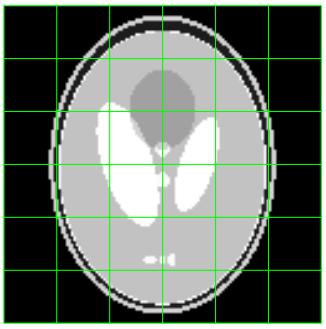


(d) Edge-preserving regularization.

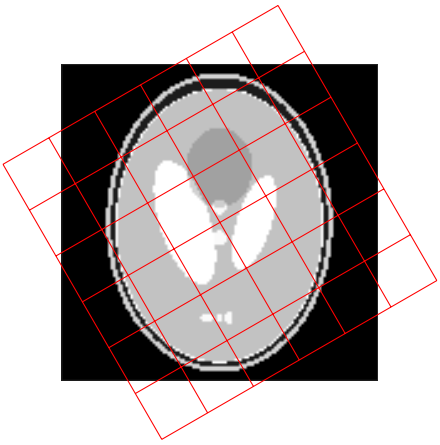
Figure 8.5: The results obtained by using three different kinds of regularization.

8.2 Rotation

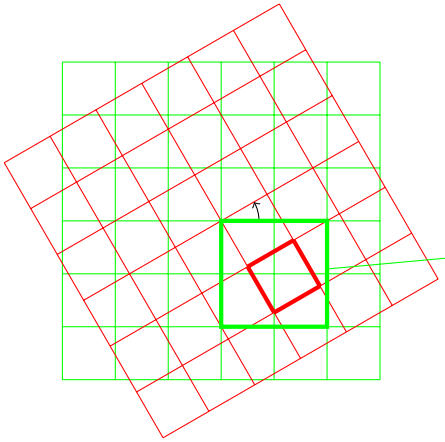
In order to obtain images of the same object, rotated over different angles, either the object itself or the MRI scanner can be rotated. The Halbach array designed in Leiden lends itself well to rotation, so we will assume that the object is fixed and the source of the rotation is the MRI scanner, which would allow us to control the rotation angle accurately. This means that when the scanner is rotated over an angle θ, the pixel grid is rotated over that same angle, as shown in Figure 8.6. Figures 8.6c and 8.6d show that one pixel in the rotated grid partially covers multiple pixels in the original grid. This means that the intensity of the pixel in the rotated grid is a weighted average of the intensity of the pixels in the original grid it coincides with. The weights are equal to the normalized area of the overlaps. These weights are stored in the geometric deformation matrix G_k. In order to compare super-resolution reconstruction to direct high resolution reconstruction, simulations were done using the simulated magnetic field shown in Figure 3.6. A Shepp-Logan phantom of 64×64 pixels, as shown in Figure 8.7, was used as the HR image. Because the LUMC prototype is very small, the Field of View was set to 2 cm. Again, T_2^* was calculated using Equation (7.1).



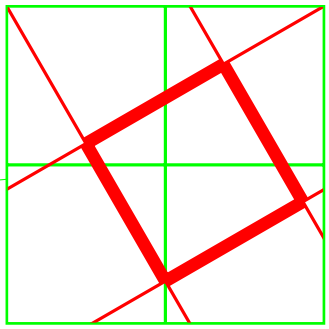
(a) Original grid.



(b) Rotated grid.



(c) The rotated grid superimposed on the original grid.



(d) One pixel in the rotated grid partially overlaps multiple pixels in the original grid.

Figure 8.6: Rotation of the pixel grid.

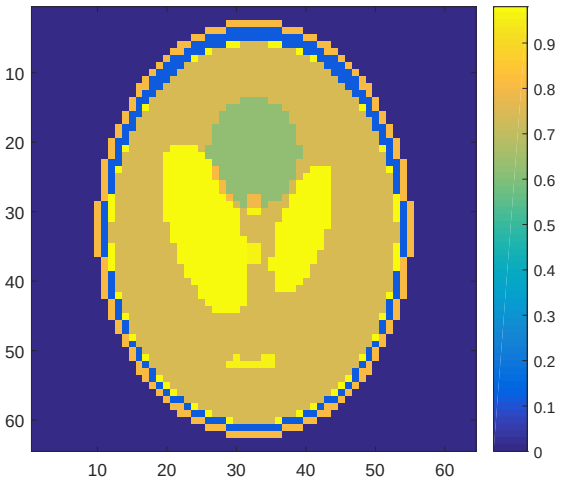
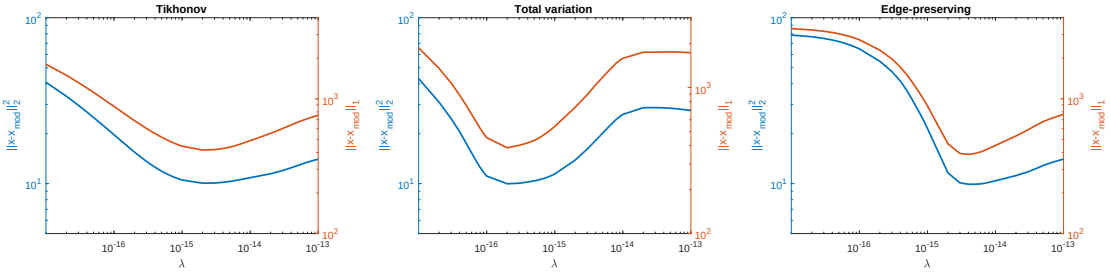


Figure 8.7: The HR image.

As in the previous chapter, the noise was generated using Equation (3.15). The same values were used, leading to a standard deviation of 20 nV. The receiver field value was assumed to be 60 mT/A.

8.2.1 High resolution

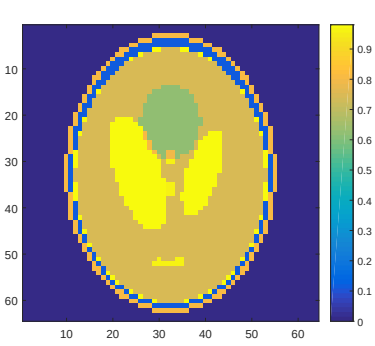
The signal induced by the phantom was generated for 36 different rotations: $\theta = 0, 10, 20, ..., 350°$. We used the simulated magnetic field shown in Figure 3.6. Pulses with a bandwidth of 20 kHz were applied until all frequencies had been excited once, and the noise was assumed to be Gaussian with a zero mean and a standard deviation of 20 nV. The total recording time per pulse was set to 1 ms. The resulting matrix had a size of 93600×4096. The SNR, calculated as the ℓ_2 norm of the signal vector divided by the ℓ_2 norm of the noise vector, was very low: 0.45. For the three types of regularization and for a range of values of the regularization parameter λ, the error in the ℓ_1 and ℓ_2 norm was calculated. The results are shown in Figure 8.8.



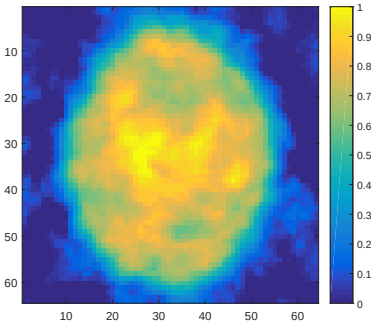
(a) Tikhonov (b) Total variation. (c) Edge-preserving.

Figure 8.8: Errors in ℓ_1 and ℓ_2 norms for the three types of regularization. For total variation regularization, a was set to $a = 10\lambda$. For edge-preserving regularization, T was chosen to be $T = 0.05$.

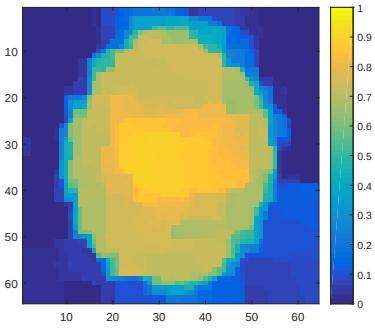
$\lambda = 2 \cdot 10^{-15}$, $\lambda = 2 \cdot 10^{-16}$ and $\lambda = 4 \cdot 10^{-15}$ yield the images that are closest to the original in the ℓ_1 and the ℓ_2 norm for Tikhonov, total variation and edge-preserving regularization respectively. The corresponding results are shown in Figure 8.9. These images are of very poor quality. While Tikhonov and edge-preserving regularization manage to retain some of the main features of the original phantom, like the two large yellow ellipses in the center, total variation regularization yields an image that does not even capture that. So visually speaking, edge-preserving and Tikhonov regularization yield the best results. However, when we look at the errors in the ℓ_1 and ℓ_2 norm, which are shown in Table 8.2, we see that when the ℓ_1 and ℓ_2 norm of the difference between two images is used as a measure of their similarity, the total variation image is closer to the model solution than the Tikhonov image, while total variation and edge-preserving regularization are comparable in terms of performance.



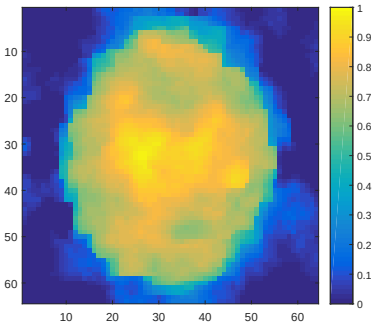
(a) Model solution.



(b) Tikhonov.



(c) Total variation.



(d) Edge-preserving.

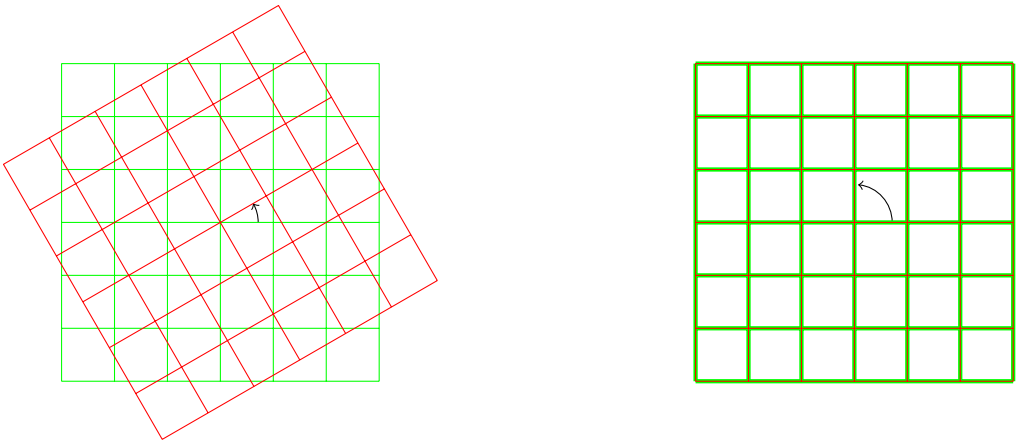
Figure 8.9: The resulting HR images for the three different types of regularization. λ was chosen such that the resulting image was as close to the model solution as possible.

Table 8.2: ℓ_1 and ℓ_2 norm of the errors of the three obtained HR solutions.

Regularization	Error in ℓ_1 norm	Error in ℓ_2 norm
Tikhonov	416.7599	10.0607
Total variation	386.6856	9.9747
Edge-preserving	387.1366	9.9129

8.2.2 Super-resolution

The phantom was down-sampled and an image was formed by using the signals generated for $\theta = 0°, 90°, 180°, 270°$. The next image was formed using the signals generated for $\theta = 10°, 100°, 190°, 280°$. Continuing this, 9 different images were formed of the phantom, such that all the same angles were covered as in the high resolution case. (So in the super-resolution model, the second image depicts a down-sampled version of the phantom rotated over 10°, the third image represents a 20° rotation, etc.) The reason for choosing to create 9 LR images is that rotations over $0°, 10°, ..., 80°$ all yield information about different weighted averages of pixels in the original grid, while a rotation over 90° degrees will simply project a pixel of the new grid on top of one other pixel in the original grid, rendering super-resolution useless. This is illustrated in Figure 8.10.



(a) Rotation over 30°. (b) Rotation over 90°.

Figure 8.10: A rotation over 30° provides a new weighted average of the original pixels, while a rotation over 90° does not.

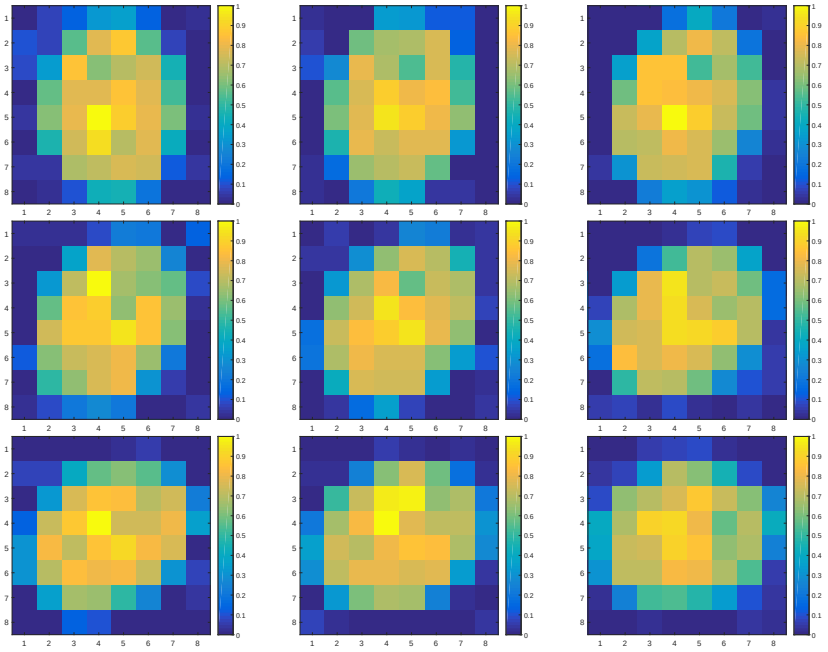
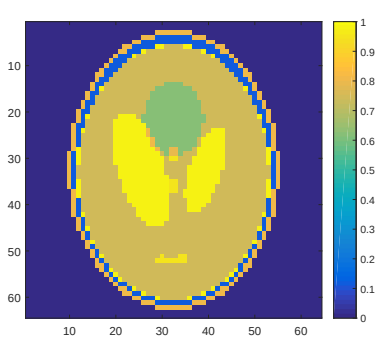


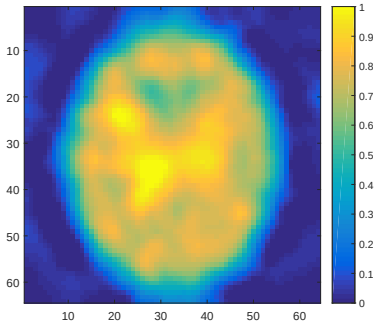
Figure 8.11: The LR images.

