The Application of FROID in MR Image Reconstruction

by

Linda Vu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2010

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In magnetic resonance imaging (MRI), sampling methods that lead to incomplete data coverage of k-space are used to accelerate imaging and reduce overall scan time. Non-Cartesian sampling trajectories such as radial, spiral, and random trajectories are employed to facilitate advanced imaging techniques, such as compressed sensing, or to provide more efficient coverage of k-space for a shorter scan period. When k-space is undersampled or unevenly sampled, traditional methods of transforming Fourier data to obtain the desired image, such as the FFT, may no longer be applicable. The Fourier reconstruction of optical interferometer data (FROID) algorithm is a novel reconstruction method developed by A. R. Hajian that has been successful in the field of optical interferometry in reconstructing images from sparsely and unevenly sampled data. It is applicable to cases where the collected data is a Fourier representation of the desired image or spectrum. The framework presented allows for *a priori* information, such as the positions of the sampled points, to be incorporated into the reconstruction of images. Initially, FROID assumes a guess of the real-valued spectrum or image in the form of an interpolated function and calculates the corresponding integral Fourier transform. Amplitudes are then sampled in the Fourier space at locations corresponding to the acquired measurements to form a model dataset. The guess spectrum or image is then adjusted such that the model dataset in the Fourier space is least squares fitted to measured values. In this thesis, FROID has been adapted and implemented for use in MRI where k-space is the Fourier transform of the desired image. By forming a continuous mapping of the image and modelling data in the Fourier space, a comparison and optimization with respect to data acquired in k-space that is either undersampled or irregularly sampled can be performed as long as the sampling positions are known. To apply FROID to the reconstruction of magnetic resonance images, an appropriate objective function that expresses the desired least squares fit criteria was defined and the model for interpolating Fourier data was extended to include complex values of an image. When an image with two Gaussian functions was tested, FROID was able to reconstruct images from data randomly sampled in k-space and was not restricted to data sampled evenly on a Cartesian grid. An MR image of a bone with complex values was also reconstructed using FROID and the magnitude image was compared to that reconstructed by the FFT. It was found that FROID outperformed the FFT in certain cases even when data were rectilinearly sampled.

Acknowledgements

I would like to thank my thesis supervisors, Arsen R. Hajian and Paul H. Calamai, for their support and guidance. I would like to thank members of the ALISS group, in particular, Andrew T. Cenko, whom I have consulted on numerous occasions in order to complete the work for this thesis. I would like to acknowledge and thank Christopher Tycner, Bradford B. Behr, and others for their work in developing FROID prior to this thesis.

Dedication

This is dedicated to my parents for their support throughout my academic endeavours.

Contents

List of Tables

List of Figures

Nomenclature

χ^2	Error statistic
Δk_x	Step size along k_x-axis of image
Δk_y	Step size along k_y-axis of image
Δk_{s_j}	Density compensation weight for sample j
Δx	Pixel resolution along x-axis of image
Δy	Pixel resolution along y-axis of image
ΔE	Energy difference between two spin states
η	Weights for estimating missing lines in GRAPPA
γ	Gyromagnetic ratio
$\hbar$	Plank's constant divided by 2π
$\mathbf{d}^{(n)}$	Step to be taken by variable at iteration n of optimization routine
$\mathbf{G}_{fe}$	Gradient applied during frequency encoding
$\mathbf{G}_{pe}$	Gradient applied during phase encoding
$\mathbf{G}$	Gradient of the magnetic field along the x, y and z axes
$\mathbf{p_m}$	Vector containing modelled pixel values in FROID
$\mathcal{I}(x, y)$	Image intensity at pixel position (x,y)
$\mathcal{Q}$	Spin quantum number
ω	Angular frequency
ω_0	Larmor angular frequency

ω_{rf}	Angular frequency of radio wave
Φ	Transformation for coded acquistion
ϕ	phase angle of magnetization vector induced by phase encoding
Ψ	Sparsifying transform
ρ	Spin density
τ	Duration of radio frequency pulse
θ	Flip angle of $\vec{M}_0$
α	Sparse vector for image compression
$\vec{\mu}$	Magnetic dipole moment of a nucleus
$\vec{B}_0$	External magnetic field vector
$\vec{B}_1$	Magnetic field induced by radio frequency pulse
$\vec{M}_0$	Magnetization vector cause by $\vec{B}_0$
a	Acceleration rate achieved from undersampling of rows in k-space
BW_s	Spectral bandwidth of signal
c_k	Sensitivity profile of k^{th} coil used in parallel imaging
E	Energy
f	Frequency
f_s	Sampling frequency
$g(k_x, k_y)$	Gridding kernel
G_ϕ	Magnitude of gradient used during phase encoding
H	Entropy of image
k_x	Horizontal axis in k-space
k_y	Vertical axis in k-space
M_d	Number of samples along vertical axis

m_I Magnetic quantum number

M_x x-component of $\vec{M}_0$

M_y y-component of $\vec{M}_0$

M_z z-component of $\vec{M}_0$

N_c Number of coils used during parallel imaging

N_d Number of samples along horizontal axis

p Pixel value

s Sampled value

$S(k_x, k_y)$ K-space signal

T_1 Spin-lattice relaxation time

T_2 Spin-spin relaxation time

T_E Echo time

T_R Repetition time

T_s Sampling period

T_{acq} Duration of data acquisition

T_{pe} Duration of phase encoding per acquisition

$\mathbf{A}$ Coefficient matrix for calculating real Fourier data in FROID

$\mathbf{B}$ Coefficient matrix for calculating imaginary Fourier data in FROID

M Bulk magnetization

Chapter 1

Introduction

Magnetic resonance imaging (MRI) captures energy emitted by a volume of excited nuclei with intrinsic spins that are aligned in the presence of a magnetic field. Energy is absorbed when nuclei placed in a magnetic field are pulsed with an electromagnetic wave allowing them to transition to a higher energy spin state. After excitation, this energy is released and the volume of nuclei returns to equilibrium generating signal that can be captured. The received signal consists of complex Fourier components corresponding to an image in the spatial domain. Data is acquired by sampling amplitudes in the Fourier domain known as k-space. A series of magnetic field gradients and radio frequency (RF) pulses are used to generate signals that facilitate a particular sampling trajectory. The common practice in MRI is to fill k-space with equally spaced samples corresponding to a rectilinear grid so that images may be reconstructed using a fast Fourier transform (FFT). However, there are many cases in MRI where it is desirable to fill k-space with non-rectilinear data and the use of an FFT leads to aliased images. More advanced algorithms (e.g., the gridding method) are required to reconstruct images from data sampled using spiral, radial, and random trajectories. Techniques aimed at reducing imaging time while providing sufficient data coverage are the basis of many studies. Undersampling techniques such as sensitivity encoding (SENSE), generalized autocalibrating partially parallel acquisitions (GRAPPA), and compressed sensing are used to accelerate imaging and are discussed in Chapter 3. In order to explore accelerated imaging and other novel imaging techniques the issue of reconstructing images from unevenly sampled and/or undersampled data needs to be addressed.

This thesis focuses on an image reconstruction algorithm, Fourier reconstruction of optical interferometer data (FROID), that interpolates Fourier data at arbitrary locations from a continuous model of an image. This allows the sampling locations of the observed data to be modelled accordingly and images can be reconstructed from non-rectilinear datasets. In Chapter 4, an overview of the development of FROID and its origins in optical interferom-

etry is given. The framework presented is applicable to any image reconstruction problem where the acquired data can be modelled using Fourier components. FROID shall be applied to the reconstruction of magnetic resonance (MR) images by building upon a working model that has been used to interpolate interferometer data. Necessary modifications to reflect characteristics of data acquired in k-space shall be made. The implementation of FROID for MRI and the resulting performance in image reconstruction shall be discussed in Chapters 5 and 6. The objective of this thesis is to adapt the framework of FROID to MRI and successfully reconstruct images that demonstrate the advantages of FROID.

Chapter 2

MRI Background

MRI was invented using the principles of Nuclear Magnetic Resonance (NMR), which realizes that nuclei with an odd number of protons and/or neutrons have magnetic dipole moments that can be manipulated to absorb radio frequency (RF) energy. The following sections discuss the mechanism of NMR and the extension to MRI through the use of magnetic gradients to perform two dimensional (2D) imaging.

2.1 Nuclear spins and magnetic effects

Many nuclei possess a natural spin with an associated magnetic field and magnetic dipole moment, $\vec{\mu}$. The spin angular momentum of a nucleus is parameterized using the spin quantum number, $\mathcal{Q}$, which has either a non-negative integer or half-integer value. If the atomic mass number of a nucleus is odd then $\mathcal{Q}$ equals a half-integer and if the atomic mass number is even then $\mathcal{Q}$ equals an integer for an atomic number that is odd and 0 for even.[1] Under normal conditions, the orientations of the spins are random and there is no net magnetization among all the spins combined as illustrated in Figure 2.1. However, when placed in an external magnetic field of strength B_0 (referred to as the B_0 field) each spin lines up and assumes a particular state. Depending on the spin quantum number, $\mathcal{Q}$, of the nucleus, there exists $2\mathcal{Q}+1$ spin states and orientations. Assuming that the B_0 field is applied in the z direction of a Cartesian coordinate system, μ_z is the z-component of the magnetic dipole moment used to determine the energy levels of the different spin states. The energy levels of the spin states are calculated according to [2]:

$$E = -\mu_z B_0, \tag{2.1}$$

and

$$\mu_z = \gamma m_I \hbar, \tag{2.2}$$

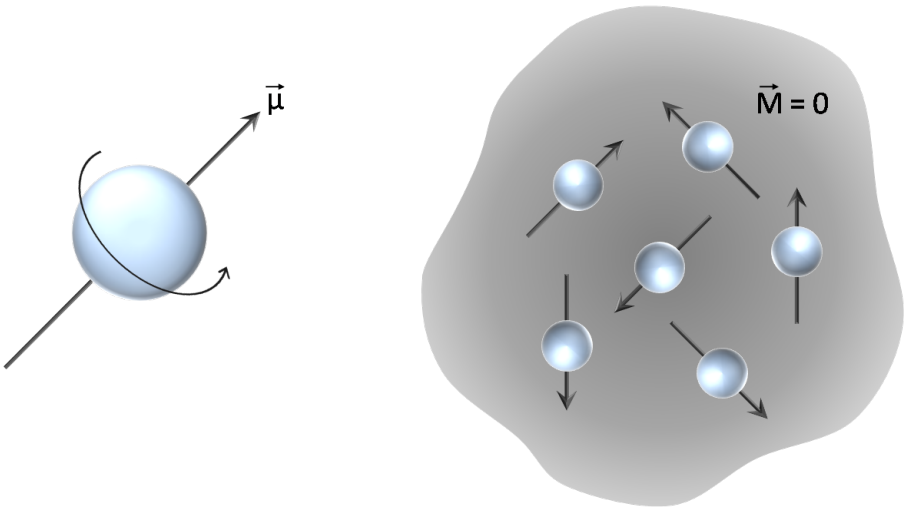


Figure 2.1: a) Magnetic dipole moment of a nucleus and b) a volume containing randomized spins resulting in zero net magnetization where $\vec{M}$ is the net magnetization vector.

where m_I is the magnetic quantum number belonging to a set of values that range from $-\mathcal{Q}$ to $+\mathcal{Q}$ separated by intervals of 1 (i.e., $m_I = -\mathcal{Q}, -\mathcal{Q}+1, \ldots, \mathcal{Q}$), γ is the gyromagnetic ratio, and $\hbar$ is Plank's constant divided by 2π. The difference in energy between two spin states is

$$\Delta E = \gamma \hbar B_0. \tag{2.3}$$

In MRI, hydrogen atoms (H^1), in particular, are of interest. Hydrogen atoms have a nuclear spin quantum number of 1/2 and therefore, two spin states. When a hydrogen atom is placed in a magnetic field, its nucleus containing only one proton assumes one of two spin states (parallel or anti-parallel with respect to $\vec{B}_0$, the direction in which the magnetic field is applied) as shown in Figure 2.2, where spins pointing upwards are in a lower energy level than spins pointing downwards. Once nuclei in a given voxel reach equilibrium in the presence of $\vec{B}_0$, more spins will be aligned in the lower energy state creating a net magnetization of $\vec{M}_0$ that points in the positive z-direction,

$$\vec{M}_0 = \sum_{n=1}^{N_s} \vec{\mu}_n, \tag{2.4}$$

where N_s is the number of spins. The interaction between the nuclear magnetic dipole moments and the external magnetic field also forces the spins to precess about the z-axis as shown in Figure 2.3 (obeying the left-hand rule).[2] The angular frequency at which the nuclei precess, ω_0, is defined using the Larmor equation,

$$\omega_0 = \gamma B_0. \tag{2.5}$$

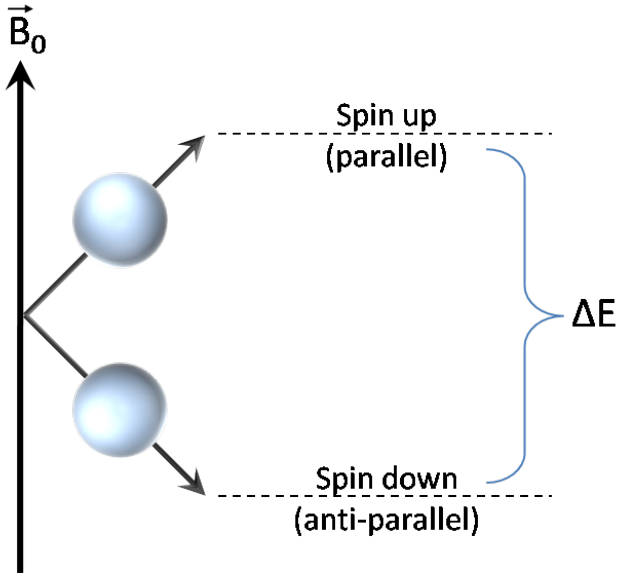


Figure 2.2: The two energy states of a hydrogen nucleus when placed in a magnetic field.

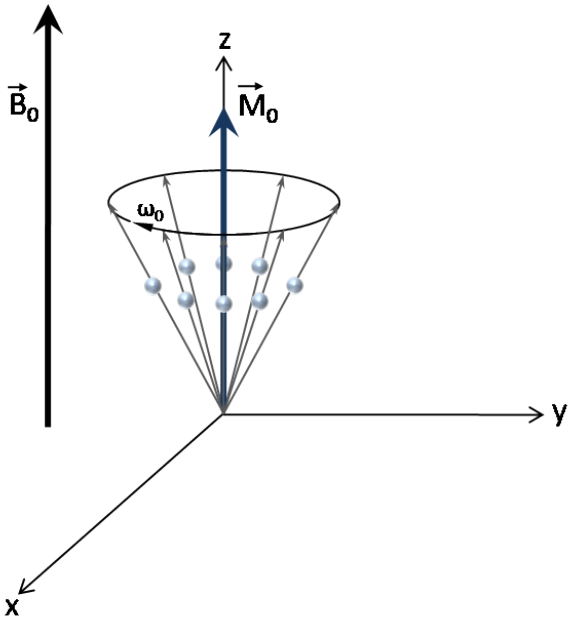


Figure 2.3: Precession of several spins (separated in phase) viewed in the x, y and z coordinates.

The precessions of the spins are at different phases resulting in the x and y components of the magnetic dipole moments of all the nuclei to cancel out and no net magnetization along the x-y plane. Therefore, $\vec{M}_0$ only has a z-component (i.e., $M_z = |\vec{M}_0|$ and $M_{xy} = 0$).[3, 4]

2.2 RF pulses and signal generation

Energy can be supplied to nuclei, aligned in a magnetic field, using electromagnetic waves that causes them to transition to a higher energy spin state. The energy of a photon in an electromagnetic wave is

$$E_p = \hbar\omega_{rf} \tag{2.6}$$

where ω_{rf} is the angular frequency at which the wave travels. Equating ΔE from equation (2.3) to E_p gives

$$\hbar\omega_{rf} = \gamma\hbar B_0,$$

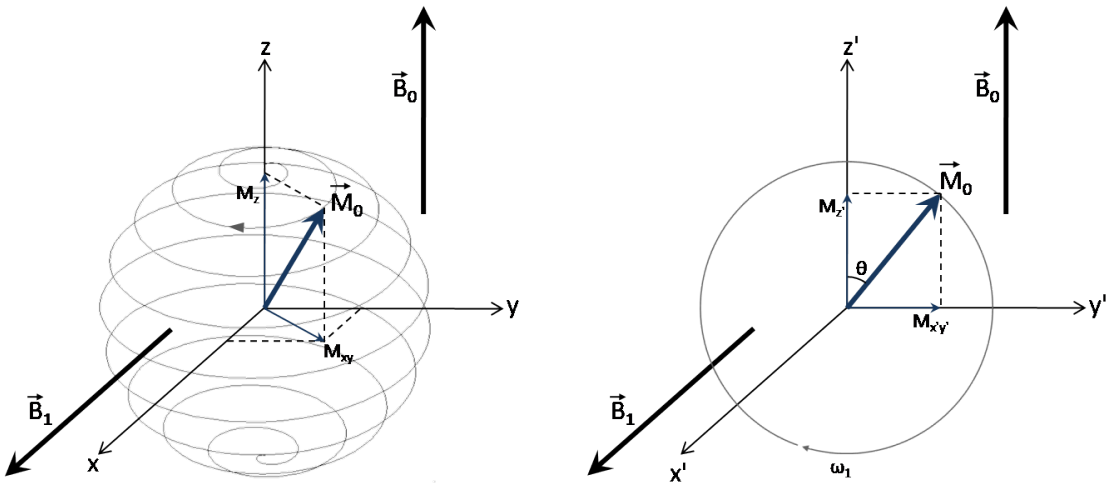
and the angular frequency required to provide packets of energy that would move nuclei to a higher energy spin state is

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

Therefore, an electromagnetic wave travelling at the same frequency as the precession (Larmor frequency) of the spins is said to be in resonance with the system and supplies energy that can be absorbed. In MRI and NMR spectroscopy, resonance is achieved in the MHz range, which corresponds to RF. Referring back to the Cartesian coordinate system, the z-component of $\vec{M}_0$ decreases as more and more nuclei receive energy from the RF pulse and move to the anti-parallel state. The RF pulse introduces another external magnetic field (B_1) in the x-direction. This causes the spins to experience an additional precession about the x-axis and as the phases of the precessions line up, an M_{xy} component (a net magnetization vector in the x-y plane) is formed. As a result, $\vec{M}_0$ travels in a spiral motion while the RF pulse is applied as shown in Figure 2.4. When the number of nuclei in the higher energy state equal the number in the lower state and the precessions of all the spins are in phase, M_z will equal zero and M_{xy} will be at its maximum value (i.e., $M_{xy} = |\vec{M}_0|$). This can be viewed as a 90° clockwise flip in $\vec{M}_0$ when observing in a rotating frame of reference (x′, y′ and z′) that follows the precession about the z-axis at frequency ω_0; the pulse is referred to as a 90° RF pulse. Accordingly, a 180° RF pulse causes $\vec{M}_0$ to flip 180° (i.e., $M_z = -|\vec{M}_0|$ and $M_{xy} = 0$). The flip angle can be determined using

$$\theta = \gamma B_1 \tau, \tag{2.7}$$

where τ is the duration of the pulse.[3]



(a) (b)

Figure 2.4: a) Motion of $\vec{M}_0$ while RF pulse is applied and b) as viewed in a rotating frame of reference (x′ and y′ form a plane rotating at ω_0).

Once the RF pulse is removed, the nuclei relax and lose energy, returning to the lower energy state (i.e., the equilibrium state). M_z recovers as more nuclei re-orient to the spin-up direction. The nuclei are also free to precess about the z-axis (i.e., there is no longer a precession about the x-axis). Inhomogeneity in the B_0 field and interactions between spins cause the nuclei to precess at slightly different frequencies. Therefore, the spins dephase and M_{xy} reduces to zero. As M_{xy} varies from the precession and the dephasing, a current can be induced in a coil placed in the x-y plane and resonating at the Larmor frequency. This generates a signal corresponding to a decaying sinusoid and is referred to as the free induction decay (FID).[2, 3]

2.3 The Bloch equation and time constants

The motion of a net magnetization vector, $\vec{M}_0$, obtained by summing over all the nuclear magnetic dipole moments in a given voxel, was empirically modelled by Felix Bloch using time dependent functions, which are referred to as the Bloch equations. Assuming a rectangular RF pulse of

$$B_1(t) = \begin{cases} B_1 & 0 \leq t \leq \tau \\ 0 & \text{else} \end{cases} \tag{2.8}$$

and a rotating frame of reference with coordinates x', y', and z', the scalar forms of the Bloch equations[2, 4] are

$$\frac{dM_{x'}}{dt} = \Delta\omega M_{y'} - \frac{M_{x'}}{T_2}, \tag{2.9}$$

$$\frac{dM_{y'}}{dt} = -\Delta\omega M_{x'} + \gamma B_1(t) M_{z'} - \frac{M_{y'}}{T_2}, \tag{2.10}$$

and

$$\frac{dM_{z'}}{dt} = -\gamma B_1(t) M_{y'} - \frac{(M_{z'} - |\vec{M}_0|)}{T_1}, \tag{2.11}$$

where $\Delta\omega$ is the difference between the Larmor frequency and the frequency of the electromagnetic wave used to excite the nuclei (i.e., the RF pulse),

$$\Delta\omega = \omega_0 - \omega_{rf}.$$

For an RF pulse that is in resonance, $\Delta\omega$ equals 0. The behaviour of $\vec{M}_0$ in the rotating frame of reference is expressed using component vectors with magnitudes $M_{x'}$, $M_{y'}$, and $M_{z'}$. If the duration of the pulse, τ, is much less than constants T_1 and T_2, then equations (2.9) to (2.11) can be further simplified and the solutions to the differential equations during RF excitation, as given in [2], are

$$M_{x'}(t) = 0, \tag{2.12}$$

$$M_{y'}(t) = M_z(0)\sin(\omega_1 t), \tag{2.13}$$

and

$$M_{z'}(t) = M_z(0)\cos(\omega_1 t) \tag{2.14}$$

for $0 \leq t \leq \tau$. $M_z(0)$ is the initial condition of M_z and is equal to $|\vec{M}_0|$ as discussed in Section 2.1. Equations (2.12) to (2.14) describe the rotation of the magnetization vector about the x-axis in the presence of the RF pulse as previously described.

After RF excitation, $B_1(t)$ equals 0 and

$$M_{x'}(t) = M_{x'}(\tau)e^{-t/T_2}, \tag{2.15}$$

$$M_{y'}(t) = M_{x'}(\tau)e^{-t/T_2}, \tag{2.16}$$

and

$$M_{z'}(t) = M_z(0)(1 - e^{-t/T_1}) + M_{z'}(\tau)e^{-t/T_1}, \tag{2.17}$$

for $t > \tau$. During spin-lattice relaxation (i.e., when excited nuclei emit energy to return to the equilibrium state) and free precession (i.e., when spins no longer precess about the x-axis and dephase from one another), $M_{z'}$ exponentially recovers to $|\vec{M}_0|$ while simultaneously $M_{x'}$ and $M_{y'}$ exponentially decay to 0. The exponential recovery and decay

are characterized by time constants T_1 (spin-lattice relaxation time) and T_2 (spin-spin relaxation time) respectively. T_1 is subject to the interactions between nuclei and their environment, which cause a loss in energy and return to equilibrium. T_2 is influenced by the interactions between pairs of nearby nuclei. When two spins are in close proximity, the magnetic moment of one spin causes the other to experience a slightly different magnetic field and lose phase coherence from the rest of the system.[3] Substituting equations (2.12) and (2.13) into (2.15) and (2.16) respectively gives

$$M_{x'y'} = M_z(0)sin(\theta)e^{-t/T_2}, \tag{2.18}$$

where $M_{x'y'}$ is the magnetization vector in the x'-y' plane of the rotating frame of reference. Since the path of the rotating frame of reference can be described using a complex exponential (moving clockwise about the z-axis), $e^{-i\omega_0 t}$, M_{xy} as observed in the laboratory frame is derived by [2] as

$$M_{xy} = M_z(0)sin(\theta)e^{-t/T_2}e^{-i\omega_0 t}. \tag{2.19}$$

The decaying envelope of the FID generated by M_{xy} can also be characterized using a modified exponential function, e^{-t/T_2^*}, where

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \gamma\Delta B_0. \tag{2.20}$$

T_2^* is the time constant adjusted to include the effect of the inhomogeneity in the B_0 field, which causes spins to experience slightly different magnetic fields and induces a phase difference.[2, 3, 4]

2.4 Imaging and magnetic field gradients

The human body consists of an abundance of hydrogen atoms, which facilitates the application of NMR to generate signal for imaging. In MRI, a particular slice of tissue is selected for imaging by applying a spatially varying magnetic field with a gradient of G_z across the body. Ideally, the magnetic field and the Larmor frequency of the spins vary linearly along the z-direction of the tissue volume. So, when an RF pulse is applied, only a section would be in resonance as depicted in Figure 2.5. The thickness of the slice is dependent on the magnitude of G_z and the bandwidth of the RF pulse. The centre frequency of the RF pulse determines the location of the slice.[3, 4]

To discern the tissue content within a slice, magnetic field gradients along the x and y axes are used. Assuming a 2D rectangular slice on an x-y plane divided into packets

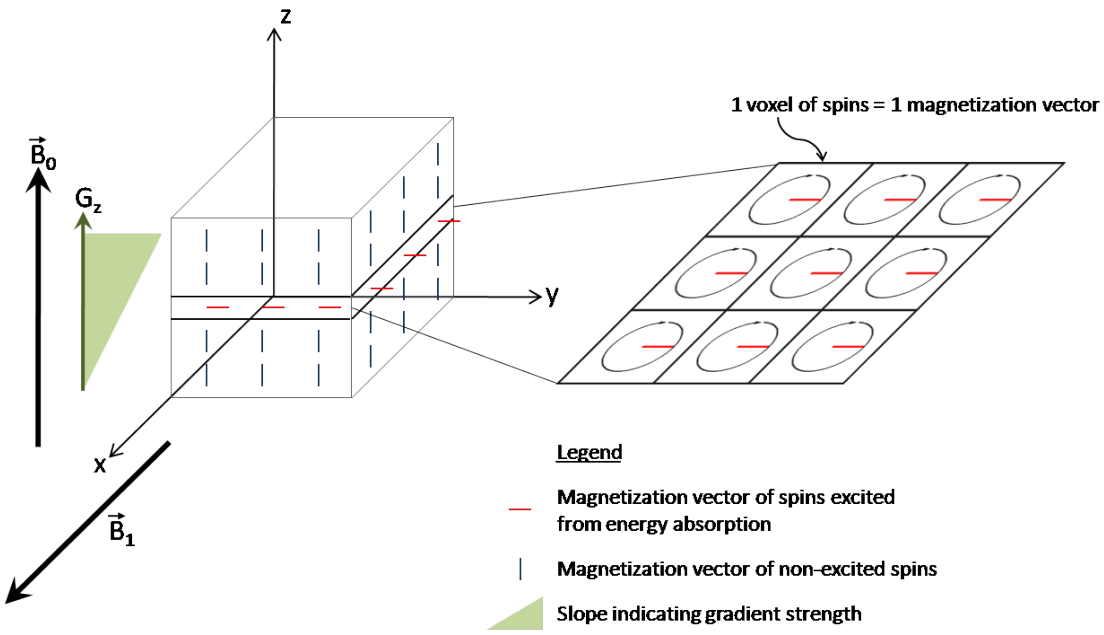


Figure 2.5: Slice selection with z-gradient and voxels of spins in selected slice precessing at the same frequency and phase after excitation with 90° pulse

of spins, the emitted FID signal, $S(t)$, generated by M_{xy} after RF excitation satisfies (according to [2])

$$\mathrm{d}S(\mathbf{r},t) \propto \rho(\mathbf{r})\mathrm{d}\mathbf{r}\; e^{-i\omega(\mathbf{r})t}, \tag{2.21}$$

where $\rho(\mathbf{r})$ is the spin density of the infinitesimal area at position $\mathbf{r}$,

$$\mathbf{r} = \begin{bmatrix} x & y \end{bmatrix}.$$

For simplicity, the amplitude of the signal is assumed to depend only on ρ (i.e., ignore T_1 and T_2^* effects) and

$$S(t) = \int\limits_{slice} \rho(\mathbf{r}) e^{-i\omega(\mathbf{r})t}\, \mathrm{d}\mathbf{r}. \tag{2.22}$$

If the magnetization vector of every voxel was precessing at the same Larmor frequency, then $\omega(\mathbf{r})$ is equal to ω_0 for all $\mathbf{r}$, and equation (2.22) becomes

$$S(t) = e^{-i\omega_0 t} \int\limits_{slice} \rho(\mathbf{r})\, \mathrm{d}\mathbf{r}, \tag{2.23}$$

where

$$M = \int\limits_{slice} \rho(\mathbf{r})\, \mathrm{d}\mathbf{r} \tag{2.24}$$

is the bulk magnetization of the slice. If a spatially varying magnetic field with a gradient of $\mathbf{G}$ (often referred to as the frequency-encoding gradient) was applied across the slice

during data collection, then each spin packet would experience a slightly different magnetic field and the corresponding magnetization vector would precess at an angular frequency dependent on its spatial position,

$$\omega(\mathbf{r}) = \omega_0 + \gamma \mathbf{G} \cdot \mathbf{r}. \tag{2.25}$$

$\mathbf{G}$ contains components along the x and y axes,

$$\mathbf{G} = \begin{bmatrix} G_x & G_y \end{bmatrix},$$

and

$$\mathbf{G} \cdot \mathbf{r} = G_x x + G_y y$$

determines the effective magnetic field at location $\mathbf{r}$. Equation (2.22) becomes

$$S(t) = \iint\limits_{slice} \rho(x, y) e^{-i[\omega_0 + \gamma(G_x x + G_y y)]t} \, \mathrm{d}x\mathrm{d}y \tag{2.26}$$

and

$$S(t) = e^{-i\omega_0 t} \int\limits_{slice} \rho(\omega) e^{i(-\omega)t} \, \mathrm{d}\omega. \tag{2.27}$$

Therefore, the signal from each spin packet is encoded with an angular frequency that can be used to locate its position within the slice. If the captured signal is considered to be in the time domain, then the spin density profile is related to its spectrum in the frequency domain.

K-space formation

The variables in equation (2.26) can be redefined according to [2] using

$$k_x = \frac{\gamma G_x t}{2\pi} \tag{2.28}$$

and

$$k_y = \frac{\gamma G_y t}{2\pi} \tag{2.29}$$

for $t \geq 0$, which gives

$$S(k_x, k_y) = e^{-i2\pi f_0 t} \iint\limits_{-\infty}^{\infty} \rho(x, y) e^{-i2\pi(k_x x + k_y y)} \, \mathrm{d}x\mathrm{d}y. \tag{2.30}$$

A data space called k-space in the spatial frequency domain is formed using coordinates k_x and k_y. The signal sampled in k-space is the Fourier transform of an image in the

spatial domain. The data coverage in k-space is determined by the values G_x and G_y as N_d samples are collected in times t_1,..,t_{N_d}.