Again, the applied bandwidth was 20 kHz, the total pulse length 1 ms and the noise had a standard deviation of 14 nV. Different down-sampling factors L were investigated and the best results were obtained for $L = 8$. So the resulting LR images are 8×8 pixels. The dimensions of the matrices used to calculate the LR results were 10400×1024. Total variation regularization with $\lambda = 2 \cdot 10^{-16}$ and $a = 2 \cdot 10^{-15}$ was used to obtain the LR images, which are shown in Figure 8.11. The SR images obtained from these 9 LR images are shown in Figure 8.12. The matrix used in these calculations was of size 9216×4096. The regularization parameter was chosen such that the ℓ_1 norm was minimized. While their quality is still quite poor, these images clearly capture the main features of the phantom much better. Total variation regularization in particular results in an image where the green circle is clearly visible, whereas in the HR case, total variation yields an image where neither the yellow ellipses nor the green circle can be

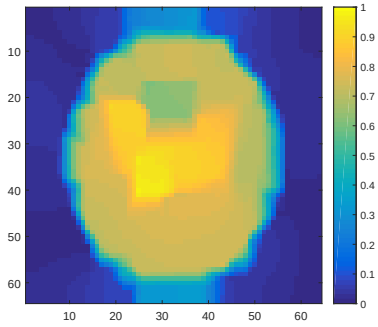
discerned.



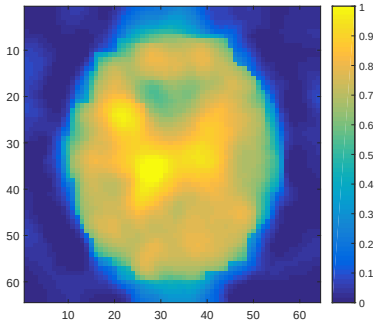
(a) Model solution.



(b) Tikhonov. ($\lambda = 0.04$)



(c) Total variation. ($\lambda = 0.004,\ a = 0.04$)



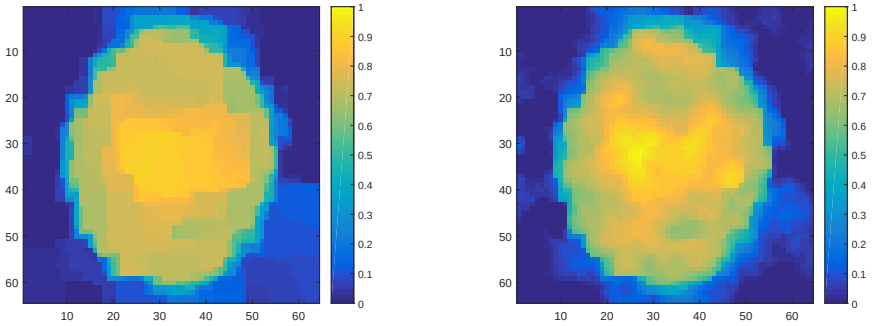
(d) Edge-preserving. ($\lambda = 0.07,\ T = 0.05$)

Figure 8.12: The resulting SR images for the three different types of regularization.

The total variation and the edge-preserving results for HR and SR are shown in one figure, Figure 8.13. From this figure, it is clear that super-resolution yields better images than a direct HR approach. The main feautures of the phantom are more clearly visible and the edges are sharper. In Table 8.3, the errors of the three obtained SR solutions are shown. These errors are smaller than the errors of the HR solutions, which is consistent with our observations when we visually compare the HR and the SR images. It is clear that Tikhonov regularization is outperformed by the other two types of regularization. Therefore, Tikhonov regularization will not be used from now on.

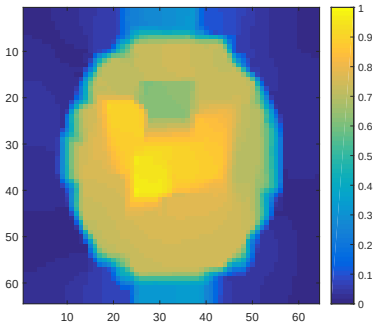
Table 8.3: ℓ_1 and ℓ_2 norm of the errors of the three obtained SR solutions.

Regularization	Error in ℓ_1 norm	Error in ℓ_2 norm
Tikhonov	377.5713	9.6809
Total variation	336.1646	9.7383
Edge-preserving	355.7411	9.6682

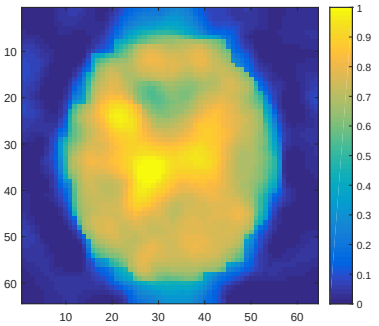


(a) HR: total variation.

(b) HR: edge-preserving.



(c) SR: total variation.

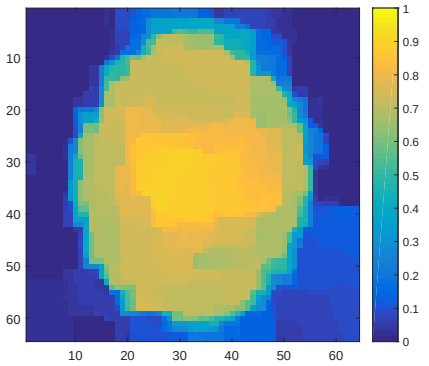


(d) SR: edge-preserving

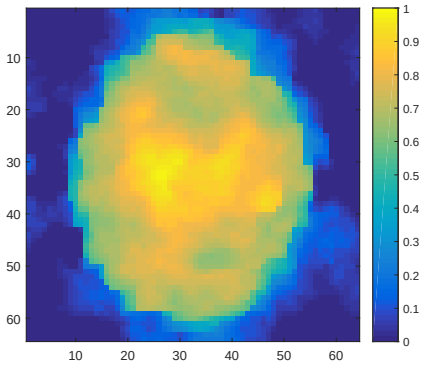
Figure 8.13: HR and SR results using two different types of regularization.

Down-sampling factors $L = 2$ and $L = 4$ yielded results that were less satisfying than $L = 8$. The resulting images are shown in Figure 8.14, alongside the HR result and the SR result with $L = 8$. Especially the images created using a downsampling factor of $L = 2$ are of very poor quality. The $L = 4$ images look better, arguably even better than the HR images, but $L = 8$ clearly yields the best result. These results show that in the case of very noisy signals, super-resolution can be employed to obtain a better final image, due to the improved SNR ratio per pixel in LR images. When the FoV is increased, the signal strength increases too. Figure 8.15 shows the final images when the Field of View is set to 3 cm (all other parameters stay the same). In this case, the SNR was 1.03. SR with $L = 4$ and $L = 8$ both yield clearer images than HR reconstruction. When the FoV is set to 4 cm, the HR images look much better than before (for 2 and 3 cm), as can be seen in Figure 8.16. In this case, the SNR is 1.83. The SR images created using a down-sampling factor of 4 and the HR images are of comparable quality, but the SR images manage to capture the shape of the two large yellow ellipses somewhat better, while the edges of the phantom look sharper in the HR images. $L = 2$ yields very noisy images. Total variation combined with $L = 8$ yields a very blocky image compared to the HR result, while edge-preserving with $L = 8$ yields an image that looks like a blurrier version of the image generated when $L = 4$ is used.

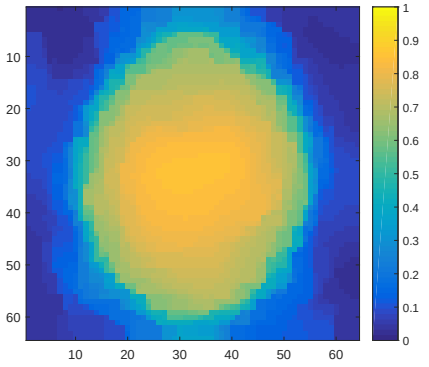
Figure 8.17 shows the results when the FoV is set to 10 cm, and the SNR is 11.42. In this case, the HR images are of much better quality than all the LR images. This is especially clear when looking at the edges of the phantom. When super-resolution is used, it is difficult to capture the different colors close to these edges, whereas in the HR image, these different colors are clearly visible. These findings are consistent with the high signal-to-noise ratio.



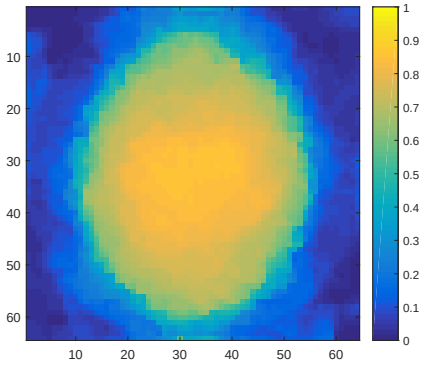
(a) HR: total variation.



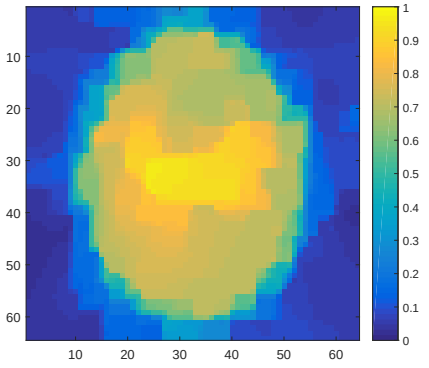
(b) HR: edge-preserving.



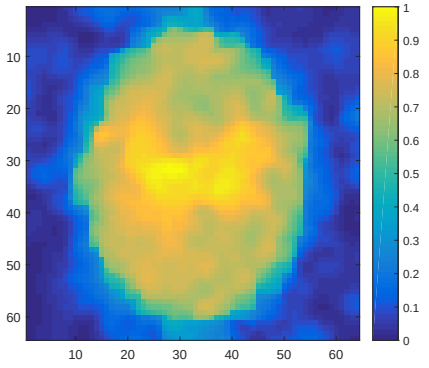
(c) SR with $L = 2$: total variation.



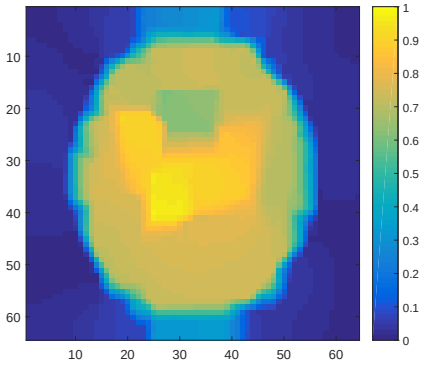
(d) SR with $L = 2$: edge-preserving



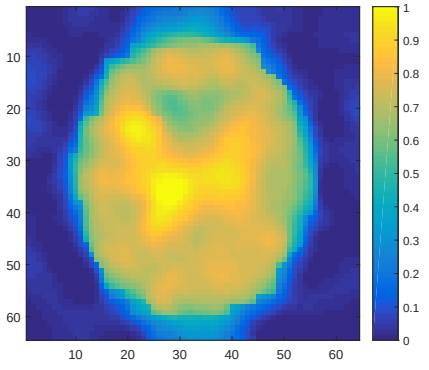
(e) SR with $L = 4$: total variation.



(f) SR with $L = 4$: edge-preserving

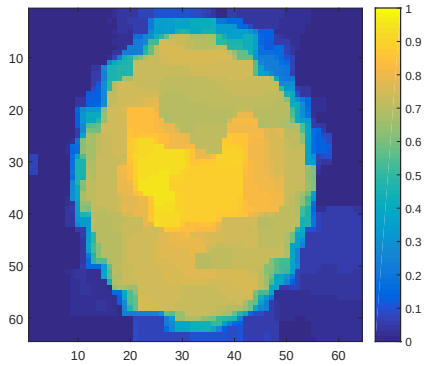


(g) SR with $L = 8$: total variation.

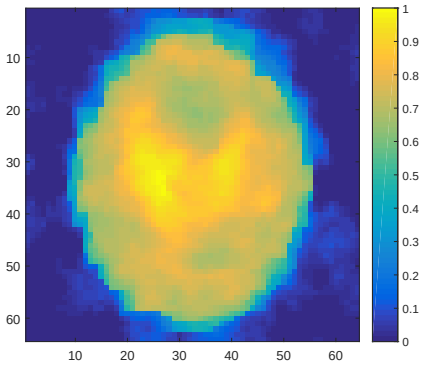


(h) SR with $L = 8$: edge-preserving

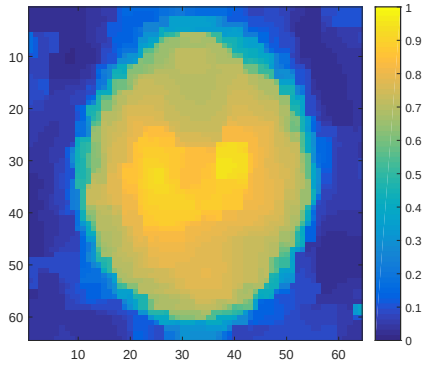
Figure 8.14: HR and SR results for different down-sampling factors (FoV = 2 cm.).



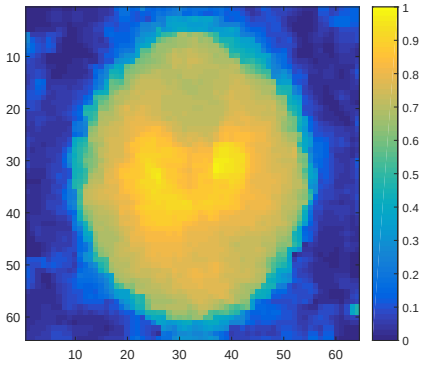
(a) HR: total variation.



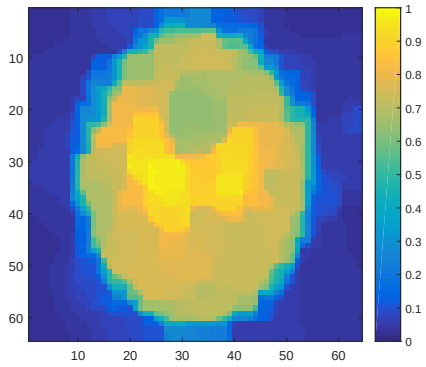
(b) HR: edge-preserving.



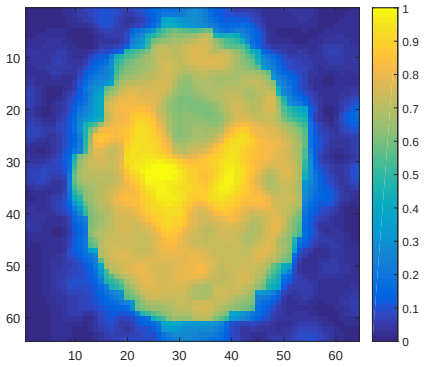
(c) SR with $L = 2$: total variation.