Before frequency encoding of the tissue slice is performed, the magnetic field gradients can also be used to set the phase angles (ϕ) of the magnetization vectors of the spin packets. This is known as phase encoding (pe) and

$$\phi(\mathbf{r}) = -\gamma \mathbf{G}_{pe} \cdot \mathbf{r} T_{pe}. \tag{2.31}$$

This corresponds to positioning the start of data collection in k-space at (k_{x_0},k_{y_0}) where

$$k_{x_0} = \frac{\gamma G_{x_{pe}} T_{pe}}{2\pi}, \tag{2.32}$$

and

$$k_{y_0} = \frac{\gamma G_{y_{pe}} T_{pe}}{2\pi}, \tag{2.33}$$

and T_{pe} is the duration of the phase-encoding step. During frequency encoding (fe), data are sampled at positions

$$k_x(n) = k_{x_0} + \frac{\gamma G_{x_{fe}} t_n}{2\pi} \tag{2.34}$$

and

$$k_y(n) = k_{y_0} + \frac{\gamma G_{y_{fe}} t_n}{2\pi}, \tag{2.35}$$

for $n = 1, ..., N_d$. Therefore, through the use of magnetic field gradients the emitted RF signal can be encoded with specific phases and frequencies such that acquired samples belong to k-space.[2, 3, 4]

Rectilinear sampling

A basic MRI sequence involves manipulating the gradients so that data in k-space are acquired line by line, known as rectilinear sampling. During phase encoding, the magnetic field only has a y-gradient creating a frequency difference in the precession of the spins along the y-direction of the slice as shown in Figure 2.6a. As a result, the magnetization vectors from row to row in the slice begin to dephase. The phase difference between rows,

$$\Delta\phi(y) = -\gamma G_{y_{pe}} \Delta y T_{pe}, \tag{2.36}$$

increases over time while the y-gradient is applied. Therefore, the signal generated by each row in the slice has a different phase. Data is collected in k-space starting at (0, k_{y_0}) where k_{y_0} is given by equation (2.33). After phase encoding, frequency encoding with only an x-gradient is performed. The magnetization vectors from column to column now precesses at different frequencies as shown in Figure 2.6b. As a result, the RF emission of each spin

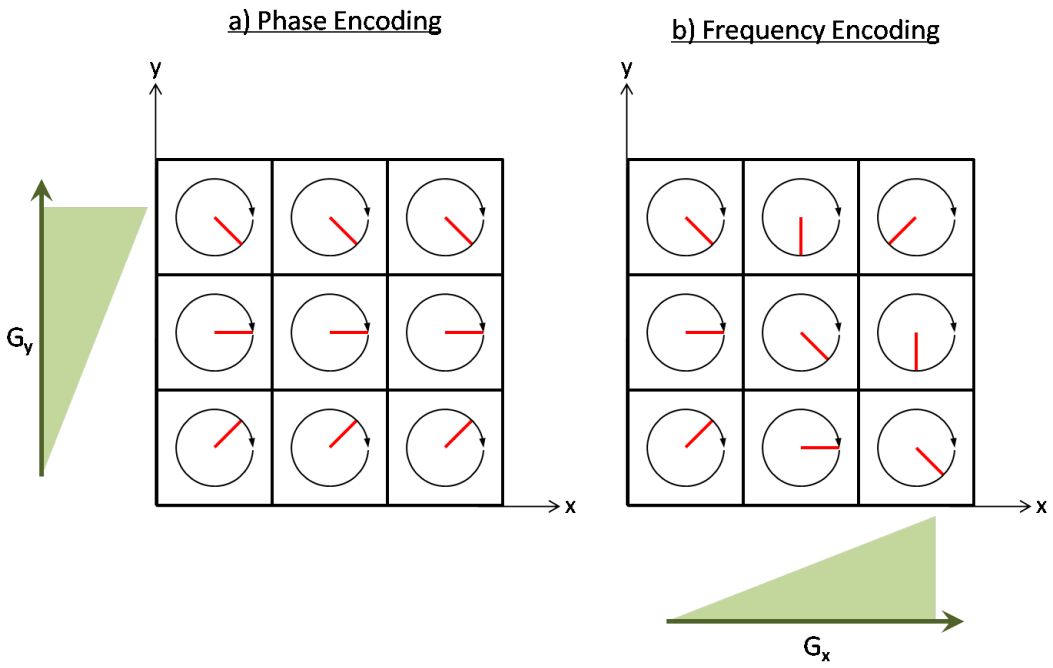


Figure 2.6: a) Phase differences in the magnetization vectors induced by phase encoding and b) frequency differences induced by frequency encoding for rectilinear sampling.

packet has a frequency and phase that is localized to its position, and equation (2.30) can be used to describe the generated signal in k-space. While the frequency encoding gradient is on, signal is recorded for a period of T_{acq}. Data is collected in the positive k_x direction for $G_{x_{fe}} > 0$, corresponding to a row in k-space. Equations (2.34) and (2.35) can be used to described the locations of the sampled points with inputs

$$k_{x_0} = 0$$

and

$$G_{y_{fe}} = 0.$$

Figure 2.7 shows a simplified timing diagram describing the sequence of gradients and RF pulse. The entire sequence is repeated, but with an increase in $G_{y_{pe}}$ so that data collection is progressed vertically along the k_y direction with each repetition. Each phase encoding step corresponds to one row of acquisition along the k_x-direction. As $G_{y_{pe}}$ increases linearly from $-G_\phi$ to $+G_\phi$ in M_d phase encoding steps, M_d rows of k-space are captured (centred at $k_y = 0$).[2]

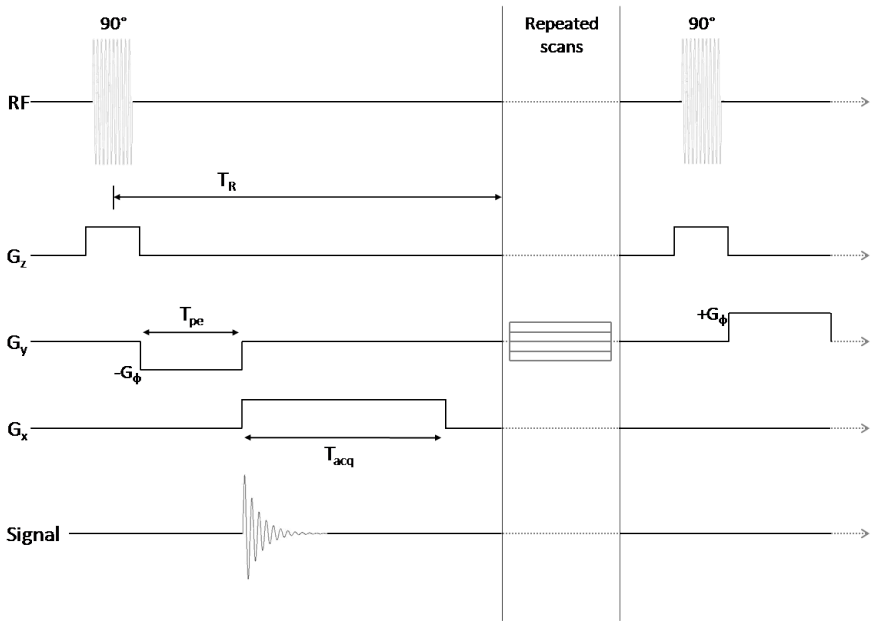


Figure 2.7: Timing diagram for rectilinear sampling without echo.

Echo generating pulse sequences

The pulse sequence described above only captures data with positive k_x positions. To collect the other half of k-space, the phase encoding step can be adjusted such that data sampling starts at position (k_{x_0}, k_{y_0}) where $k_{x_0} < 0$. Therefore, a gradient along the x-axis with a negative value, $G_{x_{pe}} < 0$, is applied with the y-gradient during phase encoding. To ensure that the signal is captured at the same k_x positions for each row of acquisition, $G_{x_{pe}}$ remains the same for all phase encoding steps. In addition, the x-gradient used during frequency encoding has the same magnitude as $G_{x_{pe}}$, but is positive. This produces an echo in the signal where the maximum amplitude is centred at (0,0) in k-space. During phase encoding, the magnetization vectors along the x and y axes precess out of phase from each other. The effects of $G_{x_{pe}}$ on the phase angles of the vectors is given by looking at only the x-component of equation (2.31). When the sign of the x-gradient switches in frequency encoding, vectors with smaller phase angles precess at higher frequencies and vice versa. Therefore, the phases of the vectors line up after $G_{x_{fe}}$ is applied for a period of $T_{acq}/2$ where T_{acq} is equal to $2T_{pe}$. A peak signal is reached before the vectors start to dephase again in the x-direction and an echo is produced. This imaging sequence is known as gradient echo and its timing diagram for rectilinear sampling is shown in Figure 2.8. T_E (echo time) is the time measured from the application of the RF pulse to when the echo peaks and T_R (repetition time) is the time between RF pulses for multiple row acquisitions. In addition to the spin density, the T_1 and T_2 constants for a packet of spins are dependent on the tissue content and affect signal generation. The maximum amplitude of the echo

signal satisfies

$$\mathrm{d}S(\mathbf{r},t) \propto \rho(\mathbf{r})(1-e^{-T_R/T_1(\mathbf{r})})e^{-T_E/T_2^*(\mathbf{r})}\mathrm{d}\mathbf{r}. \tag{2.37}$$

Between repetitions of the pulse sequence, the z-component of a particular magnetization vector may not be fully recovered (depending on T_1) and the signal contribution generated by another RF pulse would be less (refer to equation (2.19)). Also, a duration of T_E elapses after RF excitation before the signal is acquired, and the x and y components of the vector have decayed at a rate of T_2^*.[4, 5]

An echo can also be produced by applying a 90° pulse followed by a 180° pulse. This is known as a spin echo sequence and its timing diagram is shown in Figure 2.9. The slice selection gradient is turned on for both pulses. After the 90° pulse, phase encoding is performed with an x- and y-gradient. The initial point in k-space is positioned at (k_{x_0}, k_{y_0}) where k_{x_0} and k_{y_0} are non-zero. At time $T_E/2$, a 180° pulse is applied, which flips the magnetization vectors such that spins that were precessing behind (i.e., have a smaller phase angle) are now ahead and vice versa. So, when the frequency encoding gradient is turned on, vectors with smaller phase angles after the 180° pulse would precess at higher frequencies provided that

$$G_{x_{fe}} = G_{x_{pe}}. \tag{2.38}$$

At time T_E, the vectors are lined up creating a peak signal before dephasing again (similar to the gradient echo sequence). Since the x-gradient does not change from phase encoding to frequency encoding, the effects of inhomogeniety in the B_0 field is minimized.[4, 5] The maximum amplitude of the echo is dependent on T_2 and not T_2^*, and

$$\mathrm{d}S(\mathbf{r},t) \propto \rho(\mathbf{r})(1-e^{-T_R/T_1(\mathbf{r})})e^{-T_E/T_2(\mathbf{r})}\mathrm{d}\mathbf{r}. \tag{2.39}$$

$\mathbf{T_1}$, $\mathbf{T_2}$ and density weighted images

As seen in equations (2.37) and (2.39), the amplitude of an echo emitted from an infinitesimal area of spins is proportional to the spin density, T_1, and T_2. Since these constants are properties of a tissue within a voxel, influenced by spin interactions, they are discerning characteristics that can be imaged. If T_R for a particular pulse sequence was long and T_E was short, then the effects of T_1 and T_2 would be minimized since $(1-e^{-T_R/T_1}) \approx 1$ and $(e^{-T_E/T_2}) \approx 1$. As a result, the contrast throughout the image would be mainly due to the density of spins. A long T_R allows sufficient time for M_z to fully recover before the next RF excitation and the signal generated by tissue samples of similar density, but different T_1 characteristics would be similar. A short T_E is not long enough for the spins to dephase and the difference in M_{xy} for samples with different T_2's would be small. An image where

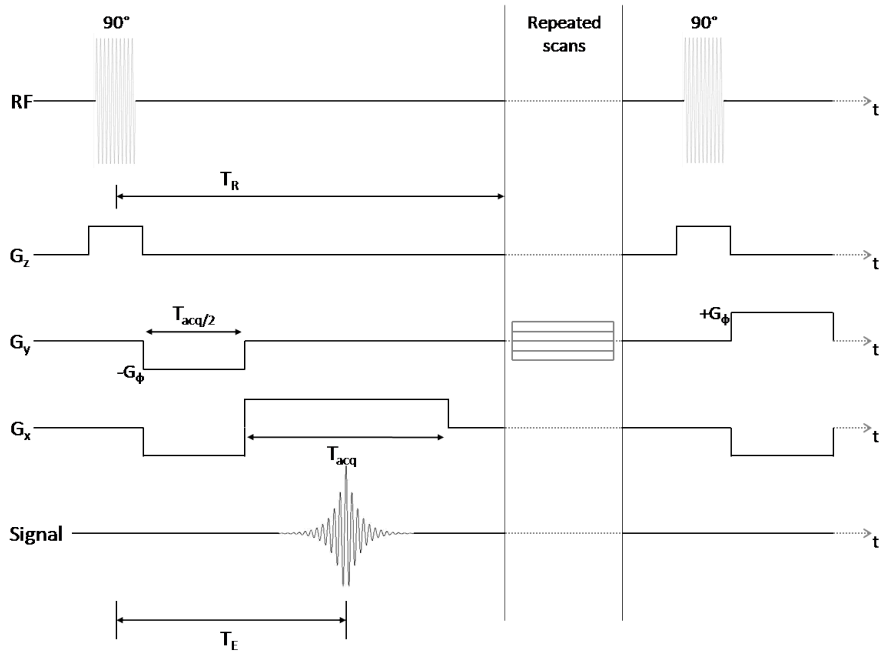


Figure 2.8: Timing diagram for gradient echo sequence, rectilinear sampling.

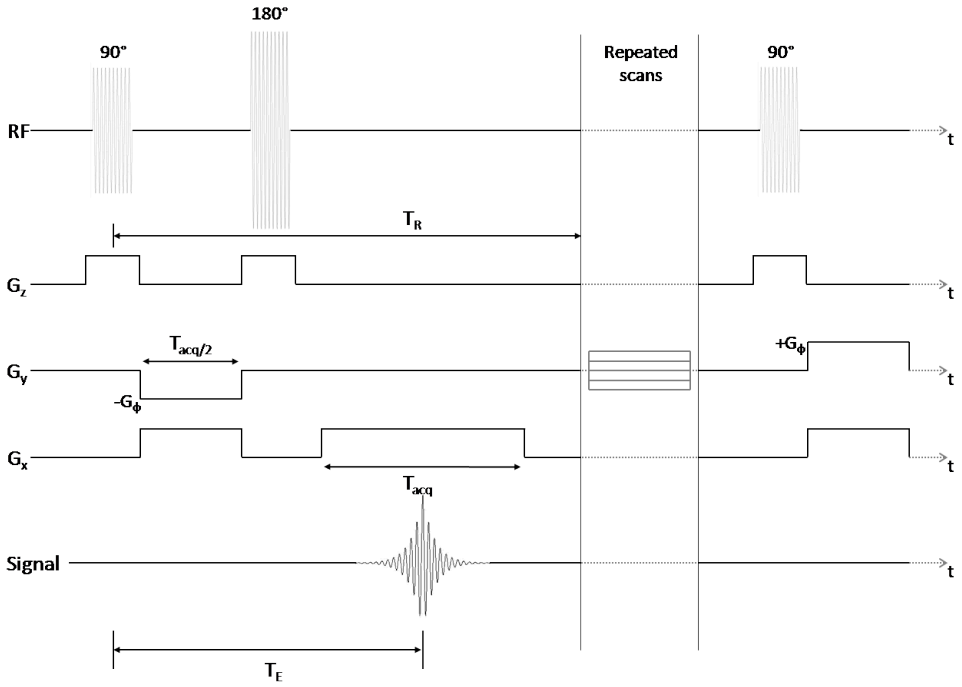


Figure 2.9: Timing diagram for spin echo sequence, rectilinear sampling.

Table 2.1: T_E and T_R Values for Different Spin Echo Imaging

	$\mathbf{T_E \leq 20\ ms}$	$\mathbf{T_E \geq 80\ ms}$
$\mathbf{T_R < 700\ ms}$	T_1-weighted	Uncommon
$\mathbf{T_R > 2000\ ms}$	Proton density weighted	T_2-weighted

the contrast is T_1-weighted is created using a short T_R and short T_E while T_2-weighted uses long T_R and long T_E. Table 2.1 shows the type of imaging achieved for different ranges of T_E and T_R for a spin echo sequence as given by [5].

2.5 Signal processing

The signal captured during a single run of frequency encoding is given by equation (2.27). The signal is amplitude modulated by a carrier frequency of f_0 (i.e., the signal was multiplied with $e^{-i2\pi f_0 t}$) and can be broken down into its real and imaginary components,

$$\Re\{S(t)\} = \int_{slice} \rho(f) \cos[2\pi(f + f_0)t] \, \mathrm{d}f \tag{2.40}$$

and

$$\Im\{S(t)\} = -i \int_{slice} \rho(f) \sin[2\pi(f + f_0)t] \, \mathrm{d}f \tag{2.41}$$

respectively, where

$$f = \frac{\omega}{2\pi}. \tag{2.42}$$

Since the signal is generated from the M_{xy} magnetization vector, the receiver coil can be placed either along the x or y axis, picking up the real or imaginary signal respectively. Capturing only the real signal is sufficient and $\mathcal{F}[S(t)]$ is conjugate symmetric where $\mathcal{F}[\cdot]$ is the Fourier transform operator. $\mathcal{F}[S(t)]$ has a positive and negative frequency spectrum centred in the $\pm$ MHz range that are mirrored. The receiver coil is often the same as the RF pulse transmitter coil because both can be located along the same axis and are required to resonate at the Larmor frequency.

Demodulation

The original signal at baseband frequency is recovered by multiplying with $e^{i2\pi f_0 t}$, which strips the carrier frequency from the signal. This is known as demodulation and creates a spectrum at $(f_0 + f_0)$ and $(f_0 - f_0)$. To remove the high frequency component, the signal is low-pass filtered. Demodulation can be performed using analog circuitry by mixing the

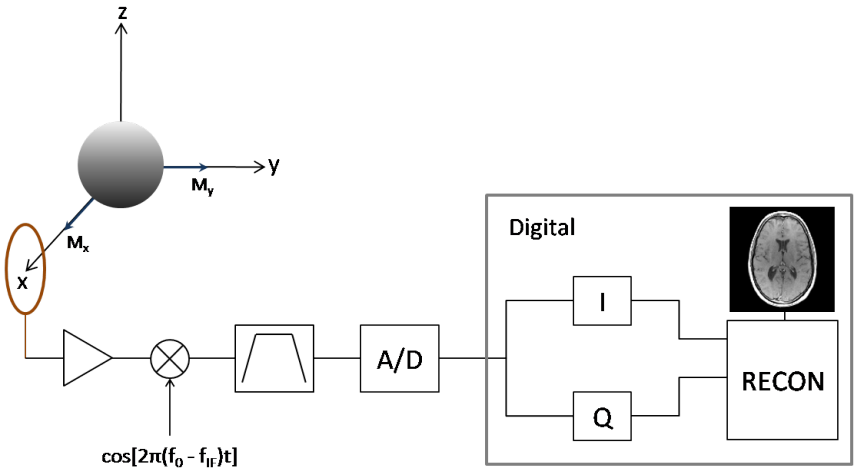


Figure 2.10: Simplified receiver chain: coil, amplifier, downmixer, bandpass filter, analog-to-digital converter, digital processor.

signal with two sinusoidal reference signals that are 90° phase shifted from each other (i.e., $\cos(2\pi f_0 t)$ and $\sin(2\pi f_0 t)$). This produces a real and imaginary channel referred to as I (in phase) and Q (in quadrature). Capturing both I and Q channels (known as quadrature detection) is required to reconstruct an image at baseband without aliasing. Mixing with only the real or imaginary reference causes the positive and negative high frequency spectra of $\mathcal{F}[S(t)]$ to overlap at baseband.

Current MRI systems prefer to digitize high frequency signals and perform demodulation using software or digital circuitry. Figure 2.10 shows a simplified receiver chain that is typical of an MRI system. The signal can be downmixed to an intermediate frequency (IF) and bandpass filtered before digitization.

Sampling requirements

According to the sampling theorem [6, 7], a continuous signal can be uniquely determined from its samples,

$$s_n = S(nT_s), \qquad \text{for } n = 0, \pm 1, \pm 2, \ldots \quad ,$$

where the sampling frequency (f_s) is $1/T_s$, if the signal is bandlimited in the frequency domain (necessary condition) and $f_s > 2f_{max}$ (sufficient condition); f_{max} is the maximum non-zero frequency component of $\mathcal{F}[S(t)]$, centred at 0 Hz, referred to as the Nyquist rate. When a signal is sampled by an impulse train of frequency, f_s, its corresponding spectrum is repeated every f_s in the frequency domain. A sampling frequency of at least $2f_{max}$ ensures that there will be no-overlaps between repetitions, which allows the signal to be

fully recovered without aliasing. For a band-pass signal not centred at 0 Hz, namely

$$X(f) = \mathcal{F}[S(t)]$$

and

$$X(f) = 0 \quad \text{for } f \notin [f_1, f_2] \cup [-f_2, -f_1],$$

a sampling frequency of twice the bandwidth,

$$BW_s = f_2 - f_1, \tag{2.43}$$

is sufficient to ensure that there are no-overlaps in the spectrum as derived in [7].

Since current MRI systems digitize RF signals before demodulation, the sampling frequency is only required to be twice the bandwidth of the image. The bandwidth of an image is determined by the frequency-encoding. It is common to oversample signals and then decimate by averaging every set number of samples or only keeping every n^{th} sample.[4]

To reconstruct a bandlimited signal, ideally, an infinite number of samples is required since a signal can not truly be bandlimited and time limited. For example, the Fourier Transform of a rectangular spectrum is a sinc function defined for an infinite period of time. Any bandlimited signal can be equated to having a rectangular bandpass of appropriate length multiplied with its spectrum, and therefore, convolved with a sinc function for all time. Fortunately, many signals, such as those generated in MRI, decay over time and can be considered almost time limited. In this case, a limited set of discrete samples are sufficient to reasonably reconstruct an accurate image.

2.6 Image reconstruction

In Section 2.4, it was shown that signal is captured in k-space, which is the Fourier space of a density, T_1, or T_2 contrast weighted image. Assuming rectilinear sampling and a spin echo sequence, multiple lines of signal are acquired during 2D imaging and arranged into rows in k-space. The signal peaks of each row should be centred at $k_x = 0$. Once the data has been re-arranged in k-space, an inverse Fourier transform is required for image reconstruction,

$$\mathcal{I}(x, y) = \iint_{-\infty}^{\infty} S(k_x, k_y) e^{i2\pi(k_x x + k_y y)} \, \mathrm{d}k_x \mathrm{d}k_y. \tag{2.44}$$

If the signal was demodulated to baseband prior to reconstruction then the resulting image would be centered at point (0,0).

Discrete Fourier transform

Assuming a Fourier series representation and a set of discrete uniform pixels in the image, the discrete Fourier transform (DFT),

$$s_{n_{kx},m_{ky}} = \sum_{n_x=0}^{N-1} \sum_{m_y=0}^{M-1} p_{n_x,m_y} e^{-i2\pi(n_{kx}n_x/N + m_{ky}m_y/M)}, \tag{2.45}$$

can be used where

$$\begin{aligned} p_{n_x,m_y} =& \mathcal{I}(x(n_x), y(m_y)), \\ x(n_x) =& x_0 + n_x \Delta x, \\ y(m_y) =& y_0 + m_y \Delta y, \end{aligned}$$

and Δx and Δy are the pixel resolutions along the x and y axes respectively. For $N_d \times M_d$ rectilinear measurements in k-space, the following system of $N_d \times M_d$ linear equations is formed:

$$\begin{gathered} s_{n_{kx},m_{ky}} = S(k_x(n_{kx}), k_y(m_{ky})) \\ \text{for} \quad n_{kx} = 0, \ldots, N_d - 1 \text{ and } m_{ky} = 0, \ldots, M_d - 1. \end{gathered} \tag{2.46}$$

The positions, $k_x(n_{kx})$ and $k_y(m_{ky})$, given by

$$k_x(n_{kx}) = k_{x_0} + \frac{\gamma G_{x_{fe}} n_{kx} T_s}{2\pi} \tag{2.47}$$

and

$$k_y(m_{ky}) = \frac{\gamma[-G_\phi + 2m_{ky}G_\phi/(M_d - 1)]T_{pe}}{2\pi}, \tag{2.48}$$

are derived from equations (2.32) to (2.35) for M_d different phase encodings of a rectilinear sampling sequence. Amplitudes at $N_d \times M_d$ image pixels can then be uniquely solved from the system of equations. A closed form solution often used in MR image reconstruction is given by the inverse DFT,

$$\begin{gathered} p_{n_x,m_y} = \frac{1}{N_d M_d} \sum_{n_{kx}=0}^{N_d-1} \sum_{m_{ky}=0}^{M_d-1} s_{n_{kx},m_{ky}} e^{i2\pi(n_{kx}n_x/N_d + m_{ky}m_y/M_d)} \\ \text{for} \quad n_x = 0, \ldots, N_d - 1 \text{ and } n_y = 0, \ldots, M_d - 1. \end{gathered} \tag{2.49}$$

The DFT and inverse DFT apply only to uniformly sampled data that fall onto a Cartesian grid (i.e., rectilinear sampling).[6, 7]

Fast Fourier transform (FFT) algorithms are implementations of the DFT that are well developed and have a computational complexity of $N_d \log N_d$ for N_d samples. Common FFT algorithms (e.g., Cooley-Tukey algorithm and variants) recursively divide samples into subsets so that the DFT can be computed on these smaller sets before recombining.[7] For a 2D image, an FFT can be applied along each dimension one at a time (i.e., perform an FFT along k_x followed by FFT along k_y).

Magnitude image

When the peak signal is centred at (0,0) in k-space the image contains only real values. Referring to the frequency shifting property of the Fourier transform[6], a shift in the signal along the k_x or k_y direction (by K or L samples) corresponds to a multiplication with a complex exponential ($e^{i2\pi Kn_x/N_d}$ and $e^{i2\pi Ln_y/M_d}$ respectively) in the image. Therefore, when k-space is not properly centred, the image contains both real and imaginary values. The magnitude and phase of the image can be calculated as the following:

$$|p_{n_x,m_y}| = \sqrt{\Re\{p_{n_x,m_y}\}^2 + \Im\{p_{n_x,m_y}\}^2} \tag{2.50}$$

and

$$\arg p_{n_x,m_y} = \tan^{-1}\left(\frac{\Im\{p_{n_x,m_y}\}}{\Re\{p_{n_x,m_y}\}}\right). \tag{2.51}$$

The magnitude image is often the desired reconstruction since it is invariant to shifts in k-space.

Discrete Fourier transform properties

The pixel size and step size in k-space are inversely proportional. Simplifying the step sizes in k-space as Δk_x and Δk_y, [5] approximates the integral in equation (2.44) as

$$\mathcal{I}(n_x\Delta x, m_y\Delta y) = C\sum_{n_{kx}=-\infty}^{\infty}\sum_{m_{ky}=-\infty}^{\infty} S(n_{kx}\Delta k_x, m_{ky}\Delta k_y)e^{i2\pi(n_{kx}n_x\Delta k_x\Delta x + m_{ky}m_y\Delta k_y\Delta y)} \tag{2.52}$$

for a series of impulses in k-space and a constant, C. By comparing the exponential terms to that of the inverse DFT (equation (2.49)) the following relationship exists:

$$e^{i2\pi(n_{kx}n_x\Delta k_x\Delta x + m_{ky}m_y\Delta k_y\Delta y)} = e^{i2\pi(n_{kx}n_x/N_d + m_{ky}m_y/M_d)}$$

and

$$n_{kx}n_x\Delta k_x\Delta x + m_{ky}m_y\Delta k_y\Delta y = n_{kx}n_x/N_d + m_{ky}m_y/M_d.$$

Therefore,

$$\Delta k_x\Delta x = \frac{1}{N_d} \tag{2.53}$$

and

$$\Delta k_y\Delta y = \frac{1}{M_d}. \tag{2.54}$$

A wider extent in k-space corresponds to a finer resolution (smaller pixel sizes) in the image. The sampling theorem is also reiterated since the furthest pixels of an image reconstructed using the DFT are located at $(\pm N_d\Delta x/2, \pm M_d\Delta y/2)$ for sampling rates of $1/\Delta k_x = N_d\Delta x$ and $1/\Delta k_y = M_d\Delta y$ in k-space. Table 2.2 lists other important Fourier Transform properties to be considered when reconstructing MR images.

Table 2.2: Discrete Fourier Transform Properties

Description	**K-space**	**Image**
1) Pixel resolution	$\frac{1}{\Delta k_x N_d}$	Δx
	$\frac{1}{\Delta k_y M_d}$	Δy
2) Shift in k-space	$s_{n_{kx}-K,m_{ky}-L}$	$e^{i2\pi(Kn_x/N_d+Lm_y/M_d)}p_{n_x,m_y}$
3) Shift in image	$e^{-i2\pi(Kn_{kx}/N_d+Lm_{ky}/M_d)}s_{n_{kx},m_{ky}}$	p_{n_k-K,m_y-L}
4) Parseval's theorem	$\frac{1}{N_d M_d}\sum_{n_{kx}=0}^{N_d-1}\sum_{m_{ky}=0}^{M_d-1}\lvert s_{n_{kx},m_{ky}}\rvert^2$	$\sum_{n_x=0}^{N_d-1}\sum_{m_y=0}^{M_d-1}\lvert p_{n_x,m_y}\rvert^2$
5) DC offset	$\frac{1}{N_d M_d}\sum_{n_{kx}=0}^{N_d-1}\sum_{m_{ky}=0}^{M_d-1}s_{n_{kx},m_{ky}}$	$p_{0,0}$
6) Power of image	$s_{0,0}$	$\sum_{n_x=0}^{N_d-1}\sum_{m_y=0}^{M_d-1}p_{n_x,m_y}$

Chapter 3

Accelerated Imaging Techniques

The length of signal produced and the number of samples acquired during a single scan (one run of the pulse sequence) is limited due to the decay in magnetizaton vectors. Therefore, multiple acquisitions may be required to adequately sample k-space. In a spin echo imaging sequence, the repetition time (T_R) can be greater than 2000 ms for a proton density or T_2 weighted image (refer to Table 2.1). This corresponds to a total imaging time of over 8 minutes for a typical 2D image with 256 rows. Time becomes a limiting factor when imaging constantly moving organs, such as the heart, and 8 minutes may be too long to avoid motion artifacts. So, there is a great desire to develop techniques that will accelerate imaging without deteriorating the quality of reconstructed images. This chapter shall review the state of art of accelerated imaging techniques and corresponding image reconstruction methods used in MRI.

3.1 Rectilinear undersampling with parallel imaging

An obvious way to achieve faster imaging is to reduce the number of rows acquired in rectilinear sampling resulting in fewer scan repetitions and gradient changes. To preserve image resolution, the interval between samples in the k_y direction, Δk_y, is increased for a reduced number of samples, M_d. Therefore, larger steps are taken between phase encodings so that fewer scans are required to progress from $-G_\phi$ to $+G_\phi$. This effectively reduces the sampling rate along k_y and the bandwidth of the image in the y-direction that can be resolved. The resolvable bandwidth is often referred to as the field of view (FOV). The factor by which the number of row acquisitions is reduced is referred to as the acceleration rate. An imaging sequence with an acceleration rate of a (i.e., $\Delta k_{y_{acclr}} = a\Delta k_y$) only scans every a^{th} row in k-space. Also, the number of phase encoding steps and the FOV is reduced by a factor of a. For an image where the tissue content fills the original FOV, increasing Δk_y will result in unwanted artifacts since the sampling rate no longer satisfies Nyquist

and copies of the image at higher frequencies are aliased in.[8] Figure 3.1 shows the result of reconstructing an accelerated dataset using an FFT.