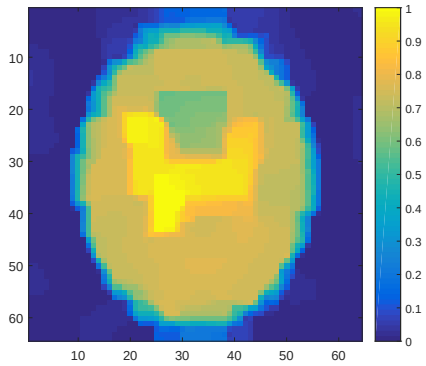
(d) SR with $L = 2$: edge-preserving



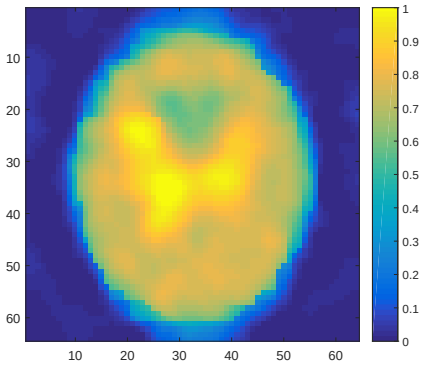
(e) SR with $L = 4$: total variation.



(f) SR with $L = 4$: edge-preserving

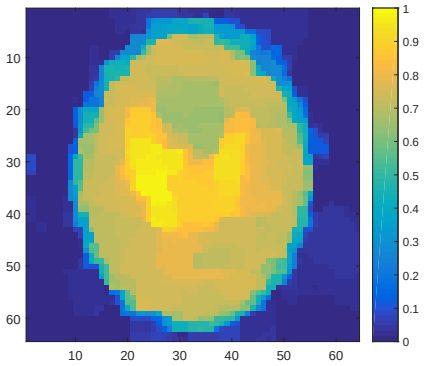


(g) SR with $L = 8$: total variation.

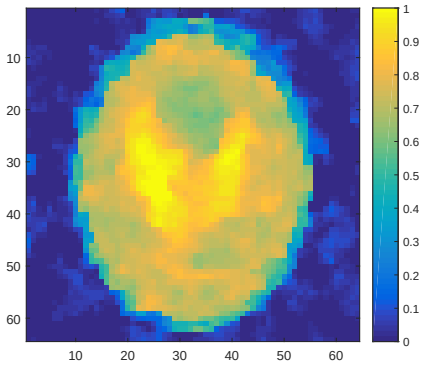


(h) SR with $L = 8$: edge-preserving

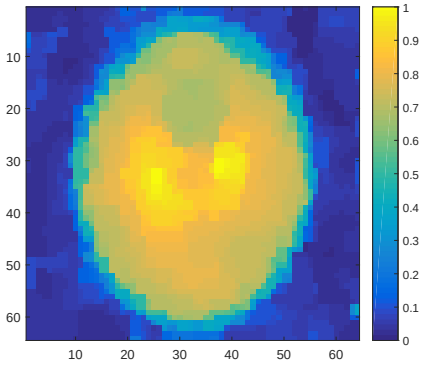
Figure 8.15: HR and SR results for different down-sampling factors (FoV = 3 cm).



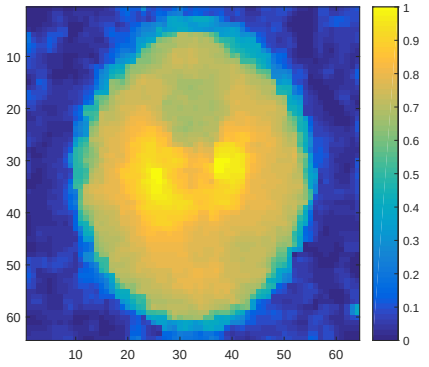
(a) HR: total variation.



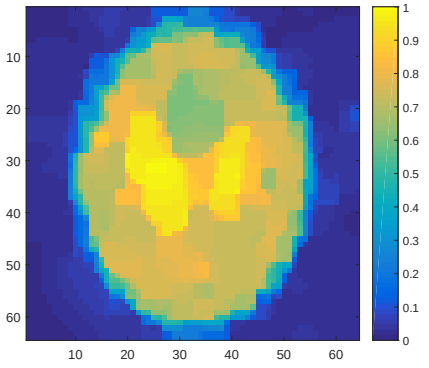
(b) HR: edge-preserving.



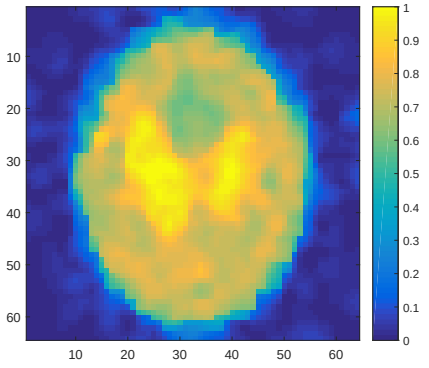
(c) SR with $L = 2$: total variation.



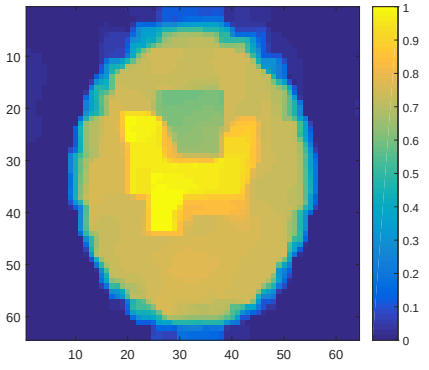
(d) SR with $L = 2$: edge-preserving



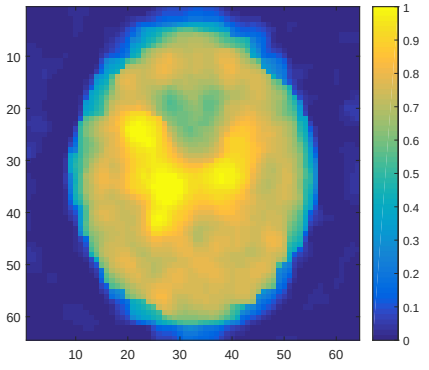
(e) SR with $L = 4$: total variation.



(f) SR with $L = 4$: edge-preserving

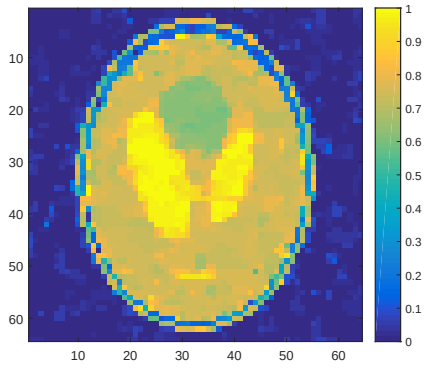


(g) SR with $L = 8$: total variation.

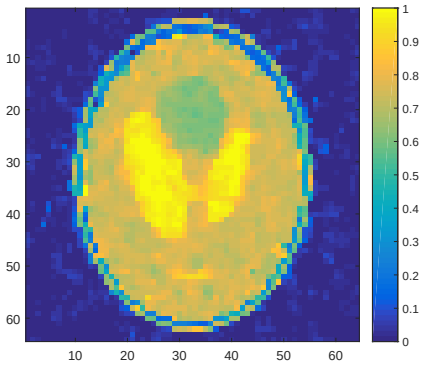


(h) SR with $L = 8$: edge-preserving

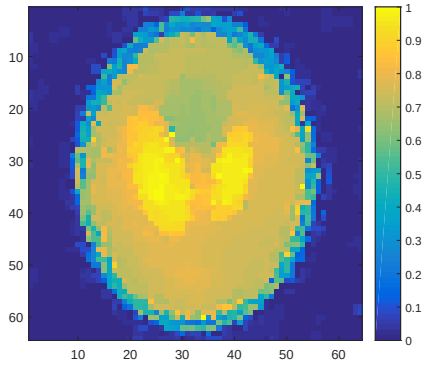
Figure 8.16: HR and SR results for different down-sampling factors (FoV = 4 cm).



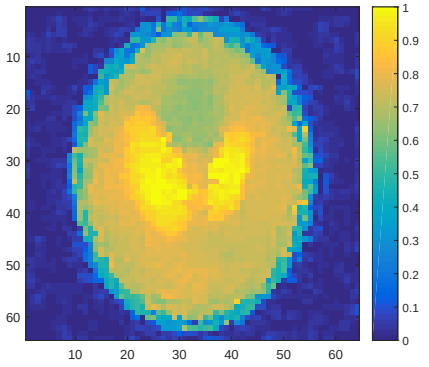
(a) HR: total variation.



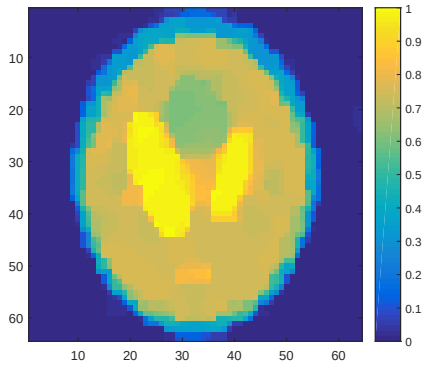
(b) HR: edge-preserving.



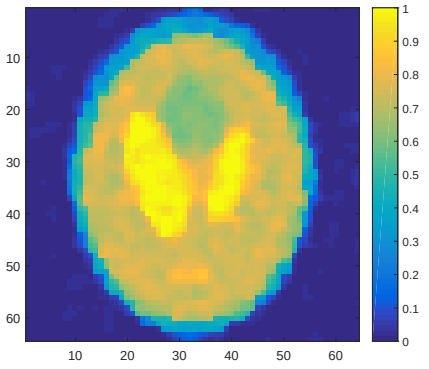
(c) SR with $L = 2$: total variation.



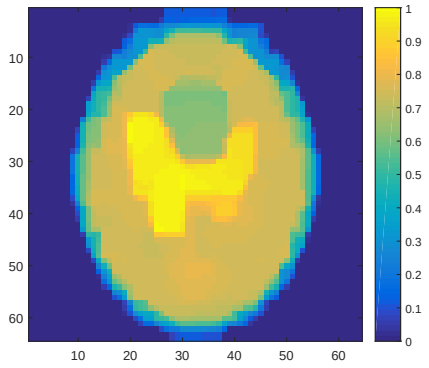
(d) SR with $L = 2$: edge-preserving



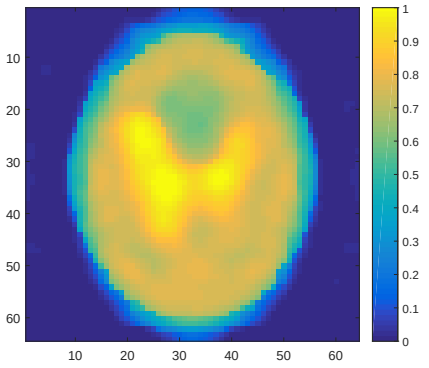
(e) SR with $L = 4$: total variation.



(f) SR with $L = 4$: edge-preserving



(g) SR with $L = 8$: total variation.



(h) SR with $L = 8$: edge-preserving

Figure 8.17: HR and SR results for different down-sampling factors (FoV = 10 cm).

Chapter 9

Application to real data

In order to test the measurement model described in section 3.3, the signal induced by a phantom object in the presence of a nonhomogeneous B_0 field was measured. This was done using a 7 T MRI scanner. However, first the phantom, which consisted of four small tubes of water, was imaged using standard techniques, in order to obtain a model result for our experiment. The resulting image is shown in Figure 9.1.

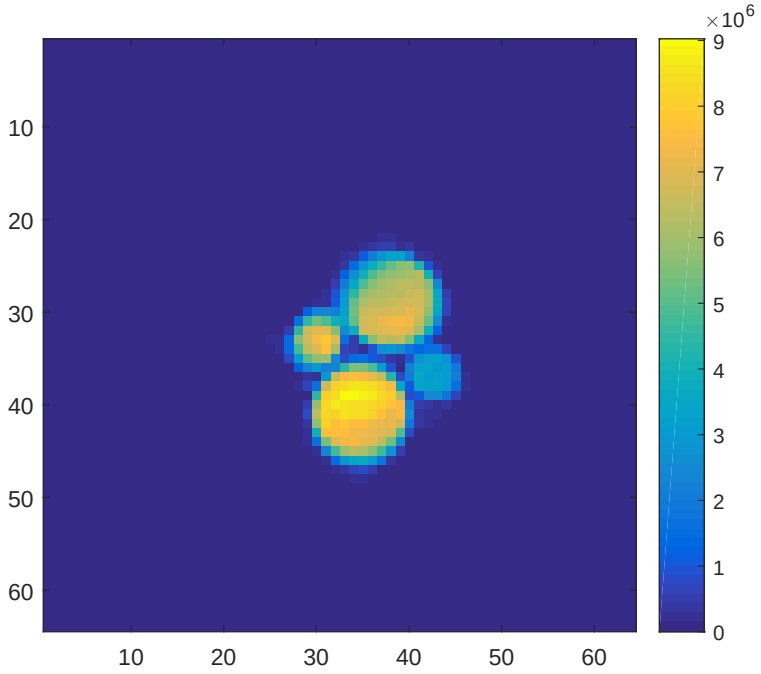
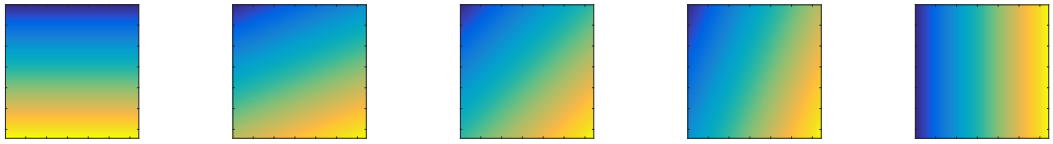


Figure 9.1: The model solution, obtained using conventional MRI techniques. The slice thickness was set to 15 cm. The four tubes of water are clearly visible.

In order to generate an inhomogeneous field, a linear gradient was applied in the xy-plane. This type of gradient was chosen because it is one of the main options that is being considered for a new Halbach array. There was no gradient in the z-plane, so no slice selection was carried out. 16 different measurements were performed, with the gradient being rotated clockwise over $\frac{2\pi}{16}$ after every measurement. The first gradient was applied in the negative y-direction. The inhomogeneity in the magnetic field for the first 5 measurements is shown in Figure 9.2, where

yellow indicates the strongest field and dark blue the weakest. However, the variations are very small, considering that the gradient is 1 mT/m and the Field of View was set to 16 cm. It is interesting to note that this experimental setup mimics the very first MRI experiment carried out by Lauterbur as described in [24].