Parallel imaging (pMRI) uses multiple receiver coils to capture signal and provides additional information that can be used to properly reconstruct images from undersampled datasets. This results in multiple signals that are processed simultaneously using a separate receiver chain for each coil. The coils have an associated sensitivity profile (c), resembling a bandpass filter, that is tuned to a specific region in the image (recalling that the image is frequency encoded). Therefore, images obtained from pMRI are filtered such that only a band is visible (at full intensity) and the image fades away in regions that the coil is not sensitive to. A final image is often produced by summing the squares of the magnitude images from each coil. If N_c is the number of coils and $\mathcal{I}_k$ is the reconstructed image from the k^{th} coil then the combined image is given by

$$\mathcal{I}_{pMRI} = \sqrt{\sum_{k=1}^{N_c} |\mathcal{I}_k|^2}. \tag{3.1}$$

The resulting pMRI image has an overall SNR improvement.[9]

Partially parallel imaging with localized sensitivities

Partially parallel imaging with localized sensitivities (PILS) is a technique for undersampling that uses pMRI. Surface coils are strategically arranged such that the sensitivity profiles line up to cover the full FOV of an image with minimal overlap. Each coil is responsible for imaging a specific region. The signal from each coil is undersampled in the k_y direction as previously described and reconstructed separately (often by means of an FFT).[10] If the bandwidth of the sensitivity profile, BW_c, for every coil is still smaller than the reduced FOV of the image after undersampling then the aliasing occurs outside the band limits. Therefore, the part of the image that the coil is localized to remains uncorrupted. Images from each coil are cropped and stitched together to form the final image. The acceleration rate that can be obtained depends on BW_{c_k} for all N_c coils. If $\max_{k=1,\ldots,N_c}[BW_{c_k}]$ is less than half the FOV, then an acceleration factor of at least 2 is attainable.[8]

Sensitivity encoding

Sensitivity encoding (SENSE) is a reconstruction method that does not rely on coil configuration. According to [11], a pixel in an image distorted by undersampling is composed of intensities of different pixels outside the reduced FOV that has been aliased in. Given an undistorted image with an original step size of Δk_y, if an acceleration factor of a along k_y

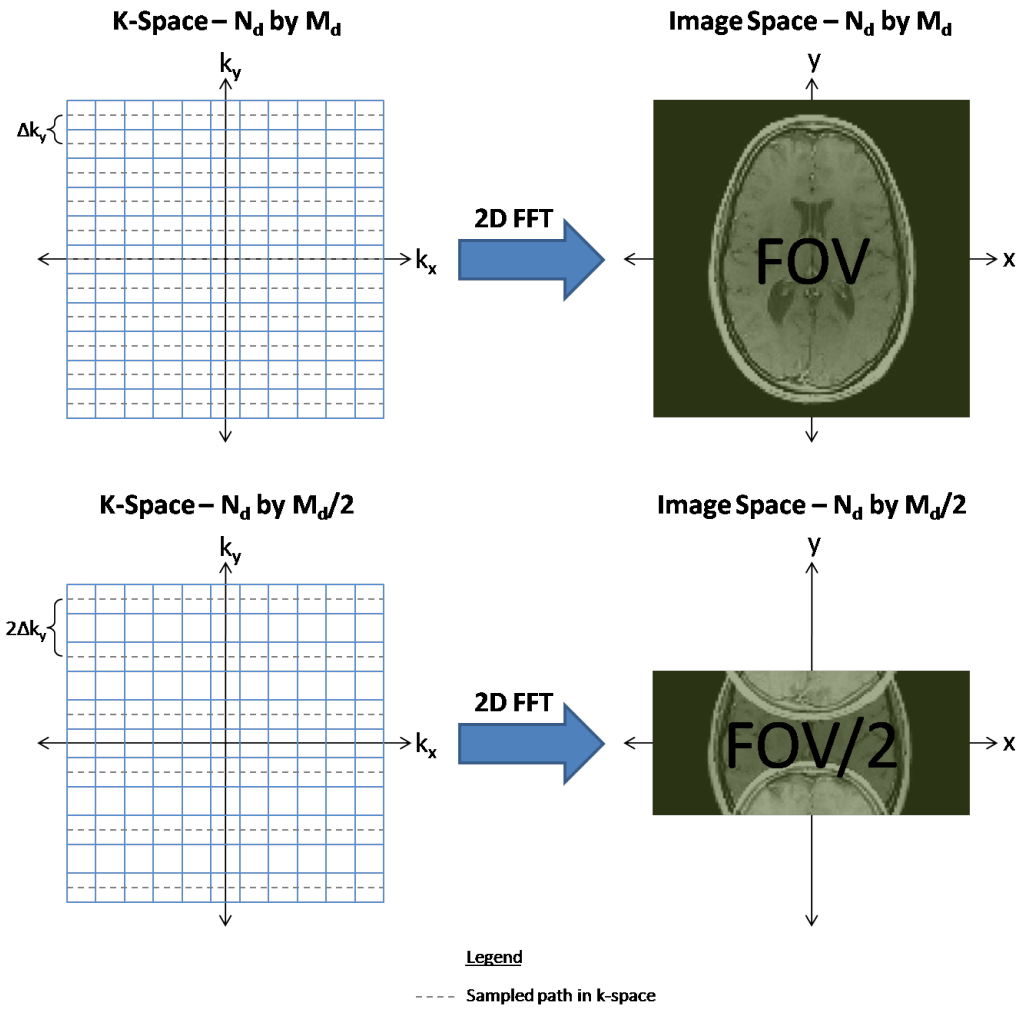


Figure 3.1: Effects of accelerated imaging by undersampling in k_y direction; an acceleration rate of 2 is shown.

was performed then pixels separated by $1/(a\Delta k_y)$ would overlap creating a reduced FOV. However, in pMRI, the pixel values are scaled by c and a weighted sum is used to more accurately describe the result of the overlapped pixels. Given a reconstructed image from an undersampled dataset received from coil k, $\tilde{\mathcal{I}}_k$, the value of each aliased pixel, $\tilde{p}_{k_n}$, can be expressed as follows (as given by [8]).

$$\tilde{p}_{k_n} = \sum_{l=1}^{a} c_{k,l} p_l, \qquad \text{for} \quad \tilde{p}_{k_n} \in \tilde{\mathcal{I}}_k,\ n = 1, ..., N_d M_d / a \qquad , \tag{3.2}$$

where the values of p belong to a unaliased pixels seperated by $1/(a\Delta k_y)$ and c_k a set of corresponding sensitivity factors of coil k. Therefore, there are a unknowns associated with each pixel in $\tilde{\mathcal{I}}_k$ and a equations are required to solve for all p's. Taking advantage of pMRI, a coils with different sensitivity profiles can be used to give the necessary system of equations,

$$\tilde{\mathbf{p}}_n = \mathbf{C}\mathbf{p}, \tag{3.3}$$

where $\mathbf{C}$ is the coefficient matrix derived from the coil sensitivities, $\mathbf{p}$ is a vector containing the a unknown pixel values, and $\tilde{\mathbf{p}}_n$ is a vector containing $\tilde{p}_{k_n}$ measured from $\mathcal{I}_k$ for $k = 1, \ldots, a$. The unaliased image (consisting of $N_d M_d$ pixels) is reconstructed by solving equation (3.3) $N_d M_d / a$ times. SENSE allows for greater acceleration than PILS because it is applicable to situations where the bandwidth of c is larger than the reduced FOV. However, SENSE is limited by the number coils that a scanner can accommodate at once.

Generalized autocalibrating partially parallel acquisitions

The method described in [12], generalized autocalibrating partially parallel acquisitions (GRAPPA), attempts to estimate missing data in k-space before performing image reconstruction of undersampled datasets obtained from pMRI. This is different from the strategy used by SENSE, which corrects for undersampling in the spatial domain of the image. GRAPPA is an extension of SMASH (Simultaneous Acquisition of Spatial Harmonics), which showed that rows of signal from multiple coils can estimate a shifted row in the signal of the composite image using

$$S^{comp}(k_x, k_{y_{ref}}) = \sum_{k=1}^{N_c} \eta_{m,k} S_k(k_x, k_{y_{ref}} + m\Delta k_y). \tag{3.4}$$

For a set of data in k-space obtained in pMRI with accelerated imaging as discussed in previous sections, the corresponding composite signal would also be undersampled. To estimate the missing rows, additional ACS (auto-calibration signal) lines located at $k_{y_{ref}}$ from every coil are collected. These ACS lines are used for calibration and sum together

to produce an estimate for $S^{comp}(k_x, k_{y_{ref}})$. Equation (3.4) is then used to fit rows that were collected during undersampling located at $k_{y_{ref}} + m\Delta k_y$, for some m, from every coil to the estimated composite signal by solving for the linear weights, $\eta_{m,k}$, in

$$\sum_{k=1}^{N_c} S^{ACS}(k_x, k_{y_{ref}}) = \sum_{k=1}^{N_c} \eta_{m,k} S_k(k_x, k_{y_{ref}} + m\Delta k_y).$$

The set of $\eta_{m,k}$'s relates any row in the composite signal to a row located $m\Delta k_y$ away in each coil. Therefore, the rest of k-space of the composite image can then be estimated using equation (3.4), the calibrated weights, and corresponding rows of signal from coils 1 to N_c. GRAPPA uses a similar strategy to estimate missing rows in k-space, but replaces the composite signal in equations (3.4) with signal from a single coil. That is,

$$S_l^{ACS}(k_x, k_{y_{ref}}) = \sum_{k=1}^{N_c} \eta_{m,k,l} S_k(k_x, k_{y_{ref}} + m\Delta k_y), \qquad \text{for } l = 1, \ldots, N_c. \tag{3.5}$$

Missing data in one coil is estimated using data from all N_c coils. To increase accuracy, a neighbourhood, N_G, consisting of several rows of k-space from all coils can be defined as shown in the example in Figure 3.2a. The entire neighbourhood of data is fitted to a single ACS line located in coil l using

$$S_l^{ACS}(k_x, k_{y_{ref}}) = \sum_{j \in \mathrm{N}_G} \sum_{k=1}^{N_c} \eta_{j,k,l} S_k(k_x, j), \qquad \text{for } l = 1, \ldots, N_c. \tag{3.6}$$

The resulting weights form a kernel that can be used to estimate any missing row in coil l from a neighbourhood that is located in the same relative position that was defined in the fitting procedure. A set of weights is calibrated for each coil. If 2D datasets obtained in k-space from all coils were stacked to form a three dimensional data matrix and the kernel was organized accordingly, then the estimation can be viewed simply as a convolution between the kernel and the data matrix as shown in Figure 3.2b. Once complete datasets in k-space are estimated for each coil, images are reconstructed using an FFT and combined (via a sum of squares) as in regular (non-accelerated) pMRI. GRAPPA can be extended to included more than one ACS line during the fitting procedure, which allows for greater flexibility and accuracy. Each ACS line corresponds to a set of calibrated weights that are used to estimate data where appropriate. In addition, GRAPPA can be applied to estimating single data points at a time rather than entire rows by fitting a concentrated neighbourhood of data to a single point in the ACS using

$$S_l^{ACS}(k_{x_{ref}}, k_{y_{ref}}) = \sum_{i,j \in \mathrm{N}_G} \sum_{k=1}^{N_c} \eta_{i,j,k,l} S_k(i, j), \qquad \text{for } l = 1, \ldots, N_c. \text{ [8, 12]} \tag{3.7}$$

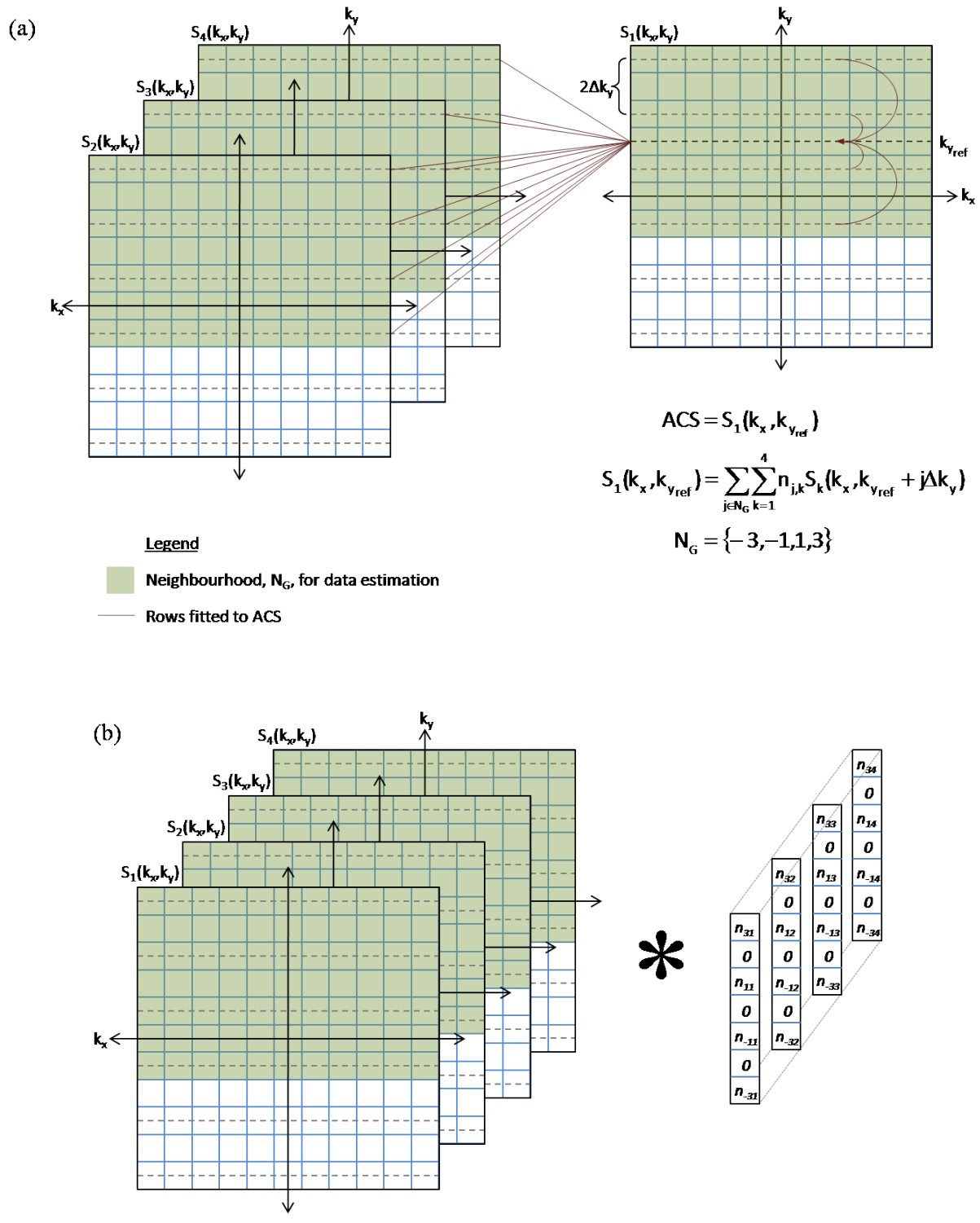


Figure 3.2: Example of GRAPPA applied to undersampled k-space with an acceleration of 2; a) Fitting procedure with neighbourhood N_G and b) convolution for data estimation using calibrated weights.

3.2 Compressed sensing

Compressed sensing (CS) takes advantage of the fact that images often contain redundant information and, as such, it may be possible to undersample without losing important information. Specifically, CS accounts for sparsity in an image when collecting data. The image may not be sparse in the spatial domain, but there may be another domain in which it is sparse.

An image can be expressed as a linear combination of orthonormal basis vectors. Consider a matrix Ψ containing the basis vectors, a vector $\boldsymbol{\alpha}$ the coefficients, and vector $\mathbf{p}$ the image pixels. The matrix form of this linear combination is given as

$$\mathbf{p} = \Psi^{\mathsf{T}}\boldsymbol{\alpha} \tag{3.8}$$

where

$$\boldsymbol{\alpha} = \Psi\mathbf{p}. \tag{3.9}$$

Ψ is the orthogonal transformation matrix that maps the image domain to the domain of $\boldsymbol{\alpha}$. If $\boldsymbol{\alpha}$ contains fewer non-zero values than $\mathbf{p}$ then Ψ is referred to as a sparsifying transform. Using the ℓ_{p} norm, the sparsity of a vector is measured by some constant $R > 0$ where

$$\|\boldsymbol{\alpha}\|_{\mathsf{p}} \leq R \qquad 0 < \mathsf{p} < 2. \tag{3.10}$$

Many signals and images are naturally compressible and there exists a transform Ψ where the transformed vector is sparse. In the sparse domain, it is only necessary to retain the N_Ψ largest coefficients to properly reconstruct an image with an error on the order of $(N_\Psi + 1)^{1/2-1/\mathsf{p}}$.[13] This is the premise behind data compression. Images with $N_d M_d$ pixels are transformed to a sparse domain where only N_Ψ coefficients ($N_\Psi << N_d M_d$) are obtained for storage or transmission.

Coded acquisition

The desired image is often unknown and therefore, $\boldsymbol{\alpha}$ cannot be obtained directly. Coded acquisition allows for certain sampling schemes to indirectly obtain $\boldsymbol{\alpha}$. This refers to capturing data in another domain through a transformation separate from Ψ, namely

$$\mathbf{y} = \Phi\mathbf{p}, \tag{3.11}$$

where $\mathbf{y}$ are measurements made in the transformed data acquisition domain. When measuring $\mathbf{y}$, there exist sampling sequences where the aliasing formed by undersampling appears noise-like in the sparse domain and the important features are still distinguishable.[14]

Measurements that result in noise-like artifacts in the sparse domain are referred to as incoherent. Figure 3.3 shows examples of incoherent measurements acquired in the Fourier domain of an image where the image itself is sparse and Ψ is the identity matrix. An image of a point was Fourier transformed and only 4 % of the data were sampled in four different ways (random, linear, spiral, and radial). The unsampled data were zeroed before reconstructing the images by taking the inverse Fourier transform (via an inverse FFT). The artifacts when viewed in the cross section of the image appear noise-like for random, spiral, and radial sampling trajectories and a single peak can be found. On the other hand, the linearly sampled image has been aliased such that a single point can not be separated. In this case, measurements taken from random, spiral, and radial sampling are more incoherent than the ones sampled linearly. The number of incoherent measurements, N_Φ, required to approximate $\mathbf{p}$ as effectively as using the N_Ψ coefficients is $N_\Psi \log(N_d M_d)$ as derived in [13], and $N_\Phi < N_d M_d$. Therefore, the coefficients necessary to reconstruct a compressed image are sensed by incoherent measurements made using coded acquisition rather than directly acquiring the sparse vector. The image can be reconstructed using interference cancellation by solving the optimization problem,

$$\begin{aligned} &\min_{\mathbf{p}} \|\Psi \mathbf{p}\|_1 \\ s.t. \quad &\|\Phi \mathbf{p} - \mathbf{y}_{measured}\|_2 < \epsilon, \end{aligned} \tag{3.12}$$

assuming that the image is represented by the sparsest vector.