(a) First measurement. (b) Second measurement. (c) Third measurement. (d) Fourth measurement. (e) Fifth measurement.

Figure 9.2: B_0 field used in the first five measurements.

The first measurement yielded the signal shown in Figure 9.3. There were two channels registering the voltage induced by the sample, with a phase difference of 90 degrees. The final signal was obtained by adding these two in the following way:

$$S(t) = S_1(t) + iS_2(t), \tag{9.1}$$

where $S_1(t)$ and $S_2(t)$ are the signals acquired in channel 1 and 2 respectively.

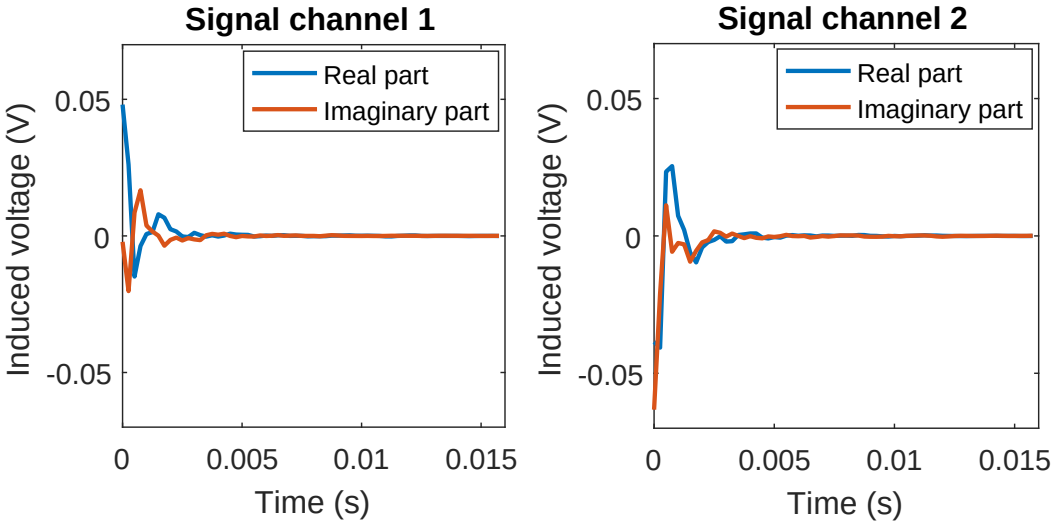


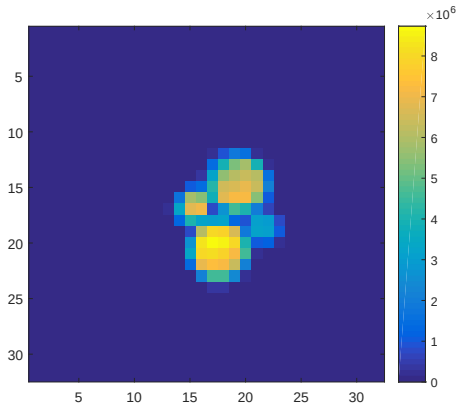
Figure 9.3: Signal acquired during the first measurement.

Two different ways of obtaining an image from these measurements will be discussed: a direct approach using Equation (3.10) and an approach based on 1D projections of the magnetization, which was inspired by the way CT images are obtained.

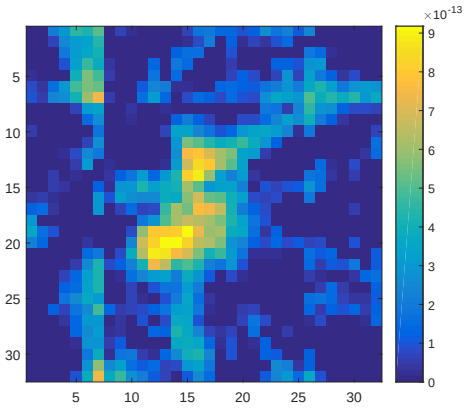
9.1 Direct reconstruction

At first, the 16 signals and corresponding matrices given by Equation (3.10) were vertically concatenated, and CGLS was used in an attempt to find a solution. However, the result was

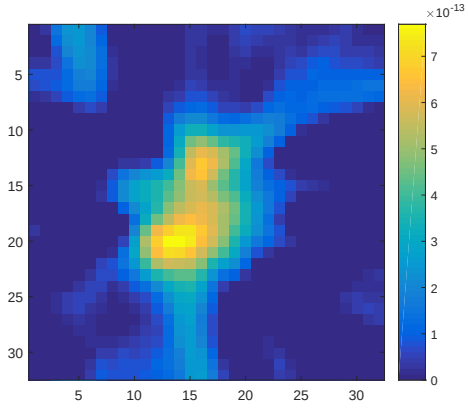
very poor. This is due to the difference in phases of the magnetization at time 0 in each of the 16 measurements. In order to resolve this issue, we set the phase of the first datapoint of the measured signal to 0 for each of the 16 angles and changed the phase of the other datapoints accordingly. This allowed us to assume that the magnetization at time 0 is strictly real for all measurements, making it possible to use Equation (3.10). Applying the CGLS method without any type of regularization yielded the 32×32 image in Figure 9.4b.



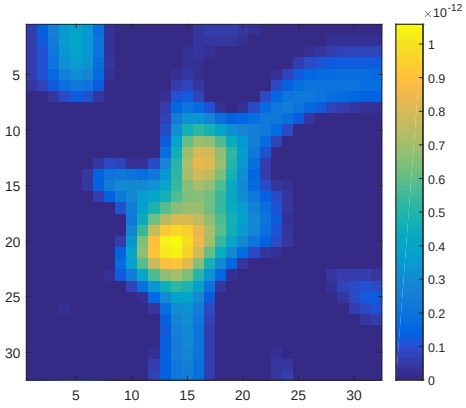
(a) Model solution.



(b) Least-squares solution to Equation (9.2).



(c) Total variation solution to Equation (3.10), with $\lambda = 2 \cdot 10^{20}$.



(d) Deblurred version of Figure 9.4c. Total variation regularization was used, $\lambda = 5 \cdot 10^{19}$.

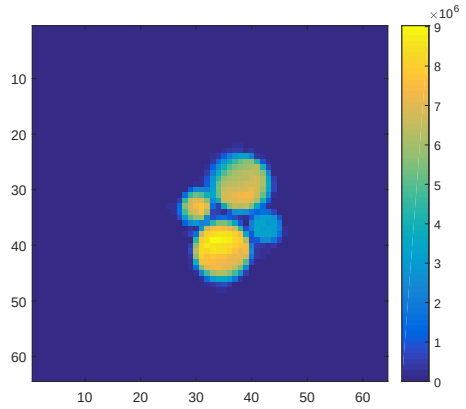
Figure 9.4: A comparison of the model image and three images using the direct reconstruction model for the 32×32 case.

Figure 9.4c was obtained using total variation regularization with $\lambda = 2 \cdot 10^{20}$. In order to make Figure 9.7d less blurry, we assumed a spatially invariant Gaussian point spread function. Setting the parameters $\rho = 0$ and $s_1 = s_2 = 2$ yielded good results. We incorporated the deblurring into the reconstruction process as follows: assuming that the solution to the minimization problem is not the actual solution $\mathbf{x}$ but the blurred version $\mathbf{x}_b = B\mathbf{x}$, we simply have to replace the reconstruction matrix W by WB in Equation (3.10). Then, when total variation regularization ($\lambda = 5 \cdot 10^{19}$) is used, the image in Figure 9.7e is obtained. It still looks somewhat blurry but less so than before.

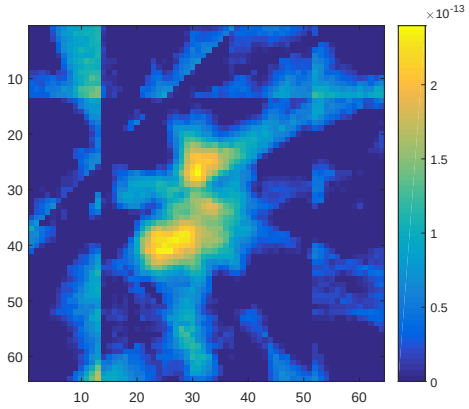
The background contains some artifacts, but we were able to reconstruct the two large water bottles. However, the lower bottle looks more stretched out than it is supposed to be. An interesting thing to note is that the water bottles are not located exactly where we expect them

to be. This can be partially explained by the fact that the gradients are not exactly linear but we do model them as such. However, in the obtained solution the water bottles are farther apart than in the model solution, which leads us to believe the gradient is actually smaller than the claimed 1 mT/m.

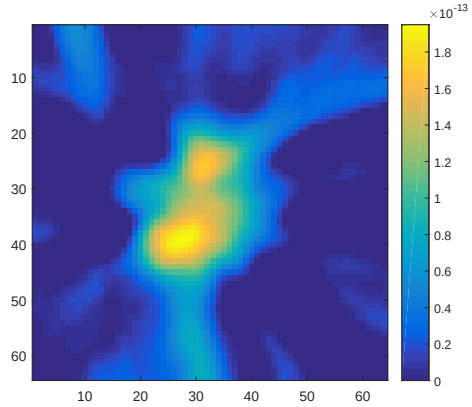
For the 64×64 case, the resulting images are shown in Figure 9.5. Here, the PSF was assumed to be Gaussian with $s_1 = s_2 = 3.5$ and $\rho = 0$. Like in the 32×32 case, only the two largest bottles show up clearly in the final deblurred image. The shapes do not look the way they are supposed to and, like in the 32×32 case, the background contains a number of artifacts.



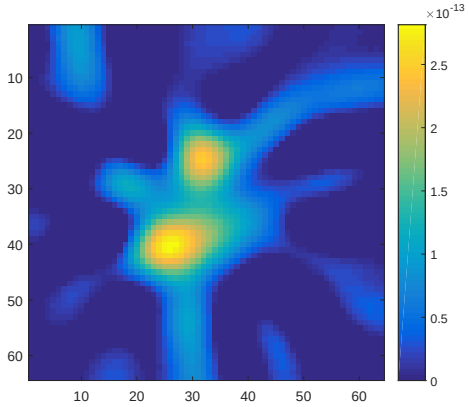
(a) Model solution.



(b) Least-squares solution to Equation (9.2).



(c) Total variation solution to Equation (9.2), with $\lambda = 3 \cdot 10^{21}$.



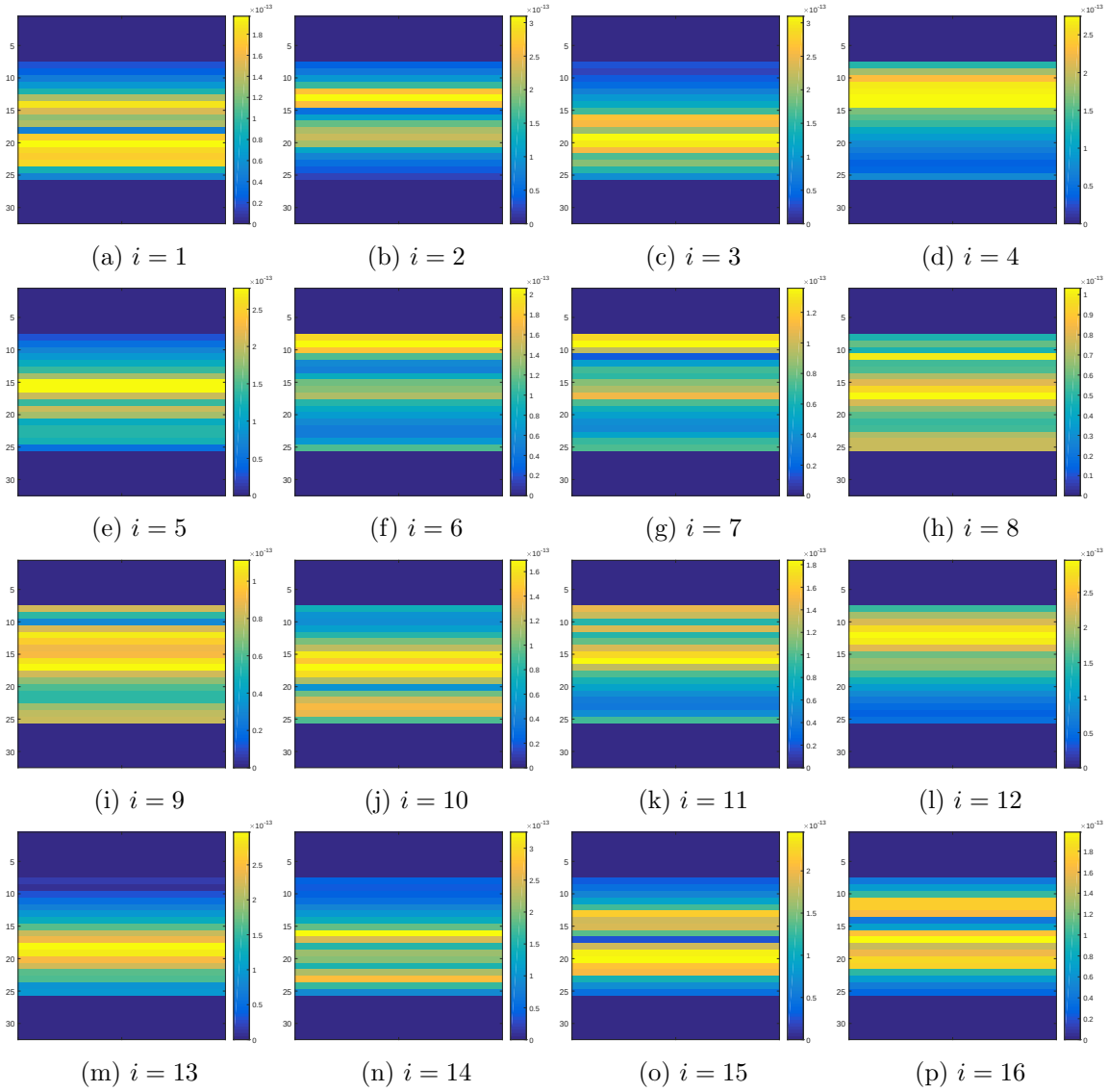
(d) Deblurred version of Figure 9.5c. Total variation regularization was used, $\lambda = 5 \cdot 10^{20}$.

Figure 9.5: A comparison of the model image and three images obtained using the direct reconstruction model for the 64×64 case.

For reasons that will become clear in section 9.2, the frequencies had to be shifted in a few of the 16 measurements. However, we had to make some educated guesses about the size of the frequency shifts. In MRI, frequency corresponds to position. Due to the clusters of high intensity pixels near the edges, the three obtained images lead us to believe that one or more of the frequency shifts that we have been using may have been incorrect. However, some different shifts were tested and the quality of the resulting images did not improve.

9.2 Using 1D projections

The approach taken in the previous section fits into the framework we have been using throughout this research. However, there is a different way of obtaining the final image, which was inspired by the way images are generated in CT scanners. In computerized tomography, the total absorption is measured along lines by transmitting X-rays and then measuring how much of the radiation was absorbed along the way. This is done for a multitude of different angles. Afterwards, the information stored in these measurements is combined to form an image where the intensity of each pixel is a measure of how much radiation it absorbs. One way of doing this is by using a filtered backprojection method [6]. Here, we will use the geometry of the magnetic field to calculate 1D projections of the total magnetization. The information in these 1D projections will be combined to generate a 2D image.



(a) $i = 1$ (b) $i = 2$ (c) $i = 3$ (d) $i = 4$

(e) $i = 5$ (f) $i = 6$ (g) $i = 7$ (h) $i = 8$

(i) $i = 9$ (j) $i = 10$ (k) $i = 11$ (l) $i = 12$

(m) $i = 13$ (n) $i = 14$ (o) $i = 15$ (p) $i = 16$

Figure 9.6: 1D projections of the magnetization for the angles $\theta_i = (i-1)\frac{2\pi}{16}$, $i = 1, ..., 16$.

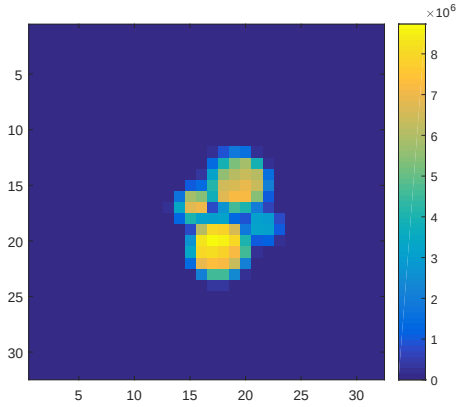
When using this kind of gradient, the problem is that the magnetic field has the same strength along lines in the FoV. So in the first measurement, all pixels on a horizontal line will correspond to the same magnetic field, making spatial encoding in the x-direction impossible. Therefore, for

each of the measurements, a 1D-projection of the total magnetization along these lines can be obtained, but not a 2D image. However, the combined information in the 16 1D projections can be used to generate the 2D image. The 1D projections were calculated using Equation (3.10). The elements of the matrix W were calculated using Equation (3.9), but the coil sensitivity and the receive field were ignored, because these can be assumed to be constant in high-field MRI. $1/T_2^*$ was calculated using Equation (7.1). (However, when the obtained 1D projections in this case were compared to the simplified case with $1/T_2^*$ set to 0, there was no significant difference in the results, due to the high sampling rate of 4000 Hz.) The model solution is a 64×64 image, but because there are only 16 1D projections, we decided to aim for a 32×32 image first. The 16 obtained 1D projections are shown in Figure 9.6. Because the excitation bandwidth is 4000 Hz, only 18 rows of pixels were excited during the measurements, leading to the zeros at the top and bottom of the 1D projections. We would expect the lowest magnetization values to occur at the boundary of the excited region, and the highest to be located somewhere in the middle, because we know that the phantom was placed in the center of the FoV and is relatively small. However, measurements 3, 4, 6, 7, 10, 11 and 12 do not reflect this behavior, making it likely that a frequency shift was carried out at some point during the imaging process. In order to correct for this, results 4, 6, 7, 11 and 12 were shifted down by 4 rows while results 3 and 10 were shifted up by 4 rows. (For the 64×64 case, frequency shifts of 8 rows were carried out.) It has to be noted that we are not completely certain about the size of the frequency shifts. In MR imaging, frequencies correspond to locations, which means that a wrong shift can seriously impact the resulting image.

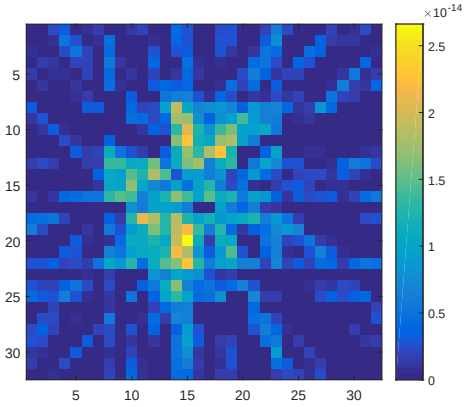
The built-in MATLAB function iradon uses a filtered backprojection algorithm to construct a 2D image based on 1D projections along different angles. When this function is applied to the results in Figure 9.6, the final image as shown in Figure 9.9b is obtained. Clearly, this is a very poor solution. The scale of our solution is of the order 10^{-14}, while the scale of the model image is of the order 10^6. This difference can be explained by the fact that we are ignoring constants in our model.

Another way of reconstructing the solution is by simply defining matrices V_i whose elements specify how much each pixel contributes to the values in the i^{th} 1D projection. This is straightforward when the gradient is pointing in the x or y direction, so for $i = 1, 5, 9, 13$. In these cases, the matrix consists solely of 1s and 0s. For the other measurements, deciding how much each pixel contributes is not as simple, because the projections occur along lines that traverse the 2D space in a way that is neither horizontal nor vertical, making it difficult to determine how much each pixel contributes exactly. However, instead of assuming that the magnetic field rotates, we can assume that the gradient points in the x or y direction and the phantom object is rotated. Then, the matrix V_i can be obtained by multiplying the matrix corresponding to that specific gradient and the geometric deformation matrix corresponding to the rotation of the object, as described in chapter 7. For instance, we can assume that the 1D projection of the second measurement depends on the model solution $\mathbf{x}$ in the following way: $\mathbf{p}_2 = V_1 G_2 \mathbf{x}$, where G_2 is the matrix that rotates the pixel grid over $\frac{2\pi}{16}$. Then all matrices V_i and all projections $\mathbf{p}_i$ can be vertically concatenated to yield a linear system of equations:

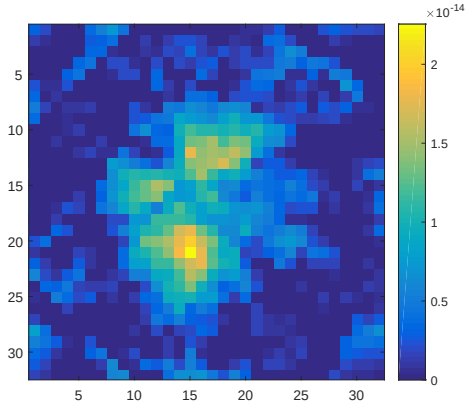
$$\mathbf{p} = V\mathbf{x}. \tag{9.2}$$



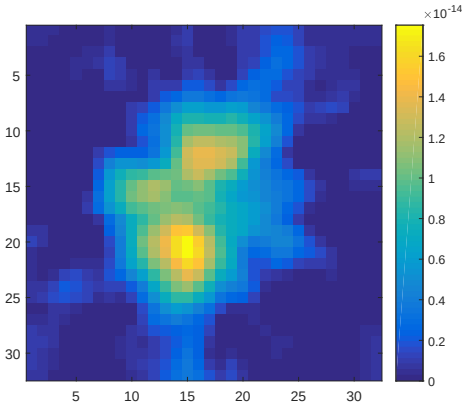
(a) Model solution.



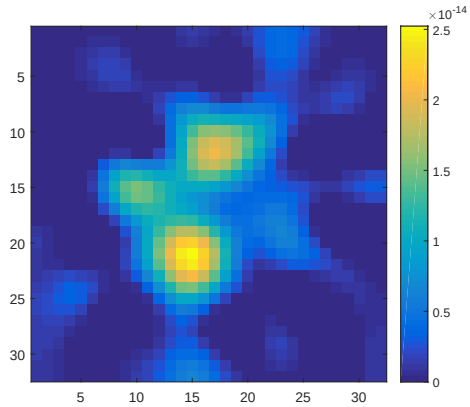
(b) Solution obtained by filtered backprojection.



(c) Least-squares solution to Equation (9.2).



(d) Total variation solution to Equation (9.2), $\lambda = 4$.

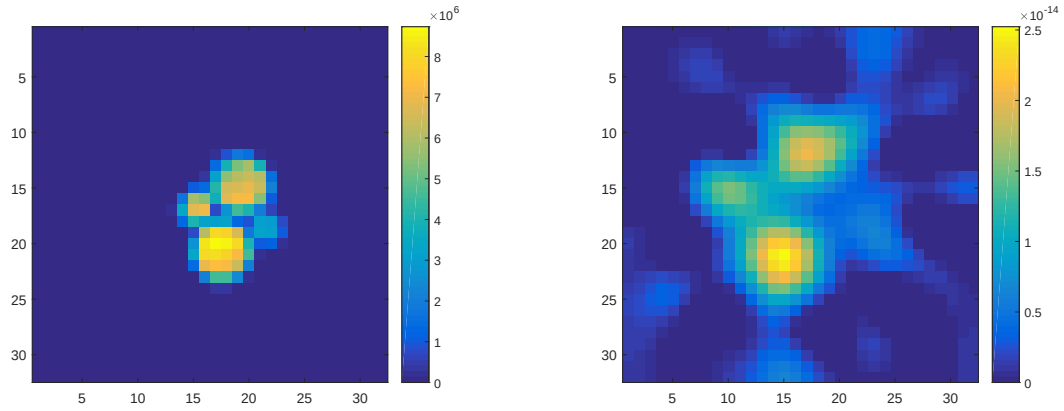


(e) Deblurred version of Figure 9.7d. Total variation regularization was used, $\lambda = 0.7$.

Figure 9.7: A comparison of the model image and four images obtained using different methods.

When the CGLS method is used to obtain the least-squares solution of this system without any kind of regularization, the image in Figure 9.7c is obtained. This image looks somewhat better than the one obtained using filtered backprojection (Figure 9.9b). The amount of back-

ground noise is reduced, and three out of the four circles of the phantom are visible to some extent. However, the image quality is still quite poor. When total variation minimization with $\lambda = 4$ is used, the image in Figure 9.7d is obtained. Now, the background noise is mostly gone and the three circles are still visible, but the image is very blurry. When the blurring matrix was included in the reconstruction problem as before, we obtained Figure 9.7e. We used the same PSF as before ($s_1 = s_2 = 2$, $\rho = 0$). Again, total variation regularization was used, with $\lambda = 0.7$. Here, three water bottles are clearly distinguishable and with a little imagination, the fourth one (on the right) is visible as well. Figure 9.8 shows the final image and the model solution side by side. Clearly, the quality of the final image leaves something to be desired, but in the obtained solution, three water bottles are clearly visible, the fourth is visible to some extent, the shape of the water bottles is approximately correct and there is not that much background noise. Compared to the images in Figure 9.7, the quality is better, because we see three bottles instead of two, the shape of the bottles is more consistent with the model solution and fewer artifacts are visible in the background.

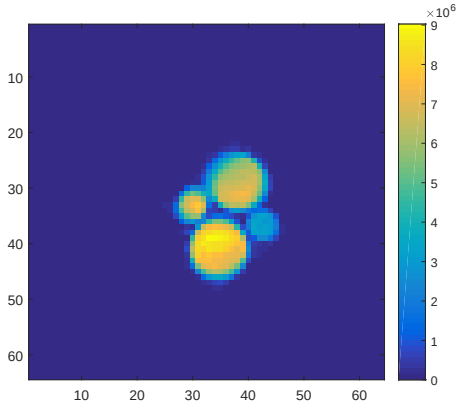


(a) Model solution.

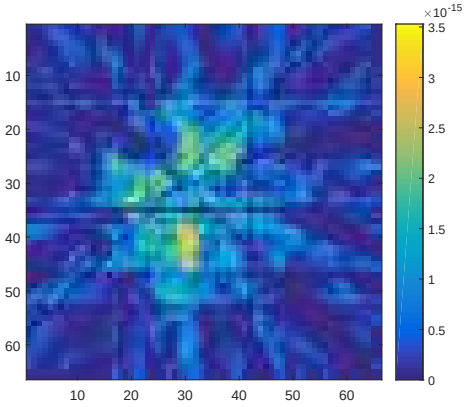
(b) Obtained solution.

Figure 9.8: Comparison of the model solution and the obtained solution.

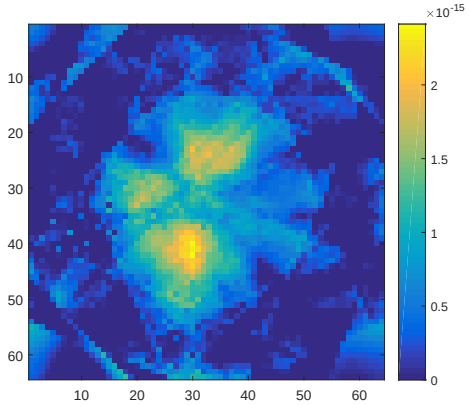
The same methods were applied to obtain a 64×64 image. The results are shown in Figure 9.9. The solutions in Figures 9.9b and 9.9c are very poor. In order to obtain a solution where most of the background artifacts are filtered out, a high regularization parameter is needed. Total variation regularization with $\lambda = 20$ was used to obtain Figure 9.9d, which is extremely blurry. An attempt at deblurring it was made using the same PSF as before ($s_1 = s_2 = 3.5, \rho = 0$). The result is shown in Figure 9.9e. These images show us that we simply do not have enough information to obtain an image of 64×64 pixels of reasonable quality with only 16 1D projections, which makes sense considering that we are dealing with a severely underdetermined system. Even after deblurring, the image looks very blurry. We can see four different shapes in the image where the bottles are supposed to be, but the shapes are not what we expect them to be. Additionally, some shapes appear in the background that are not supposed to be there.



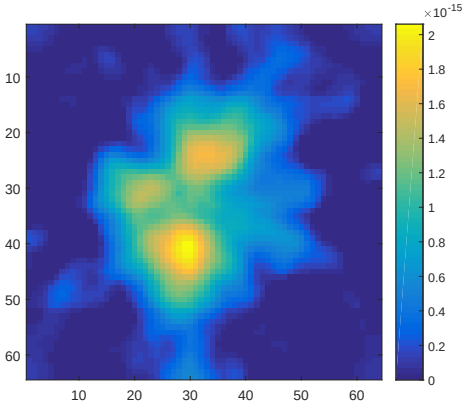
(a) Model solution.