Application in MRI

In MRI, k-space acquisition can be considered as coded acquisition where Φ is the set of complex exponentials ($e^{-i2\pi(k_x x + k_y y)}$) used to transform the image from the spatial domain to the Fourier domain. This framework and the flexibility in creating different sampling trajectories using gradients allow for the application of CS. Most MR images are also considered compressible (i.e., there exists a domain where the image is sparse). Depending on the image, a different sparsifying transform may be required. Table 3.1 lists a few common MR images and their appropriate sparsifying transform as given by [14]. The key is to find a sampling scheme for a particular image that will give incoherent measurements. In general, random sampling can provide incoherent data. However, truly random sampling is difficult to achieve due to limitations in magnetic field gradients and restrictions imposed by government agencies (e.g., Health Canada and US FDA). There are guidelines set out to control the rate of change in a magnetic field and the amount of RF energy absorption (SAR) that a patient can be subjected to.[4] Other non-Cartesian sampling trajectories that are feasible can result in incoherent and noise-like artifacts (e.g., radial and spiral) as

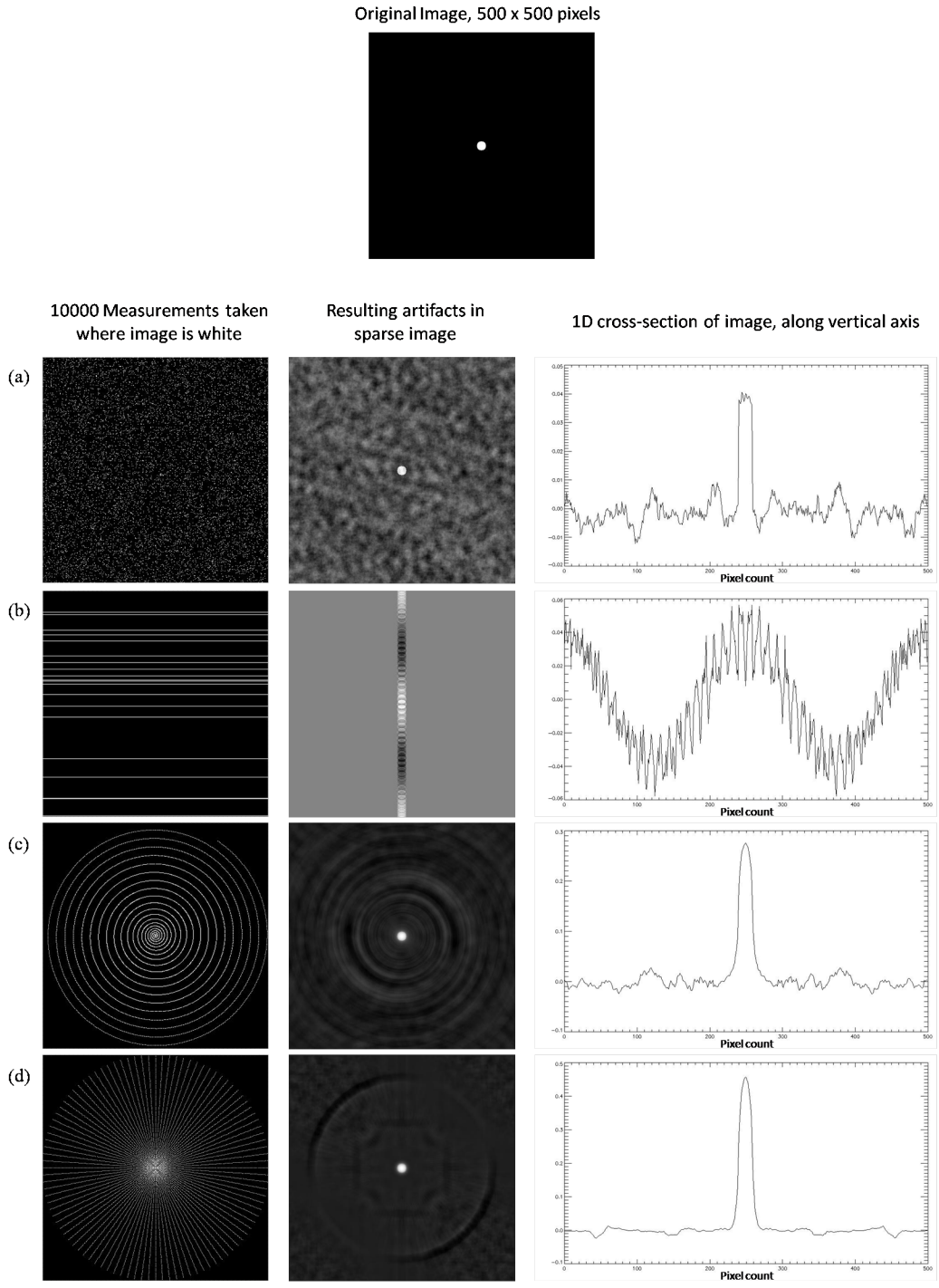


Figure 3.3: Different sampling schemes applied in the Fourier domain of a sparse image and their resulting artifacts; a) random, b) linear, c) spiral, and d) radial.

Table 3.1: Sparsifying Transforms for Common MR Images

Type of Image	**Sparsifying Transform**	**Comment**
Angiogram	Identity	Image is sparse
Brain	Spatial finite-differencing	Image is smooth
Dynamic MR images (e.g., video of a heart)	Temporal finite difference	Movements are almost periodic

previously demonstrated. The point spread function (PSF) of a sampling trajectory as seen in the sparse domain can be used to determine the amount of incoherence and suitability to CS. Therefore in MRI, the right combination of a sparsifying transform and a sampling trajectory in k-space is necessary for CS. The image is reconstructed by solving equation (3.12). CS requires fewer samples and accommodates interesting acquistion schemes that are nonlinear, which can translate to faster imaging.

3.3 Non-rectilinear sampling trajectories

Recall from Section 2.4 that during data acquisition, points along a sampling trajectory in k-space are captured. The path traversed in k-space is determined by the magnetic field gradients used in frequency and phase encodings as expressed in equations (2.32) to (2.35). Therefore, interesting sampling sequences can be created by manipulating the gradients. As discussed in Section 3.2 the flexibility in exploring different sampling schemes is required for CS.

Radial, spiral and other nonuniform (i.e., does not produce equidistant samples) k-space trajectories are advantageous in certain circumstances. Radial and spiral sampling methods can be used as a way to filter out high spatial frequency components in k-space by providing denser coverge near the origin. This results in the image being blurred from signal averaging, but a reduction in motion artifacts is gained.[5] One shot sequences are also possible with spiral sampling, which leads to faster imaging.

Radial sampling

In rectilinear sampling, only the x-gradient is turned on during frequency encoding to traverse horizontally across k-space. If the y-gradient is turned on as well then the resulting k-space trajectory would be a line with slope $G_{y_{fe}}/G_{x_{fe}}$. Radial sampling (refer to Figure 3.4) acquires data along lines that are rotated about point (0,0) in k-space. At each phase encoding step, the x and y gradients are set to $G_\phi \cos(m\varphi)$ and $G_\phi \sin(m\varphi)$ respectively for

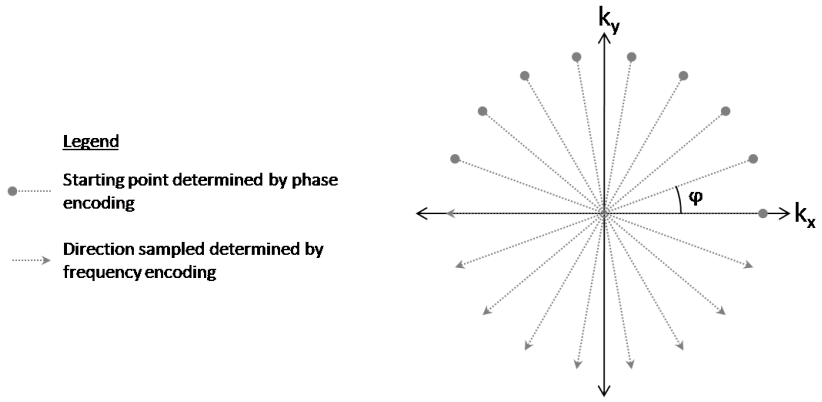


Figure 3.4: Example of k-space trajectory in radial sampling.

some constant G_ϕ. The pulse sequence is repeated as m steps from 0 to $M_d - 1$. Therefore, the start of each line of acquisition is placed around a circle separated by a degree of φ. For a spin echo sequence, $G_{x_{fe}}$ and $G_{y_{fe}}$ are equal to $G_{x_{pe}}$ and $G_{y_{pe}}$; this ensures that data acquisition passes through the origin of k-space.[2]

Spiral sampling

G_x and G_y can fluctuate during frequency encoding and the resulting k-space trajectory is defined by

$$k_x(t) = \frac{\gamma}{2\pi} \int_0^t G_{x_{fe}}(\tau)\, \mathrm{d}\tau \tag{3.13}$$

and

$$k_y(t) = \frac{\gamma}{2\pi} \int_0^t G_{y_{fe}}(\tau)\, \mathrm{d}\tau. \tag{3.14}$$

In rectilinear and radial sampling the gradients are constant resulting in a line in k-space. If the gradients have a sinusoidal waveform (i.e., $G_x = \cos(\xi t)$ and $G_y = \sin(\xi t)$) then data would be acquired along a curved path. A spiral trajectory is created using sinusoidal gradients that are amplitude and frequency modulated as shown in Figure 3.5. Phase encoding is not necessary since the sampling path can start at the origin and spiral outwards. The rate at which the amplitude of the gradients rises determines the rate at which data acquisition spirals outward and is limited by the slew rate of the gradients. Only one application of the RF pulse(s) is required to capture data that sufficiently covers k-space. This is referred to as a one-shot sequence. However, the pulse sequence can be repeated with a rotation in the gradients to form interleaved spirals for denser sampling. Spiral sampling is compatible with spin echo and gradient echo sequences.[5]

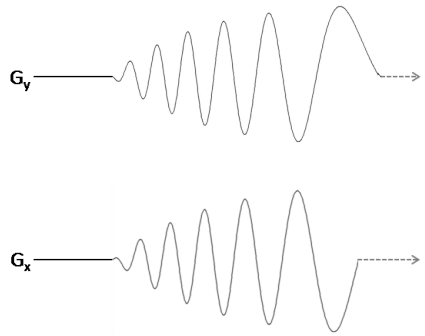


Figure 3.5: Example of sinusoidal gradients for spiral sampling.

Image reconstruction

An FFT can not be used to properly reconstruct images from nonuniform datasets. When not performing CS, images from nonuniformly sampled data are often reconstructed using interpolation or gridding methods. One of the more popular gridding methods is described in [5]. Uniform samples are interpolated from the acquired dataset before applying the Fourier transform. The interpolation is performed using convolution. When a discrete set of data is convolved with a function, the function is replicated at every impulse and scaled according to the sampled value. Therefore, a continuous mapping of k-space, $S_c(k_x, k_y)$, can be approximated by convolving with a continuous function, $g(k_x, k_y)$. If the chosen gridding function is assumed to be separable such that

$$g(k_x, k_y) = g_x(k_x)g_y(k_y), \tag{3.15}$$

then the 2D convolution can be expressed as

$$S_c(k_x, k_y) = \sum_{j=1}^{N_{samples}} s_j g_x(k_x - k_{x_j}) g_y(k_y - k_{y_j}) \Delta k_{s_j}, \tag{3.16}$$

where s_j is a sample in k-space located at (k_{x_j},k_{y_j}). The Δk_{s_j} factor in equation (3.16) is referred to as the density compensation weight, which accounts for the non-uniform spacing between points. Δk_{s_j} is obtained by estimating the area between samples. Polar coordinates can be used to calculate Δk_{s_j} as

$$\Delta k_{s_j} = k_j \Delta k_j \Delta \varphi \tag{3.17}$$

for radial and spiral samples, where k_j is the arc length from the origin to sample j in k-space. Data in between samples are approximated using a series of overlapping $g(k_x, k_y)$ functions. As such, it is practical to choose $g(k_x, k_y)$ as a function with bounded inputs referred to as a gridding kernel. A gridding kernel has a width that limits the neighbourhood of points that can be interpolated from a given sample. The Kaiser-Bessel function

is a popular gridding kernel used in MRI and is considered optimal in that it gives results similar to sinc interpolation (i.e., using sinc functions in the convolution), but is less computationally expensive.[5] To interpolate uniform data that follow a Cartesian grid, equation (3.16) is only calculated for points separated by $n\Delta k_x$ and $m\Delta k_y$ in the k_x and k_y directions respectively. A new set of data in k-space is formed that can be processed using conventional means such as an FFT.

Convolution with a gridding kernel in k-space is equivalent to a multiplication with $\mathcal{F}^{-1}[g(k_x, k_y)]$ in the image domain. Therefore, using convolution to interpolate data results in the image being filtered by the bandpass of $\mathcal{F}^{-1}[g(k_x, k_y)]$. This can be beneficial in removing artifacts created by radial or spiral sampling that often surround the object of interest.[5] As observed in Figure 3.3, undersampling effects from radial or spiral trajectories do not appear as spectral overlaps, but as aliasing around the centre of the FOV.

Chapter 4

Fourier Reconstruction of Optical Interferometer Data

In Chapter 3, techniques for reconstructing images from undersampled datasets in MRI are discussed. Algorithms such as SENSE and GRAPPA work in conjunction with parallel imaging while CS (compressed sensing) operates under the condition that a sparsifying domain exists where the sampling sequence used corresponds to incoherent measurements. The gridding method is slightly more flexible in that it can reconstruct images from nonuniformly (e.g., spiral, radial or other non-Catesian trajectories) sampled data without imposing additional hardware or imaging requirements. It interpolates data centred on the grids of a Cartesian plane from samples measured in k-space so that an FFT may be used to reconstruct images. The Fourier reconstruction of optical interferometer data (FROID) is an algorithm developed by Hajian et al. [15] that interpolates Fourier data, which do not have to follow a Cartesian grid, from a continuous mapping of an image by utilizing the integral Fourier transform. FROID can be applied to reconstruct images from data sampled in various forms and like the gridding method, it is not restriced to certain imaging sequences. In addition, it can be expanded to include *a priori* information when applicable in reconstructing images from underdetermined datasets. The flexibility that FROID offers in reconstructing images has proven to be effective in the field of optical interferometry where it has been applied in reconstructing spectra from unevenly sampled data.[15, 16] It has also been used to reconstruct astronomical images from special datasets where the amplitudes and phases of the complex Fourier data are indirectly captured through other observables known as the squared visibilities and complex closure amplitudes.[17] This thesis encompasses the work in adapting and applying FROID to reconstructing MR images.

4.1 Overview of FROID

FROID initially assumes a model spectrum or image consisting of amplitudes at discrete points which it interpolates to form a continuous function, $\mathcal{I}_m$. $\mathcal{I}_m$ is then Fourier transformed to give a model Fourier signal, S_m. This allows a set of model Fourier data, $s_{m_l}$for $l = 1, \ldots, N_{samples}$, to be sampled in S_m at locations where measurements were taken during data acquisition, which can follow any sampling trajectory. The error, χ^2, between the modelled dataset and the measured samples in the Fourier domain is calculated. The amplitudes in the model spectrum are adjusted such that χ^2 decreases and the procedure repeats. The objective is to find a spectrum or image that would correspond to Fourier data that are least squares fitted to measured values (i.e., minimizes χ^2). Therefore, the reconstruction is posed as an optimization problem that starts with an initial guess and uses the integral Fourier transform to compute data that can be fitted to acquired samples. Figure 4.1 shows a flow diagram of the steps in FROID.

Through interpolation, a continuous representation of an image is produced and the integral Fourier transform can be used rather than a DFT. Intensities between image pixels are modelled and factored into the construction of Fourier data, which more accurately represent reality since tissue variation in MRI or light distribution in optical interferometry is continuous.

4.2 Prior art in optical interferometry

Data obtained from optical interferometry are inherently sparse and nonuniform. The spectrum of a light source (e.g., a star) is imaged using a spectrometer and the absorption lines observed are used to characterize the object being studied. Data are captured in the form of an interferogram, which is related to the spectrum through a Fourier transform. Interferograms are traditionally sampled more finely near signal peaks to facilitate the location of central fringes, which are placed at the 0^{th} position for proper reconstruction of spectra; this leads to nonuniform sampling. In radio astronomy, the spatial brightness distribution of an astronomical object is also captured in the form of interferogram data obtained from an array of telescopes. The sampled data is in the Fourier domain (also referred to as the u-v plane) and an inverse Fourier transform is required to reconstruct images. The u-v plane is the Fourier transform of the image domain as k-space is in MRI.