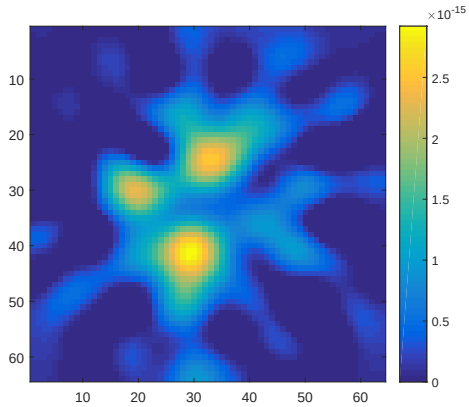
(b) Solution obtained by filtered backprojection.



(c) Least-squares solution to Equation (9.2).



(d) Total variation solution to Equation (9.2), with $\lambda = 20$.



(e) Deblurred version of Figure 9.9d. Total variation regularization was used, $\lambda = 2$.

Figure 9.9: A comparison of the model image and four images obtained using different methods for the 64×64 case.

9.3 Super-resolution

Finally, super-resolution reconstruction was applied to this dataset. Because the images based on the 1D projections proved to be better than the images that were reconstructed directly, we used the former method to obtain four LR images of 16×16 pixels. We used measurements 1, 5, 9 and 13 to reconstruct the first image, 2, 6, 10 and 14 for the second one, 3, 7, 11 and 15 for the third one and 4, 8, 12 and 16 for the last image. The resulting LR images are shown in Figure 9.10. The second, third and fourth LR images can be seen as rotated versions of the first one, the rotations being 22.5°, 45° and 67.5°. Due to the very low number of 1D projections used to produce each LR image, they are of very poor quality.

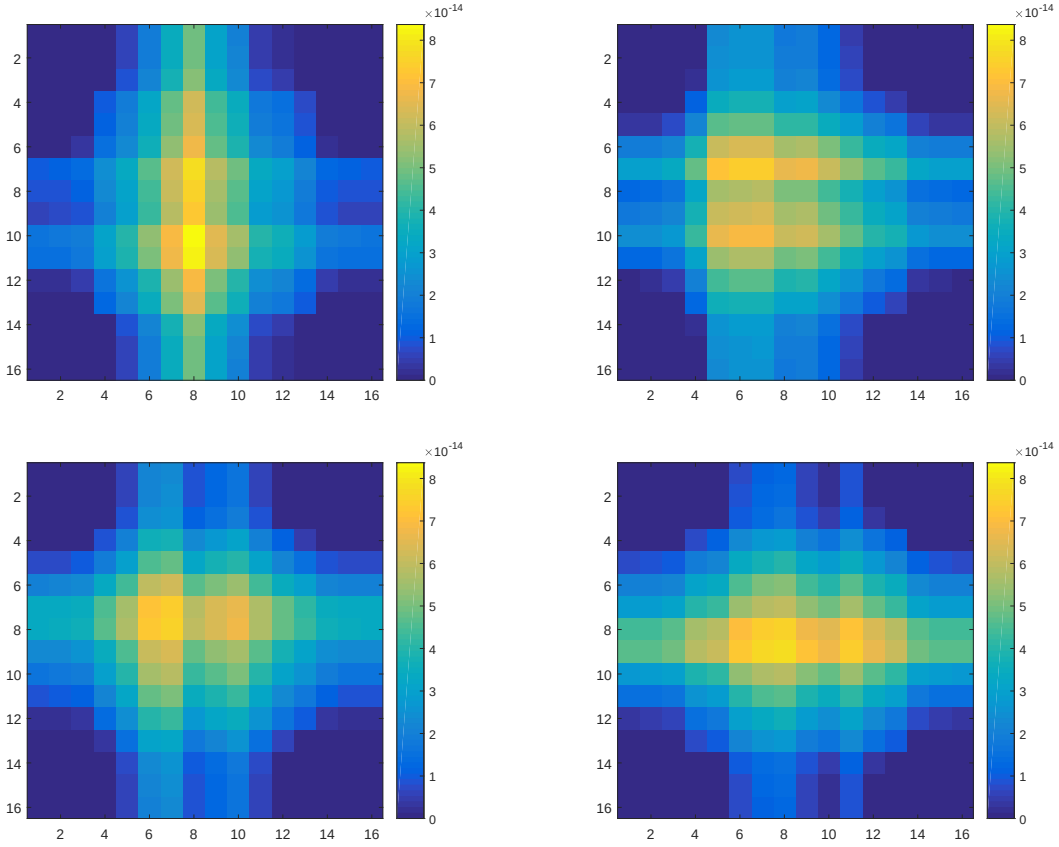
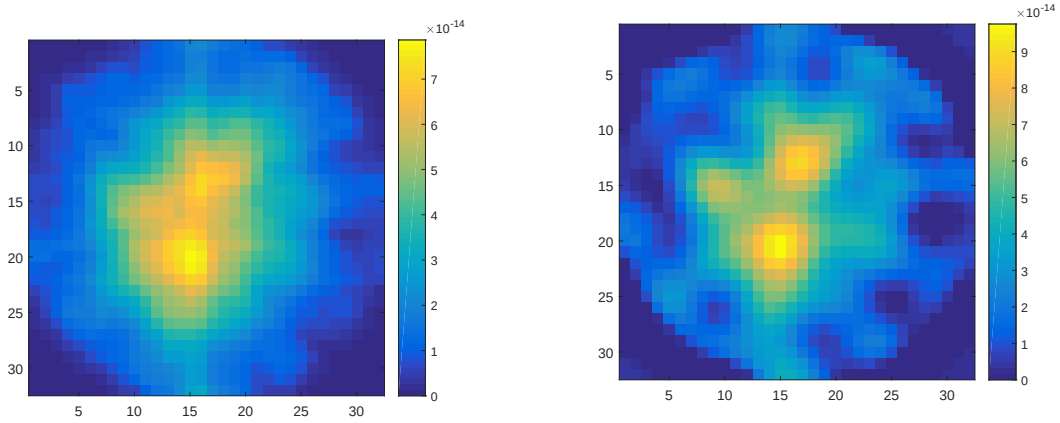


Figure 9.10: Four LR images.

The resulting HR image is shown in Figure 9.11a. It is extremely blurry. When the blurring matrix was included in the model, as in Equation (4.2), Figure 9.11b is obtained. The PSF was the same as before ($\rho = 0$, $s_1 = s_2 = 2$). The two large bottles show up clearly in the deblurred image, and the third bottle is visible as well. However, it does not look as good as the HR reconstruction in Figure 9.11b; it is blurrier and the background contains a higher number of artifacts.

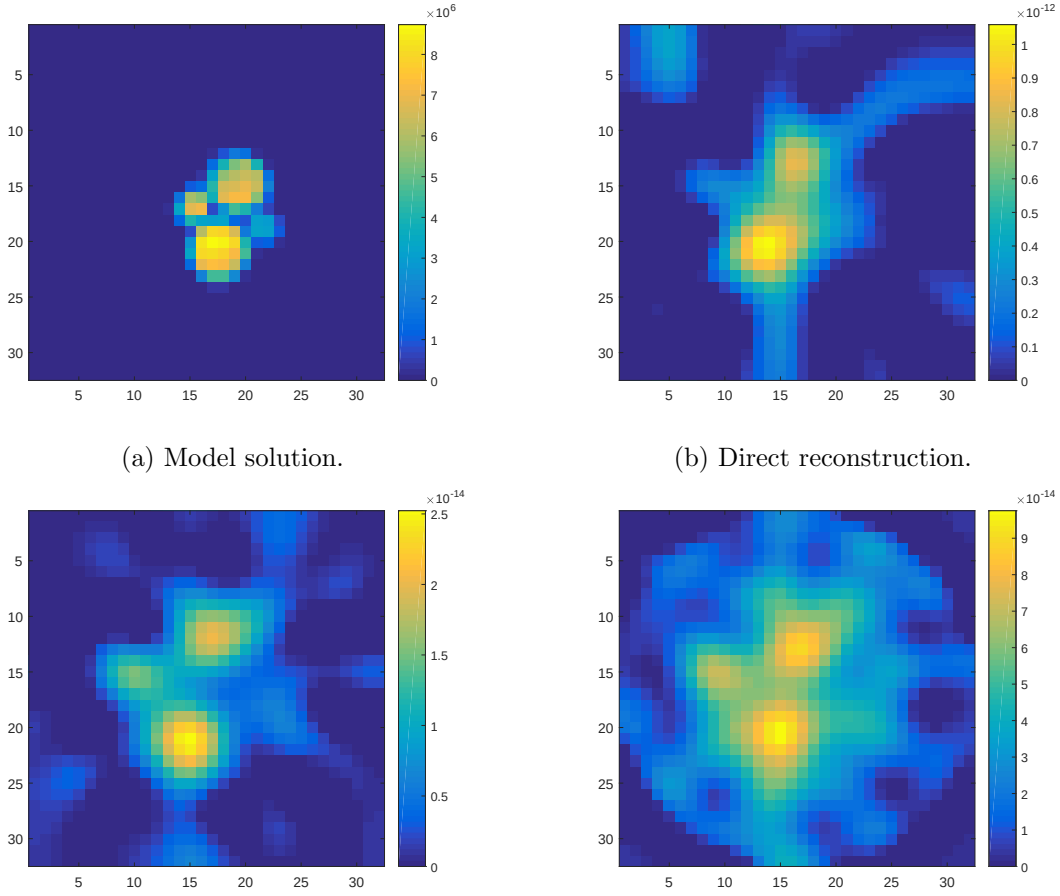


(a) Super-resolution image.

(b) Deblurred version of Figure 9.11a. Total variation regularized was used, $\lambda = 0.005$.

Figure 9.11: Super-resolution results: before and after deblurring.

9.4 Analysis of the results



(a) Model solution.

(b) Direct reconstruction.

(c) Reconstruction from 1D projections.

(d) Super-resolution reconstruction.

Figure 9.12: Final images obtained using three different methods.

Three different methods were used to obtain an image from the 16 measured signals: direct reconstruction, reconstruction based on the 16 1D projections and super-resolution reconstruction. The three different results are shown in Figure 9.12.

Reconstruction using 1D projections yields the best final image: three bottles are clearly visible and the background looks calm. Direct reconstruction yields an image where only two bottles are visible but the background is calm, while super-reconstruction yields an image with three visible bottles but a background containing a high number of artifacts. The SR image looks so blurry because only four 1D projections were used for each LR image, which caused smearing in the horizontal and vertical direction, as can be seen in Figure 9.10. This is reflected in the smearing in all directions in the HR image. It is not immediately clear why the direct reconstruction method and the method based on 1D projections yield such different results.

The main challenge in this experiment is the fact that we have very little information. This is due to the gradient varying in only one direction combined with the limited number of angles and the fact that the signal dies out very rapidly. This leads to a severely underdetermined system when we try to obtain images directly from the measured signals, making it very difficult to determine the contribution of each pixel to the induced voltage. The matrix that is used to calculate the 1D projections is not underdetermined, leading to reliable estimations among lines in the phantom image. Also, because we are basing our final image on the sum of pixel values, noise will not have as strong an influence as in the direct reconstruction case. However, because we only have 16 different 1D projections and we want to obtain images of size 32×32 or 64×64, we are dealing with another underdetermined system. Super-resolution reconstruction (based on 1D projections) does not yield a better result than a high resolution reconstruction. The LR images that were used to create the HR image were very smeared due to the fact that only four measured signals were available for each image.

In all cases, using total variation regularization leads to somewhat better, if very blurry, images. Applying a deblurring operator increases the quality of the images slightly.

Chapter 10

Conclusions and further research

In this research, the goal was to investigate whether super-resolution reconstruction can be used to improve the quality of the images produced using low-field MRI to such an extent that it can be used to detect hydrocephalus. Because we do not yet have access to a prototype that can be used in clinical trials for hydrocephalus patients, it is impossible to answer this question. Instead, phantom images were used, in simulations and measurements.

The magnetic field produced by the LUMC Halbach array has been simulated in [9]. This magnetic field was used to simulate the signals generated by the MATLAB Shepp-Logan phantom, which had been altered to accommodate for MRI physics. The noise level was based on realistic values of the system parameters. For high signal-to-noise ratios, super-resolution reconstruction is not a viable alternative for direct high resolution reconstruction. However, our simulations show that in a very noisy setting (signal-to-noise ratios lower than 1.5-2), which is what we expect in the low-field MRI prototypes, super-resolution reconstruction can in fact yield better results than a direct high resolution reconstruction, due to the higher signal-to-noise ratio per pixel.

In order to validate the measurement model, a phantom of four bottles of water was imaged in a 7 T MRI scanner. A linear gradient was applied in one direction in the xy plane. This was done 16 times, and the direction of the gradient was rotated by $\frac{2\pi}{16}$ each time. We found out that during some measurements had been shifted, but it was not clear by how much. We tried to correct for this as well as we could, but some error might have been introduced in this process. Because the gradient only varied in one direction, only 16 angles were considered and the signal dies out rapidly for each measurement, the amount of useful information was low, leading to an underdetermined system. A direct reconstruction was possible if we changed the phase of the obtained signal to 0 at the beginning of the measurement. We were able to obtain an image in which the 2 largest water bottles were visible. Because of the type of gradient that was applied, spatial encoding was only possible in one direction per measurement, making it very easy to obtain 1D projections of the magnetization for 16 different angles. This led us to reconstruct the image based on these 16 1D projections, similar to the way images are formed in CT scans. In this scenario, the phase of the signals did not have to be altered. The resulting image was better than the one obtained using a direct reconstruction: three out of the four water bottles were visible. Super resolution reconstruction did not prove to be a good approach here. The four low resolution images were generated using only four measurements each, which introduced a lot of smearing. This was reflected in the resulting high resolution image. For all three methods, the resulting image was quite blurry. The blurring operator was modeled using a spatially invariant Gaussian PSF. Deblurring improved the visibility of the individual water bottles.

The minimization problems encountered in this research were solved using CGLS or CGNE.

When $R = I$ is chosen as the regularization matrix and the regularization parameter is sufficiently large, CGNE and CGLS behave similarly. A small regularization parameter causes the first CGNE iterates to blow up, meaning that the iterates start looking acceptable long after the CGLS iterates do. When the regularization matrix is set to $R = F^*F$, where F is a first-order difference that calculates the jumps between neighboring pixels, CGLS converges much faster than CGLS. Additionally, a smaller regularization increases the number of iterations needed for convergence in CGNE more severely than in CGLS. In CGLS, calculations are performed using matrices C^{-1} and R, while CGNE uses C and R^{-1}. In this research, C is an identity matrix and R is very sparse. This observation, combined with our other findings, make CGLS the obvious choice. However, in cases where C is (nearly) singular or where calculations with C^{-1} are computationally more expensive than with R^{-1}, CGNE is a promising method.