Image reconstruction in optical interferometry

Algorithms used to reconstruct images in optical interferometry often employ an iterative procedure, and sometimes are based in finding an optimal image with respect to some

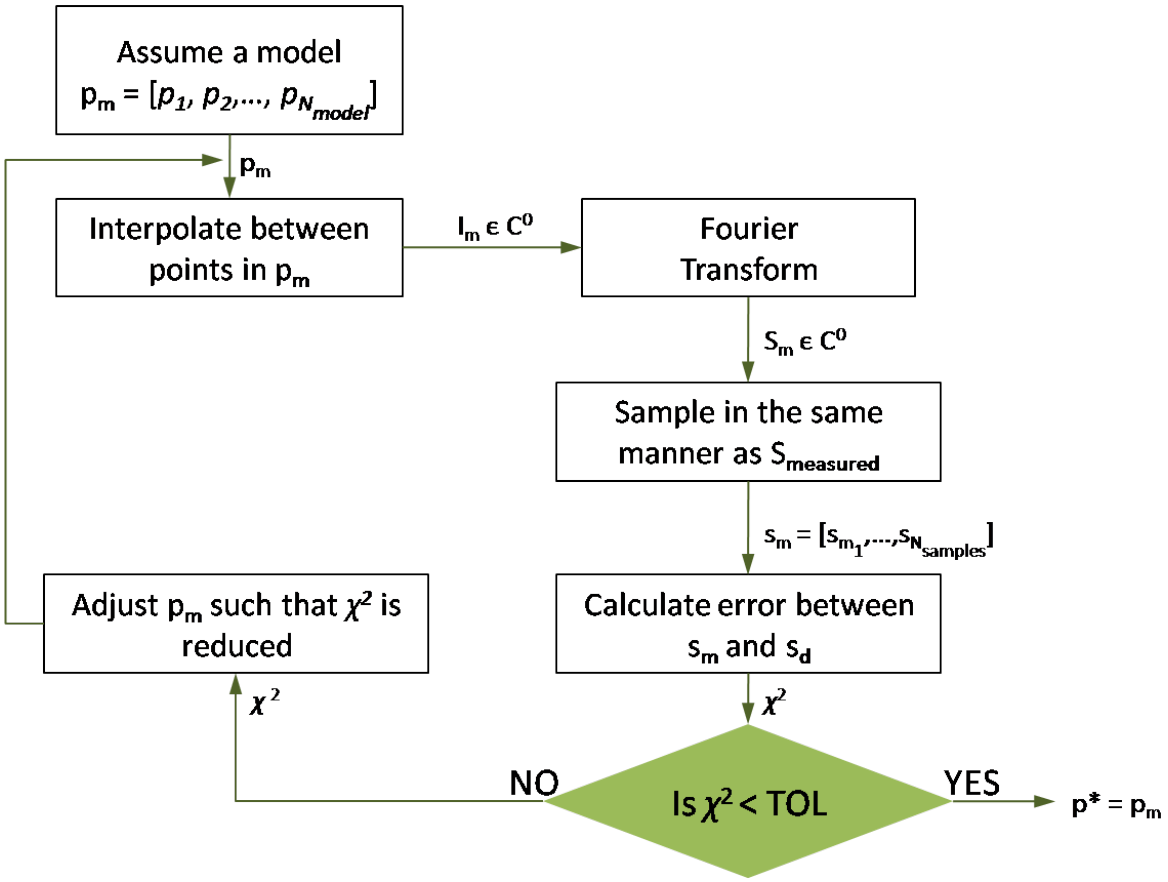


Figure 4.1: Flow diagram of FROID algorithm.

criteria. The CLEAN algorithm[18] is a popular deconvolution technique widely used in radio astronomy. It expresses an image as a series of point sources convolved with the PSF of the imaging system used. The acquired data can be expressed as a multiplication of the visible signal, $S(u, v)$, in the u-v plane with a sampling function, W. The sampling function has a value of 1 at positions where measurements are taken and zero elsewhere. For an irregular sampling structure, the corresponding inverse Fourier transform of W is known as a dirty beam. The image produced from irregularly sampled data is referred to as a dirty map, which is given by

$$\mathcal{I}_{DM} = \mathcal{F}^{-1}[S] * \mathcal{F}^{-1}[W], \tag{4.1}$$

and $\mathcal{I}_{CM}$, the clean map, which is the desired outcome of the reconstruction is given by

$$\mathcal{I}_{CM} = \mathcal{F}^{-1}[S]. \tag{4.2}$$

The clean map is obtained by deconvolving the dirty beam from the dirty image. This is performed in an iterative manner. At each iteration, the largest absolute value in the dirty image is located and a dirty beam convolved with the maxima is subtracted off:

$$\mathcal{I}_{DM}^{(n+1)}(x, y) = \mathcal{I}_{DM}^{(n)}(x, y) - \varrho \mathcal{I}_{DM}^{(n)}(x_{pk_n}, y_{pk_n}) \mathcal{F}^{-1}[W](x - x_{pk_n}, y - y_{pk_n}) \tag{4.3}$$

where

$$\mathcal{I}_{DM}^{(n)}(x_{pk_n}, y_{pk_n}) = \max\{|\mathcal{I}_{DM}^{(n)}|\} \tag{4.4}$$

and ϱ is a factor referred to as the loop gain.[18] This is repeated until $\max\{\mathcal{I}_{DM}^{(n)}\}$ is insignificant with respect to the noise floor. The collection of peaks are input back into the image and convolved with a clean beam, usually a Gaussian with parameters fitted to the dirty beam.

The CLEAN algorithm is similar to interference cancellation used in CS. Peak intensities in the image are successively collected by removing residuals associated with the previously found peak, thereby lowering the threshold used to identify real features. However, in CS the extraction of point sources was indirectly performed by solving an optimization problem. The CLEAN algorithm can be used in place of equation (3.12) in reconstructing compressed images.[19]

Another popular image reconstruction algorithm is the Maximum Entropy method (MEM). MEM assumes that the desired image is the smoothest among several possible solutions corresponding to a sparse set of interferometer data. The entropy of an image is measured as

$$H(\mathcal{I}) = -\sum_{j=1}^{N_d M_d} f_j \log(f_j), \tag{4.5}$$

where, for $p_j \in \mathcal{I}$ and $p_j \geq 0$, $j = 1, \ldots, N_d M_d$,

$$f_j = \frac{p_j}{\sum_{p \in \mathcal{I}} p}. \tag{4.6}$$

A large entropy corresponds to a smoother image since the range of intensity values among all pixels is compressed. The reconstructed image is one that maximizes $H(\mathcal{I})$ and corresponds to Fourier data that fit with the measured values. The constraint can be expressed as

$$\chi^2 = \sum_{l}^{N_{samples}} \left\{ \mathcal{F}[\mathcal{I}](u_l, v_l) - s_l \right\}^2 / \sigma_l^2, \tag{4.7}$$

where s_l is the measured value at (u_l, v_l) and σ_l^2 is the associated noise variance.[20] The image reconstructed by MEM is also restricted to having only positive values as defined in equation (4.6).

Reconstruction of spectra using FROID

The previously described algorithms assume sparsity or smoothness in an image. However, such assumptions may not apply to images consisting of a variety of features. The FROID algorithm was developed in conjunction with a spectrometer device (dispersed Fourier transform spectrometer (dFTS)) and was used to process unevenly sampled interferometer data to produce 1D spectra.[15]

The following linear interpolation was used,

$$\mathcal{I}_m(x) = \sum_{j=0}^{N_{model}} \left[p_{m_j} + (x - x_j)\Delta_j \right], \tag{4.8}$$

where

$$\Delta_j = \frac{p_{m_{j+1}} - p_{m_j}}{x_{j+1} - x_j}, \tag{4.9}$$

to form a continuous model of the spectrum, which allowed a closed form solution of the integral Fourier transform of $\mathcal{I}_m$ to be derived (in Appendix B of [15]). The Fourier data is expressed as a linear combination of the modelled spectral points,

$$s_{m_l} = \sum_{j=0}^{N_{model}} \left[\alpha_{j,l} p_{m_j} + \Delta_j \beta_{j,l} \right], \tag{4.10}$$

where the coefficients α and β are dependent only on the positions of samples in the spectral and Fourier domain. This allows for the computation of χ^2, defined to be the mean squared

difference between the modelled data, s_m, and measured samples, s_d, using

$$\chi^2 = \frac{1}{N_{samples}} \sum_{l=1}^{N_{samples}} \left[s_{m_l} - s_{d_l} \right]^2 . \tag{4.11}$$

The reconstructed spectrum is then given by

$$p_m^* = \underset{p_{m_j} \forall\ j}{\operatorname{argmin}}\ \chi^2. \tag{4.12}$$

Reconstruction of spatial brighness distribution using FROID

A 2D version of FROID was developed in [17] and used to reconstruct images of spatial brightness distributions of astronomical objects from squared visibilities,

$$S^{(2)}(u_i, v_i) = \Re[S(u_i, v_i)]^2 + \Im[S(u_i, v_i)]^2, \tag{4.13}$$

and complex closure amplitudes,

$$S^{(3)}(u_j, v_j) = S(u_{1_j}, v_{1_j}) S(u_{2_j}, v_{2_j}) S(u_{3_j}, v_{3_j}), \tag{4.14}$$

where (u_{1_j}, v_{1_j}), (u_{2_j}, v_{2_j}), and (u_{3_j}, v_{3_j}) are locations of three different points in the u-v plane. The observations are functions of Fourier data belonging to the u-v plane.

The closed form solution of the integral Fourier transform of a bilinearly interpolated surface in the image domain shall be re-derived in this section, which provides the model for the 2D FROID algoirthm. Assuming that the measured signal is centred properly and the image contains only real values, let a set of uniform points, $p_{m_{j,k}}$ for $j = 1, \ldots, N_d$ and $k = 1, \ldots, M_d$, separated by Δx and Δy be the initial guess of the model image. That is,

$$p_{m_{j,k}} = \mathcal{I}_m(x_j, y_k) \qquad \text{for } j = 1, \ldots, N_d \text{ and } k = 1, \ldots, M_d$$

where

$$x_{j+1} = x_j + \Delta x$$

and

$$y_{k+1} = y_k + \Delta y.$$

A continuous mapping of the 2D image is formed using bilinear interpolation,

$$\mathcal{I}_m(x, y) = \sum_{j=1}^{N_d - 1} \sum_{k=1}^{M_d - 1} \left[\mathrm{T}_{j,k}^{(1)} + \mathrm{T}_{j,k}^{(2)} + \mathrm{T}_{j,k}^{(3)} + \mathrm{T}_{j,k}^{(4)} \right], \tag{4.15}$$

where

$$\mathrm{T}_{j,k}^{(1)} = p_{m_{j,k}} \left[1 - \frac{x - x_j}{\Delta x}\right] \left[1 - \frac{y - y_k}{\Delta y}\right], \tag{4.16}$$

$$\mathrm{T}_{j,k}^{(2)} = p_{m_{j+1,k}} \left[\frac{x - x_j}{\Delta x}\right] \left[1 - \frac{y - y_k}{\Delta y}\right], \tag{4.17}$$

$$\mathrm{T}_{j,k}^{(3)} = p_{m_{j,k+1}} \left[1 - \frac{x - x_j}{\Delta x}\right] \left[\frac{y - y_k}{\Delta y}\right], \tag{4.18}$$

and

$$\mathrm{T}_{j,k}^{(4)} = p_{m_{j+1,k+1}} \left[\frac{x - x_j}{\Delta x}\right] \left[\frac{y - y_k}{\Delta y}\right]. \tag{4.19}$$

Since the Fourier transform is a linear operation, the evaluation of $S_m(u, v)$ can then be broken into the following four terms,

$$\begin{aligned} S_m(u,v) = &\sum_{j=1}^{N_d-1} \sum_{k=1}^{M_d-1} \int_{x_j}^{x_{j+1}} \int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(1)} e^{-i2\pi(ux+vy)} \, \mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1} \sum_{k=1}^{M_d-1} \int_{x_j}^{x_{j+1}} \int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(2)} e^{-i2\pi(ux+vy)} \, \mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1} \sum_{k=1}^{M_d-1} \int_{x_j}^{x_{j+1}} \int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(3)} e^{-i2\pi(ux+vy)} \, \mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1} \sum_{k=1}^{M_d-1} \int_{x_j}^{x_{j+1}} \int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(4)} e^{-i2\pi(ux+vy)} \, \mathrm{d}x\mathrm{d}y. \end{aligned} \tag{4.20}$$

The x and y factors are separable in each of the Fourier transform terms, which considerably simplifies the integration. The integrals are analytically solved in Appendix A. Just as in 1D FROID, $S_m(u, v)$ can be expressed as a linear combination of the points in the model image once the integration has been performed. So,

$$S_m(u_l, v_l) = \sum_{j=2}^{N_d-1} \sum_{k=2}^{M_d-1} A_{j,k,l} p_{m_{j,k}} + i \sum_{j=2}^{N_d-1} \sum_{k=2}^{M_d-1} B_{j,k,l} p_{m_{j,k}} \tag{4.21}$$

and the coefficients, $A_{j,k,l}$ and $B_{j,k,l}$, measure the amount of influence that a point located at (x_j,y_k) has on the real and imaginary values of point (u_l,v_l) in the u-v plane. They are independent of pixel values, and can be calculated once for a constant set of sampling positions in the Fourier and image domain. $A_{j,k,l}$ and $B_{j,k,l}$ for all j, k, and l are calculated using equations (A.15) to (A.26) in Appendix A. The signal at a point in the u-v plane

is comprised of all the intensities in the image weighted by $A_{j,k,l}$ and $B_{j,k,l}$. In addition, points lying on the border of the image are equated to 0 to create a smooth transition from the image to the surrounding space, which is assumed to have zero intensity. Hence, the indices in the summation of equation (4.21) start at 2 (instead of 1).

Given $S_m(u,v)$, squared visibilities and complex closure amplitudes are modelled using equations (4.13) and (4.14), and compared to the actual observations. χ^2 is calculated as

$$\begin{aligned}\chi^2 = &\sum_{i=1}^{N_{S^{(2)}}} \left[\frac{S^{(2)}(u_i,v_i) - S_m^{(2)}(u_i,v_i)}{\sigma_i}\right]^2 \\ &+ \sum_{j=1}^{N_{S^{(3)}}} \left[\frac{S^{(3)}(u_j,v_j) - S_m^{(3)}(u_j,v_j)}{\sigma_j}\right]^2 \\ &+ \frac{S(u_0,v_0) - S_m(u_0,v_0)^2}{\sigma_{(0,0)}},\end{aligned} \tag{4.22}$$

where $N_{S^{(2)}}$ is the number of measured square visibilities, $N_{S^{(3)}}$ the number of measured complex closure amplitudes, and σ a measure of uncertainty associated with each observation.[17] The last term in equation (4.22) ensures that the total power of the reconstructed image is consistent with the acquired signal. Recall from Table 2.2 that the signal at the origin of the Fourier domain is equal to the total volume under the surface of the image so that

$$S(0,0) = \iint_{-\infty}^{\infty} \mathcal{I}(x,y)\,\mathrm{d}x\mathrm{d}y, \tag{4.23}$$

which is a measure of power if $\mathcal{I}(x,y) \geq 0 \ \forall \ x,y$. $S_m(0,0)$ is calculated by taking the limits of constants $A_{j,k,l}$ and $B_{j,k,l}$ as $u,v \to 0$ and substituting into equation (A.15), which simplifies to

$$S_m(0,0) = \Delta x \Delta y \sum_{j=2}^{N_d-1} \sum_{k=2}^{M_d-1} p_{m_{j,k}}. \tag{4.24}$$

The spatial brightness distribution is again given by minimizing χ^2.

The main objective of FROID is modelled after data fitting, specifically a least squares fit. FROID accounts for different sampling structures that are not required to be uniform or linear. Unlike CLEAN, which corrects for sampling irregularities after the image has been aliased, methods such as MEM and FROID account for the sampling structure when searching among unaliased images for an optimal reconstruction. An analytic solution for the integral Fourier transform of a bililinearly interpolated image is presented by FROID, which is not offered by any of the previously discussed algorithms. FROID is also applicable to reconstructing images from any measurements that can be modelled using Fourier data.