Three different types of regularization were used: Tikhonov, total variation and edge-preserving. In all three cases, differences between neighboring pixels were penalized. Tikhonov regularization adds a penalty term to the minimization problem that grows quadratically with the difference between pixels, while the penalty is linear for total variation regularization. Edge-preserving regularization can be seen as a combination of the two: for small differences, the penalty is quadratic, while for differences larger than a certain threshold, it becomes linear. Tikhonov regularization yields the minimization problem that is the easiest and fastest to solve. Unfortunately, the quality of the images obtained using Tikhonov regularization was poorer than that of the images resulting from total variation and edge-preserving. Our expectation was that edge-preserving would yield better results than total variation regularization, because the former combines the (relatively) large penalty for small jumps with a (relatively) small penalty for large jumps. This is desirable in MRI images, because neighboring pictures are likely to represent the same tissue and hence to have the same value, while we still want jumps between different tissues to be possible. However, total variation outperformed edge-preserving regularization. This is a very beneficial result, because it requires more work and time to solve the minimization problem resulting from edge-preserving than total variation regularization.

10.1 Recommendations for future research

The team at Pennsylvania State University will be able to generate images using their prototype soon. This will be a new interesting phase in the project, as it will be the first time we will have access to signals measured by a low-field MRI scanner. We will be able to test whether super-resolution presents a good alternative to direct high resolution reconstruction for their design. In a later stage, the same will be done for the Leiden prototype.

In Chapter 9, images were reconstructed based on measurements done in an inhomogeneous magnetic field. However, due to the type of gradient that was used, spatial encoding was possible in only one direction, leading to a very limited amount of information per measurement. Combined with the low number of measurements, this made it very difficult to obtain images of good quality. In order to move past this, there are two options: improve the number of measurements by using a higher number of angles or change the gradient design in such a way that spatial encoding is possible in two directions.

For the most part of this research, blurring was not taken into account. When we used simulated data, we did not have to estimate the properties of the blurring operator. In the case of the measured data, we assumed a spatially invariant Gaussian point spread function and estimated its parameter by simply assessing which value yielded the image of the best quality. In future research, the parameter estimation could be done in a more sophisticated manner. Additionally, moving forward in this project of developing a low-field MRI scanner, we expect to be dealing with spatially variant point spread functions. Once a functioning prototype is

available, this will present a new challenge. The first attempt at obtaining the blurring matrix would be to generate an image based on a phantom consisting of a number of point sources, estimate the point spread function for each of the sources and use interpolation to obtain a point spread function for each pixel.

In [25], super-resolution reconstruction is carried out using a dictionary of high resolution images. The solution is assumed to be a sparse combination of vectors based on this dictionary. It would be very interesting to implement this in our setting. The dictionary would consist of a high number of images of healthy and hydrocephalic brains. Another option would be to use a wavelet basis for the image.

Appendix A

CGLS algorithm for ADMM

Algorithm 6 CGLS for ADMM

Require: $A \in \mathbb{C}^{M\times N}, C \in \mathbb{C}^{M\times M}, R \in \mathbb{C}^{N\times N}, F \in \mathbb{C}^{P\times N}, \mathbf{x}_0 \in \mathbb{C}^N, \mathbf{y} \in \mathbb{C}^M, \mathbf{u} \in \mathbb{C}^P, \mathbf{v} \in \mathbb{C}^P, a \in \mathbb{R} \geq 0$;

Ensure: Approximate solution $\mathbf{x}_k$ such that $\|A^*\mathbf{r}_k - aR\mathbf{x}_k + aF^*(\mathbf{v}-\mathbf{u})\| \leqslant TOL$.

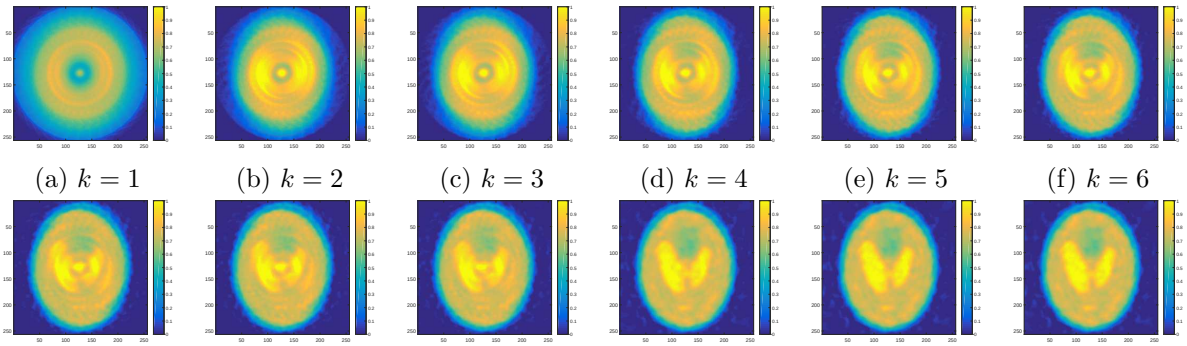
$\mathbf{r}_0 = C^{-1}(\mathbf{y} - A\mathbf{x}_0)$; $\mathbf{s}_0 = A^*\mathbf{r}_0 - aR\mathbf{x}_0 + aF^*(\mathbf{v}-\mathbf{u})$; $\mathbf{p}_0 = \mathbf{s}_0$; $\mathbf{q}_0 = A\mathbf{p}_0$; $\gamma_0 = \mathbf{s}_0^*\mathbf{s}_0$; $k = 0$;
while $\sqrt{\gamma_k} > TOL$ **and** $k < k_{max}$ **do**
  $\xi_k = \mathbf{q}_k^* C^{-1}\mathbf{q}_k + a\mathbf{p}_k^* R\mathbf{p}_k$
  $\alpha_k = \frac{\gamma_k}{\xi_k}$
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k\mathbf{p}_k$; $R\mathbf{x}_{k+1} = R\mathbf{x}_k + \alpha_k R\mathbf{p}_k$
  $\mathbf{r}_{k+1} = \mathbf{y} - A\mathbf{x}_{k+1}$
  $\mathbf{s}_{k+1} = A^*\mathbf{r}_{k+1} - \lambda R\mathbf{x}_{k+1} + aF^*(\mathbf{v}-\mathbf{u})$
  $\gamma_{k+1} = \mathbf{s}_{k+1}^*\mathbf{s}_{k+1}$
  $\beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
  $\mathbf{p}_{k+1} = \mathbf{s}_{k+1} + \beta_k\mathbf{p}_k$
  $\mathbf{q}_{k+1} = A\mathbf{p}_{k+1}$
  $k = k + 1$
end while

Appendix B

CGLS vs CGNE: additional results

B.1 Underdetermined system

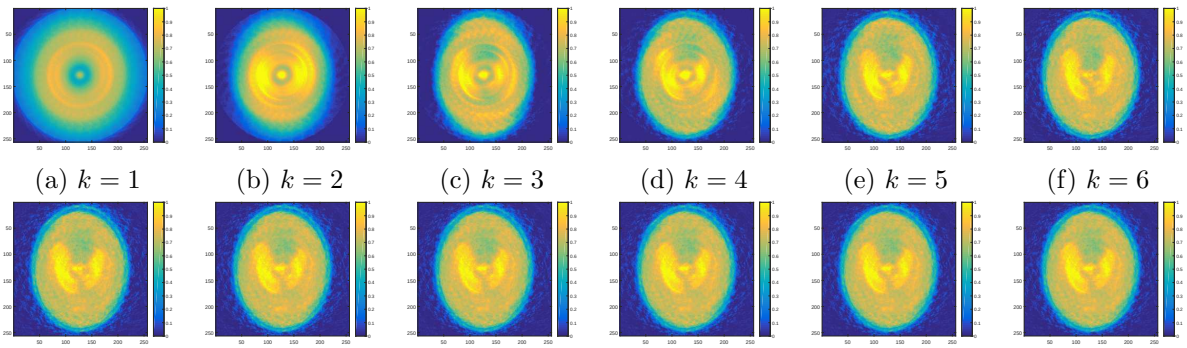
B.1.1 $R = F^*F$



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

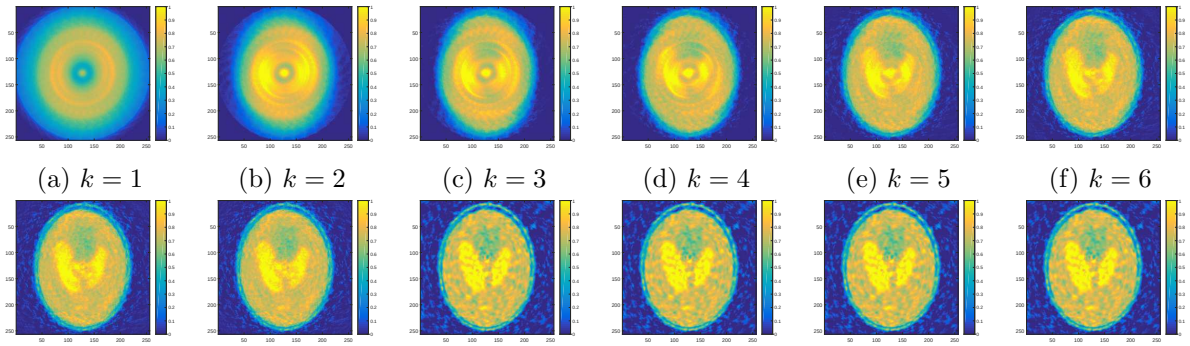
Figure B.1: CGLS iterates for $\lambda = 5 \cdot 10^{-14}$ in the case of an underdetermined system with $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

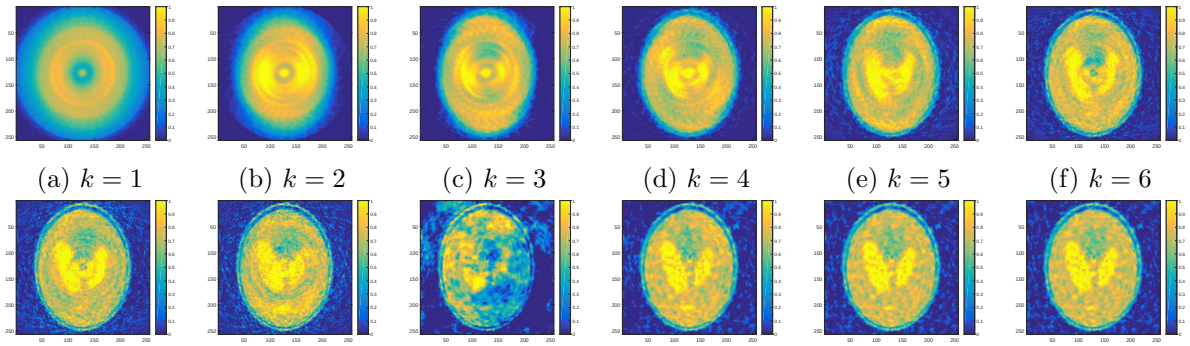
Figure B.2: CGNE iterates for $\lambda = 5 \cdot 10^{-14}$ in the case of an underdetermined system with $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 100$ (j) $k = 200$ (k) $k = 300$ (l) $k = 400$

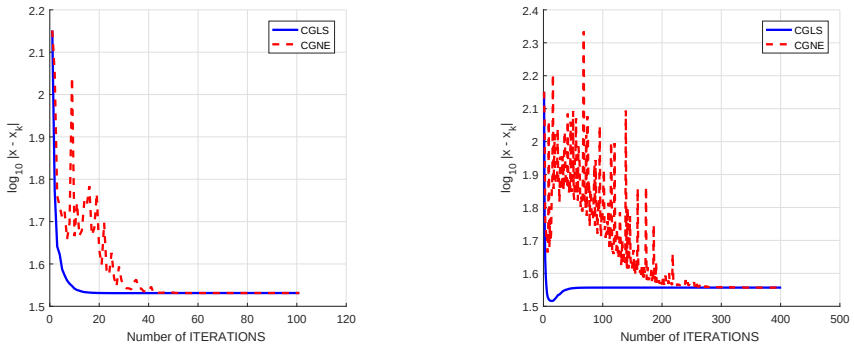
Figure B.3: CGLS iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an underdetermined system with $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 100$ (j) $k = 200$ (k) $k = 300$ (l) $k = 400$

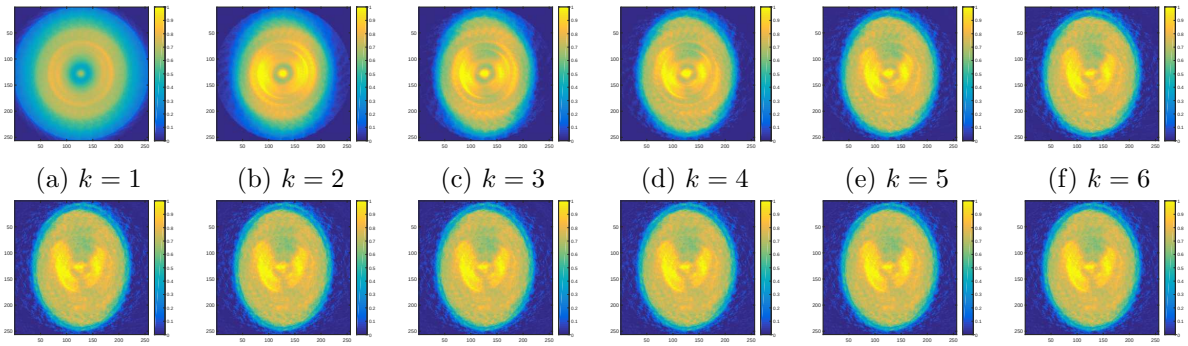
Figure B.4: CGNE iterates for $\lambda = 1 \cdot 10^{-15}$ in the case of an underdetermined system with $R = F^*F$.



(a) $\lambda = 5 \cdot 10^{-14}$. (b) $\lambda = 1 \cdot 10^{-15}$.

Figure B.5: Plots of the errors of the CGLS and CGNE iterates for two different values of the regularization parameter. Here, $R = F^*F$.

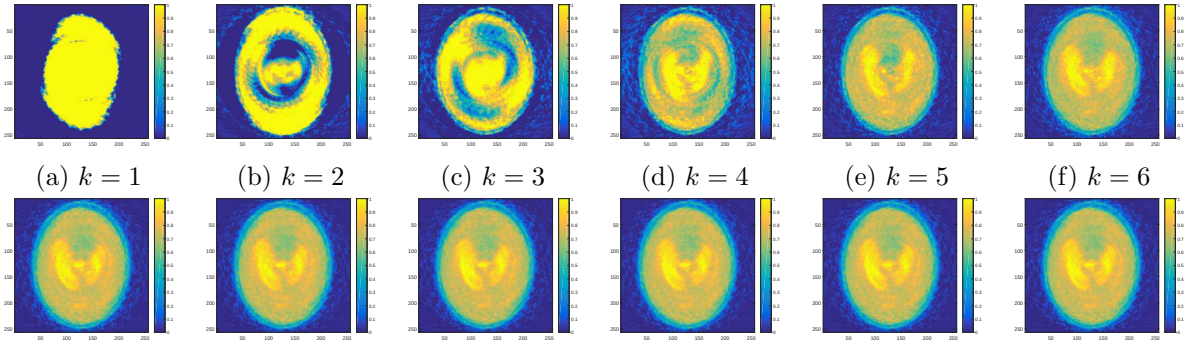
B.1.2 $R = I$



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

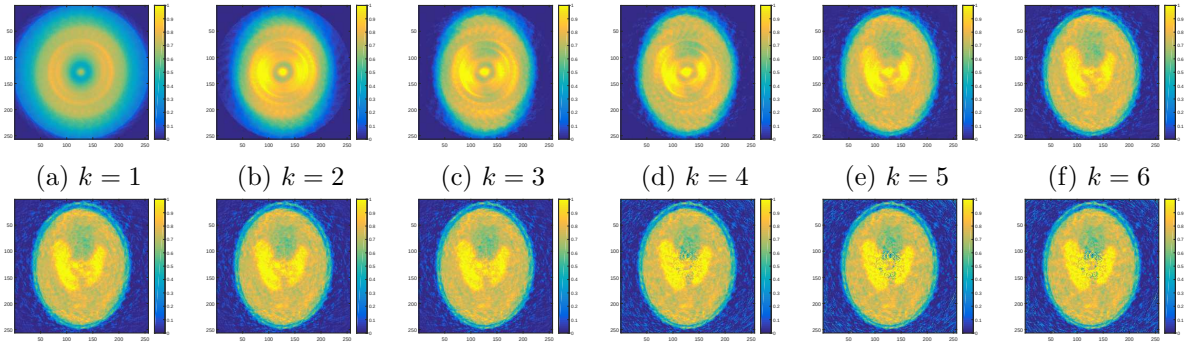
Figure B.6: CGLS iterates for $\lambda = 4 \cdot 10^{-15}$ in the case of an underdetermined system with $R = I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

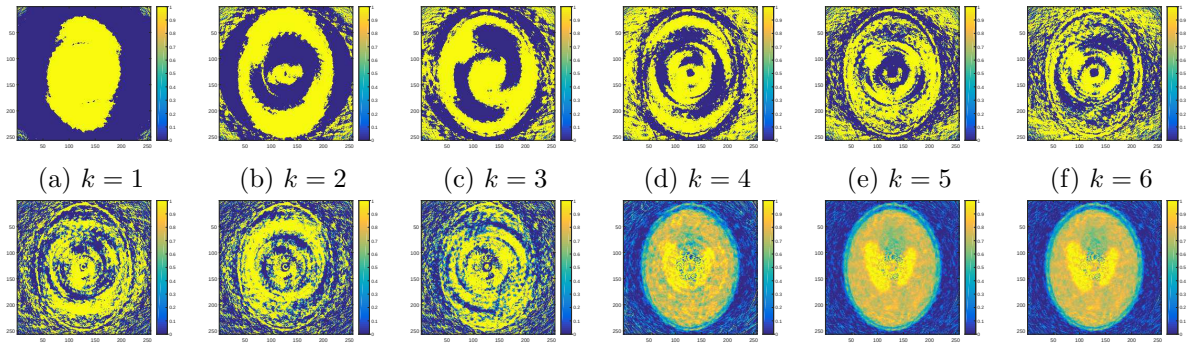
Figure B.7: CGNE iterates for $\lambda = 4 \cdot 10^{-15}$ in the case of an underdetermined system with $R = I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

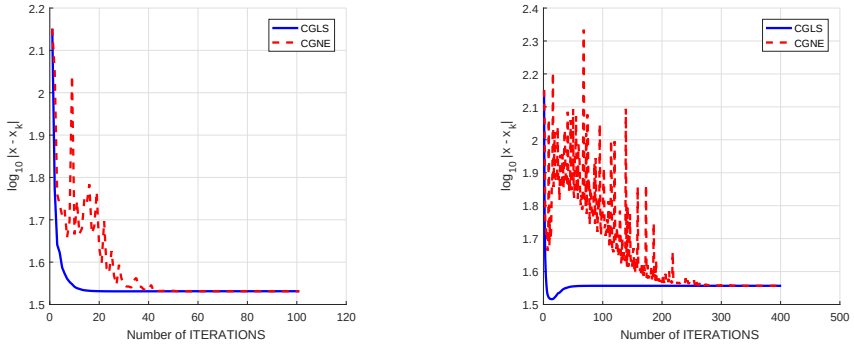
Figure B.8: CGLS iterates for $\lambda = 1 \cdot 10^{-16}$ in the case of an underdetermined system with $R = I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure B.9: CGNE iterates for $\lambda = 1 \cdot 10^{-16}$ in the case of an underdetermined system with $R = I$.



(a) $\lambda = 4 \cdot 10^{-15}$. (b) $\lambda = 1 \cdot 10^{-16}$.

Figure B.10: Plots of the errors of the CGLS and CGNE iterates for two different values of the regularization parameter. Here, $R = I$.

B.2 Correlated noise

When using C^{-1} is more cumbersome than R^{-1}, CGNE seems like the better option. In order to test this, correlated noise was generated. The noise was taken to be from a circularly symmetric complex normal distribution [26]. After every pulse, the signal consists of 20 measurements (because we are using the same overdetermined system as before). We defined the covariance for these 20 noise realizations matrix as follows: $C_{k,l} = 0.9^{|k-l|}$. Noise in signal measurements following two different pulses was considered to be independent. The results are shown in Figure B.11. λ was set to $3 \cdot 10^{-13}$.

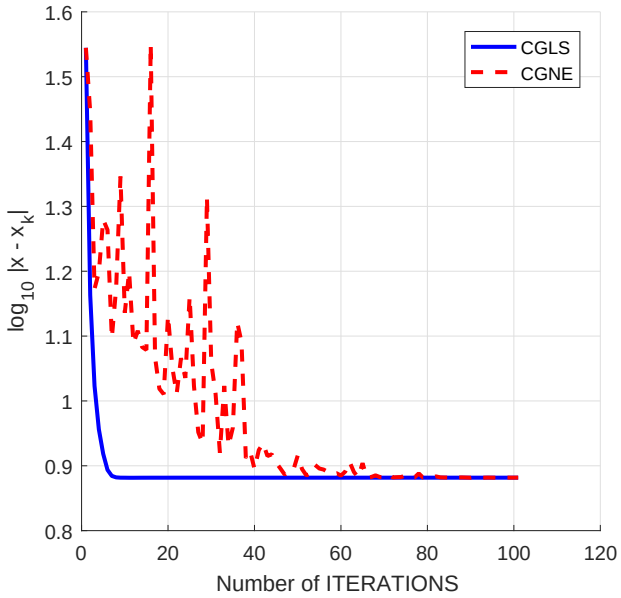
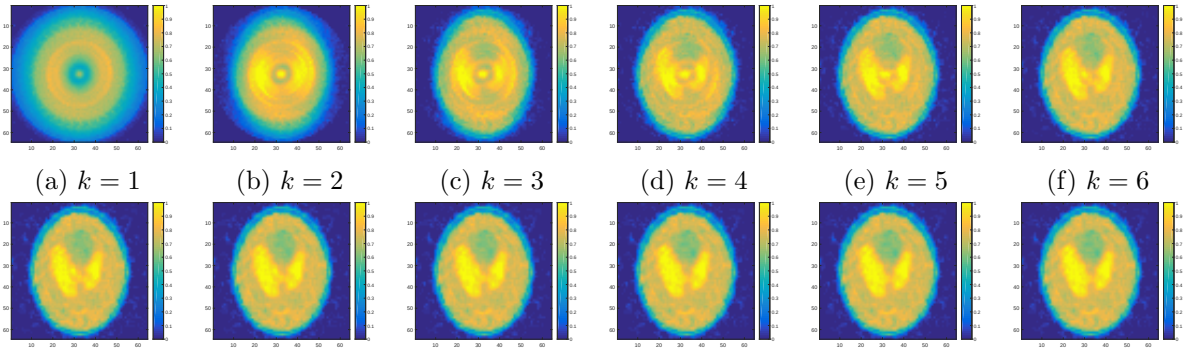


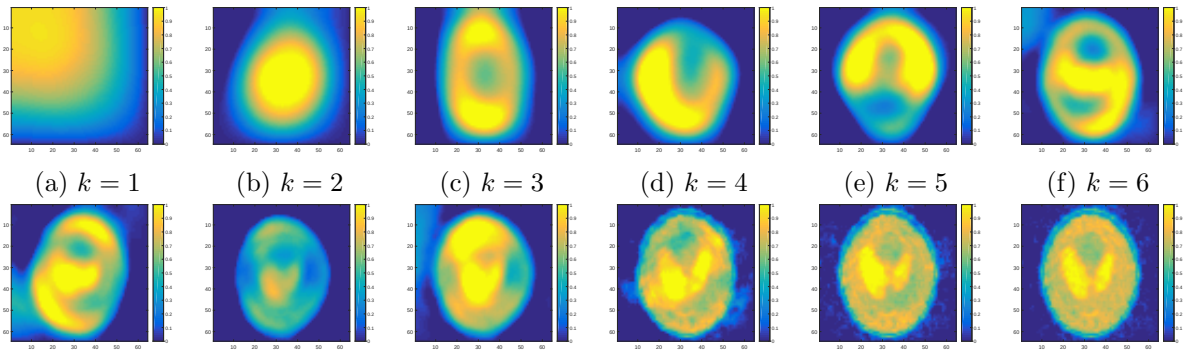
Figure B.11: Plot of the error versus the iteration number for $R = F^*F$ and $C \neq I$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure B.12: CGLS iterates for $C \neq I$ and $R = F^*F$.



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$ (d) $k = 4$ (e) $k = 5$ (f) $k = 6$

(g) $k = 7$ (h) $k = 8$ (i) $k = 9$ (j) $k = 20$ (k) $k = 40$ (l) $k = 60$

Figure B.13: CGNE iterates for $C \neq I$ and $R = F^*F$.

We see that the results look very much like what we have seen before. The CGLS and

CGNE iterates are shown in Figure B.12. Again, CGLS needs only a few iterations to converge, while CGNE needs more than 40. However, one CGNE iteration takes less time than one CGLS iteration, due to C being less sparse than R. A Cholesky decomposition was used to perform calculations with C^{-1}, speeding up the calculations by a factor of ~ 7. The average time per iteration for CGNE was shorter than for CGLS: 0.021 versus 0.025.

Bibliography

[1] M. Sarracanie, C.D. LaPierre, N. Salameh, D.E.J. Waddington, T. Witzel, and M.S. Rosen. Low-cost high-performance MRI. *Nature scientific reports*, 5, 2015.

[2] R.W. Gerchberg. Super-resolution through error energy reduction. *Journal of Modern Optics*, 21(9):709–720, 1974.

[3] P. De Santis and F. Gori. On an iterative method for super-resolution. *Journal of Modern Optics*, 22(8):691–695, 1975.

[4] D. Fiat. Method of enhancing an MRI signal, September 25 1997. US Patent 6,294,914.

[5] E. Van Reeth, I.W.K. Tham, C.H. Tan, and C.L. Poh. Super-resolution in magnetic resonance imaging: A review. *Concepts in Magnetic Resonance Part A*, 40A(6):306–325, 2012.

[6] Z.-P. Liang Liang and P.C. Lauterbur. *Principles of magnetic resonance imaging: a signal processing perspective*. The Institute of Electrical and Electronics Engineers Press, 2000.

[7] G.A. Wright. Magnetic resonance imaging. *IEEE Signal Processing Magazine*, 97:56–66, 1997.

[8] C. Kegler, H.C. Seton, and J.M.S. Hutchison. Prepolarized fast spin-echo pulse sequence for low-field MRI. *Magnetic resonance in medicine*, 57(6):1180–1184, 2007.

[9] M.S. Wijchers. Image reconstruction in MRI: The possibilities of portable, low-cost MRI scanners. Master's thesis, Delft University of Technology, 2016.

[10] D.I. Hoult and R.E. Richards. The signal-to-noise ratio of the nuclear magnetic resonance experiment. *Journal of Magnetic Resonance (1969)*, 24(1):71–85, 1976.

[11] C. Chen and D.I. Hoult. *Biomedical Magnetic Resonance Technology*. Taylor & Francis, 1989.

[12] S.C. Park, M.K. Park, and M.G. Kang. Super-resolution image reconstruction: a technical overview. *IEEE signal processing magazine*, 20(3):21–36, 2003.

[13] P.C. Hansen, J.G. Nagy, and D.P. O'Leary. *Deblurring images: matrices, spectra, and filtering*, volume 3. Siam, 2006.

[14] P.C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM review*, 34(4):561–580, 1992.

[15] R. Pan and S.J. Reeves. Efficient Huber-Markov edge-preserving image restoration. *IEEE Transactions on Image Processing*, 15(12):3728–3735, 2006.

[16] C.A. Bouman. *Model Based Image Processing*. 2013.

[17] M.B. van Gijzen. CG methods for weighted and regularised least squares problems, 2016. Personal notes.

[18] C. Vuik and D.J.P. Lahaye. Scientific computing, 2015. Lecture notes for the course Scientific Computing.

[19] M.R. Hestenes and E. Stiefel. *Methods of conjugate gradients for solving linear systems*, volume 49. NBS, 1952.

[20] J.R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain, 1994.

[21] L.A. Shepp and B.F. Logan. The Fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21(3):21–43, 1974.

[22] H.M. Gach, C. Tanase, and F. Boada. 2D & 3D Shepp-Logan phantom standards for MRI. In *Systems Engineering, 2008. ICSENG'08. 19th International Conference on*, pages 521–526. IEEE, 2008.

[23] A. van der Sluis and H. A. van der Vorst. The rate of convergence of conjugate gradients. *Numerische Mathematik*, 48(5):543–560, 1986.

[24] P.C. Lauterbur. Image formation by induced local interactions: examples employing nuclear magnetic resonance. 1973.

[25] J. Yang, J. Wright, T.S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010.

[26] R.G. Gallager. Circularly-symmetric Gaussian random vectors. *preprint*, pages 1–9, 2008.