4.3 Application in MRI

The methodology of FROID discussed in the previous sections can be transferred to reconstructing MR images since the same Fourier relationship exists between the image domain and k-space (i.e., k-space is analogous to the u-v plane). For application in MRI, the appropriate objective function is to be defined and the model in equation (4.21) shall be extended to 2D images with complex values. For the remaining discussion, the Fourier domain shall refer back to k-space.

The objective function and 2D model

A 2D image reconstruction can be performed by successively applying the 1D FROID algorithm along each dimension independently since the 2D Fourier transform can be separated as shown:

$$S(k_x, k_y) = \iint_{-\infty}^{\infty} \mathcal{I}(x, y) e^{-i2\pi(k_x x + k_y y)} \, \mathrm{d}x\mathrm{d}y$$

and

$$S(k_x, k_y) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \mathcal{I}(x, y) e^{-i2\pi k_x x} \, \mathrm{d}x \right] e^{-i2\pi k_y y} \, \mathrm{d}y. \tag{4.25}$$

Alternatively, the 2D FROID algorithm utilizing bilinear interpolation and equation (4.21) can be applied. Constants $A_{j,k,l}$ and $B_{j,k,l}$ are to be calculated in the same manner as before (using equations (A.15) to (A.26)), but with k_x and k_y positions in place of u and v respectively. The assumption that the region surrounding the image contains zero intensity is also valid for MR images since the object of interest is usually contained within the FOV.

It is common to sample points in k-space along the k_x or k_y axis where k_y or k_x equals 0. In such cases, $A_{j,k,l}$ and $B_{j,k,l}$ as previously derived are undefined since the equations require divisions by k_x and k_y. The limits of $A_{j,k,l}$ and $B_{j,k,l}$ as $k_x, k_y \to 0$ are given by [17], which were derived using l'Hôpital's rule, and equation (4.24) can be used to calculate the amplitude of the datapoint modelled at the origin of k-space. For this thesis, the limits as $k_x \to 0$ or $k_y \to 0$ have been derived (also using l'Hôpital's rule) and are presented in equations (A.29) to (A.52).

Since complex Fourier data, $s_{d_l} = S(k_{x_l}, k_{y_l})$ for $l = 1, \ldots, N_{samples}$, are directly acquired in MRI, χ^2 needs only to account for the real and imaginary values of the data modelled by equation (4.21), $s_{m_l} = S_m(k_{x_l}, k_{y_l})$. Using the Euclidean norm squared, the difference between the modelled data and measured samples is

$$\chi^2 = \sum_{l=1}^{N_{samples}} \left[\left(\Re\{s_{m_l}\} - \Re\{s_{d_l}\} \right)^2 + \left(\Im\{s_{m_l}\} - \Im\{s_{d_l}\} \right)^2 \right]. \tag{4.26}$$

Just as in equation (4.22), an additional measure is included to fit the power of the reconstructed image to the amplitude at the origin of k-space. So, the objective function to be minimized in order to reconstruct MR images using FROID is

$$p_m^* = \underset{p_{m_{j,k}} \forall\ j,k}{\operatorname{argmin}} \left\{ \chi^2 + [S_m(0,0) - S(0,0)]^2 \right\} \tag{4.27}$$

Equation (4.27) can be adjusted to include constraints that reflect *a priori* information. That is

$$p_m^* = \underset{p_m}{\operatorname{argmin}} \left\{ \chi^2 + [S_m(0,0) - S(0,0)]^2 : C_1(p_m) = 0, C_2(p_m) \geq 0 \right\} \tag{4.28}$$

where $C_i(p_m)$, $i = 1, 2$, are functions of the argument p_m. For example, the sparsity of an image (refer to equation (3.10)) can be imposed as an inequality constraint

$$\|\mathbf{p}_m\|_1 = \sum_{j=2}^{N_d-1} \sum_{k=2}^{M_d-1} |p_{m_{j,k}}| \leq R_1 \tag{4.29}$$

or just as in CS

$$\|\Psi \mathbf{p}_m\|_1 \leq R_1. \tag{4.30}$$

The smoothness of an image can also be incorporated using the entropy metric previously defined for MEM in equation (4.5),

$$-H(\mathcal{I}_m) \leq R_2. \tag{4.31}$$

It is important to note that the FROID algorithm was not developed under the premise that reconstructed images are inherently smooth (i.e., have a large entropy) or sparse. Instead, FROID offers the flexibility to include such *a priori* information when appropriate. It is not limited to one application and can be adjusted as required by incorporating proper constraints in the optimization problem. FROID can account for radial, spiral, and other nonuniform sampling trajectories in MRI since the locations of samples in k-space can be determined by the pulse sequences as discussed in Section 3.3.

Extension to complex images

Inaccuracies in centering the signal in k-space cause images to have amplitudes with imaginary values (recall the Fourier transfrom shift property). In such cases, it may be more appropriate to solve for an image with complex values, and study the magnitude and phase images. FROID can be extended to include complex images while still using the previously

derived coefficients in taking the Fourier transform of a model image. The real and imaginary components of the modelled points, $p_{m_{j,k}}$, can be interpolated separately to form a mapping of the image with complex values as given by

$$\mathcal{I}_m(x,y) = \mathcal{I}_{m_{Re}}(x,y) + i\mathcal{I}_{m_I}(x,y), \tag{4.32}$$

where

$$\mathcal{I}_{m_{Re}}(x,y) = \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1} \Re\left\{\mathrm{T}_{j,k}^{(1)} + \mathrm{T}_{j,k}^{(2)} + \mathrm{T}_{j,k}^{(3)} + \mathrm{T}_{j,k}^{(4)}\right\}, \tag{4.33}$$

$$\mathcal{I}_{m_I}(x,y) = \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1} \Im\left\{\mathrm{T}_{j,k}^{(1)} + \mathrm{T}_{j,k}^{(2)} + \mathrm{T}_{j,k}^{(3)} + \mathrm{T}_{j,k}^{(4)}\right\}, \tag{4.34}$$

and $S_m(k_x,k_y)$ can be evaluated as

$$S_m(k_x,k_y) = \iint_{-\infty}^{\infty} \mathcal{I}_{m_{Re}}(x,y)e^{-i2\pi(k_x x+k_y y)}\,\mathrm{d}x\mathrm{d}y + i\iint_{-\infty}^{\infty} \mathcal{I}_{m_I}(x,y)e^{-i2\pi(k_x x+k_y y)}\,\mathrm{d}x\mathrm{d}y. \tag{4.35}$$

The two terms in the Fourier transform can be solved separately and the integration in each term is solved in the same manner as in the previous section. Therefore, the closed form expression for the modelled Fourier data, s_{m_l}, is

$$s_{m_l} = \sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1}\left[\left(A_{j,k,l}\Re\{p_{m_{j,k}}\} + iB_{j,k,l}\Re\{p_{m_{j,k}}\}\right) + i\left(A_{j,k,l}\Im\{p_{m_{j,k}}\} + iB_{j,k,l}\Im\{p_{m_{j,k}}\}\right)\right]$$

and after collecting real and imaginary terms

$$\begin{aligned} s_{m_l} = &\sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1}\left[A_{j,k,l}\Re\{p_{m_{j,k}}\} - B_{j,k,l}\Im\{p_{m_{j,k}}\}\right] \\ &+ i\sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1}\left[B_{j,k,l}\Re\{p_{m_{j,k}}\} + A_{j,k,l}\Im\{p_{m_{j,k}}\}\right]. \end{aligned} \tag{4.36}$$

The objective function in equation (4.27) is still applicable, but the number of arguments is doubled since p_m^* consists of a real and imaginary part.

Chapter 5

Implementation of FROID

The FROID algorithm is implemented using version 7.0.6 of IDL (Interactive Data Language), which is a high level array-oriented programming language.[21] The 2D version for reconstructing images with only real values from optical interferometer data was implemented in [17]. Images were reconstructed using the unconstrained optimization model, which is outlined in the following steps:

1. Initialize guess, $\mathbf{p_m}^{(0)}$
2. Calculate constants $A_{j,k,l}$ and $B_{j,k,l}$ $\forall$ j, k, l
3. Calculate model Fourier data, $\mathbf{s_m}^{(n)}$
4. Calculate χ^2
5. Determine a step in the argument, $\mathbf{d}^{(n)}$, to take towards p_m^*
6. Adjust $\mathbf{p_m}^{(n)}$ according to $\mathbf{d}^{(n)}$: $\mathbf{p_m}^{(n+1)} = \mathbf{p_m}^{(n)} + \mathbf{d}^{(n)}$
7. Go back to step 3 unless $\|\mathbf{p_m}^{(n+1)} - \mathbf{p_m}^{(n)}\|_2 < \epsilon_1$ or $\chi^2 < \epsilon_2$

For this thesis, the implementation is modified as according to Section 4.3 to suit the reconstruction of MR images. The minimization of the objective function, defined for the application of FROID to MRI, shall be discussed in this chapter. The Levenberg-Marquardt method, which is also employed in [17], shall be used in solving the least squares problem.

Constants $A_{j,k,l}$ and $B_{j,k,l}$ need to be calculated for every combination of (k_x,k_y) and (x,y) positions corresponding to points modelled in the image and sampled in k-space. Therefore, there are two sets of $N_{samples} \times N_d M_d$ coefficients to be calculated using equations derived in Appendix A.

Equation (4.21), which models the Fourier data using a real-valued image, can also be expressed in matrix form. Let all the model points in the image be in vector $\mathbf{p_m}$ and the corresponding coefficents be in matrix $\mathbf{A}$ and $\mathbf{B}$ then

$$\mathbf{s_m} = \mathbf{A}\mathbf{p_m} + i\mathbf{B}\mathbf{p_m}. \tag{5.1}$$

$\mathbf{A}$ and $\mathbf{B}$ are transformation matrices relating points in the image to k-space and each element was derived by integrating the Fourier coefficients with interpolation factors, thus

$$\mathbf{A}_{N_{samples}\times N_d M_d} = \begin{bmatrix} A_{1,1,1} & \cdots & A_{1,M_d,1} & A_{2,1,1} & \cdots & A_{N_d,M_d,1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ A_{1,1,N_{samples}} & \cdots & A_{1,M_d,N_{samples}} & A_{2,1,N_{samples}} & \cdots & A_{N_d,M_d,N_{samples}} \end{bmatrix} \tag{5.2}$$

and $\mathbf{B}$ is similarly arranged. If images with complex values were to be reconstructed, the variables in equation (5.1) can be modified to include the corresponding imaginary components. That is,

$$\mathbf{s_m} = \mathbf{A_c p_{m_c}} + i\mathbf{B_c p_{m_c}} \tag{5.3}$$

where

$$\mathbf{p_{m_c}} = \begin{bmatrix} \mathcal{R}[\mathbf{p_m}] \\ \mathcal{I}[\mathbf{p_m}] \end{bmatrix}, \tag{5.4}$$

$$\mathbf{A_c} = \begin{bmatrix} \mathbf{A} & -\mathbf{B} \end{bmatrix}, \tag{5.5}$$

and

$$\mathbf{B_c} = \begin{bmatrix} \mathbf{B} & \mathbf{A} \end{bmatrix}. \tag{5.6}$$

In programs such as IDL, it is more efficient to perform iterative operations simultaneously using arrays than using for-loops and scalar operations.[21]

5.1 Solving the unconstrained optimization problem

The matrix form of χ^2 is

$$\chi^2 = \|\mathbf{A p_m} - \mathbf{s_{d_R}}\|_2^2 + \|\mathbf{B p_m} - \mathbf{s_{d_I}}\|_2^2, \tag{5.7}$$

where $\mathbf{s_{d_R}}$ and $\mathbf{s_{d_I}}$ are vectors containing the real and imaginary components of s_{d_l} for $l = 1, \ldots, N_{samples}$. The objective function of the optimization is rewritten as

$$\begin{aligned} f(\mathbf{p_m}) &= \|\mathbf{A p_m} - \mathbf{s_{d_R}}\|_2^2 + \|\mathbf{B p_m} - \mathbf{s_{d_I}}\|_2^2 + (\Delta_{xy}\mathbf{e}^\mathsf{T}\mathbf{p_m} - s_{d_{00}})^2 \\ &= (\mathbf{A p_m} - \mathbf{s_{d_R}})^\mathsf{T}(\mathbf{A p_m} - \mathbf{s_{d_R}}) + (\mathbf{B p_m} - \mathbf{s_{d_I}})^\mathsf{T}(\mathbf{B p_m} - \mathbf{s_{d_I}}) + (\Delta_{xy}\mathbf{e}^\mathsf{T}\mathbf{p_m} - s_{d_{00}})^2, \end{aligned} \tag{5.8}$$

where $\mathbf{e}$ is a vector of ones,

$$\mathbf{e} = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}^\mathsf{T}_{1\times N_d M_d}, \tag{5.9}$$

and $\Delta_{x,y} = \Delta x \Delta y$. A model based approach is used to minimize $f(\mathbf{p_m})$. For unconstrained optimization, it is necessary that

$$\frac{\partial f(\mathbf{p_m})}{\partial p_{m_{j,k}}} = 0 \qquad j = 1, \ldots, N_d \text{ and } k = 1, \ldots, M_d \tag{5.10}$$

for $\mathbf{p_m}$ to be a minimizing point. If $\partial f(\mathbf{p_m}^{(n)})/\partial p_{m_{j,k}} \neq 0$ then a step in the argument, $\mathbf{d}^{(n)}$, that takes $f(\mathbf{p_m}^{(n)})$ to the minimum (i.e., $\mathbf{p_m^*} = \mathbf{p_m}^{(n)} + \mathbf{d}^{(n)}$) is desired. The objective function value at $(\mathbf{p_m}^{(n)} + \mathbf{d}^{(n)})$, $f(\mathbf{p_m}^{(n)} + \mathbf{d}^{(n)})$, can be modelled as a quadratic function by taking the Taylor series expansion and ignoring higher order terms to obtain

$$m_n(\mathbf{d}) = f(\mathbf{p_m}^{(n)} + \mathbf{d}^{(n)}) = f(\mathbf{p_m}^{(n)}) + \mathbf{d}^{(n)^\mathsf{T}} \nabla f(\mathbf{p_m}^{(n)}) + \frac{1}{2} \mathbf{d}^{(n)^\mathsf{T}} \nabla^2 f(\mathbf{p_m}^{(n)}) \mathbf{d}^{(n)} \tag{5.11}$$

where $\nabla f(\mathbf{p_m})_{N_d M_d \times 1}$ and $\nabla^2 f(\mathbf{p_m})_{N_d M_d \times N_d M_d}$ are the gradient and Hessian matrices of $f(\mathbf{p_m})$. Given the expression in equation (5.8), $\nabla f(\mathbf{p_m})$ and $\nabla^2 f(\mathbf{p_m})$ can be computed as

$$\nabla f(\mathbf{p_m}) = 2\mathbf{A}^\mathsf{T}(\mathbf{A}\mathbf{p_m} - \mathbf{s_{d_R}}) + 2\mathbf{B}^\mathsf{T}(\mathbf{B}\mathbf{p_m} - \mathbf{s_{d_I}}) + 2\Delta_{x,y}\mathbf{e}(\Delta_{xy}\mathbf{e}^\mathsf{T}\mathbf{p_m} - s_{d00}) \tag{5.12}$$

and

$$\nabla^2 f(\mathbf{p_m}) = 2\mathbf{A}^\mathsf{T}\mathbf{A} + 2\mathbf{B}^\mathsf{T}\mathbf{B} + 2\Delta_{x,y}^2 \mathbf{e}\mathbf{e}^\mathsf{T}. \tag{5.13}$$

The desired step, denoted $\mathbf{d}^*$, at iteration n is one that minimizes the modelled quadratic function, $m_n(\mathbf{d})$. If $m_n(\mathbf{d})$ is convex quadratic (i.e., $\nabla^2 f(\mathbf{p_m})$ is positive semidefinite) then this minimum occurs for that value of $\mathbf{d}$ for which $\nabla m_n(\mathbf{d}) = 0$. Thus, $\mathbf{d}^*$ is found by solving the following system of linear equations,

$$\nabla^2 f(\mathbf{p_m}^{(n)}) \mathbf{d} = -\nabla f(\mathbf{p_m}^{(n)}). \tag{5.14}$$

This approach is called Newton's method and $\mathbf{d}^*$ is referred to as the Newton step. Assuming that the model, $m_n(\mathbf{d})$, is an accurate representation of the objective function and $\mathbf{p_m}^{(n)}$ is close to the solution, Newton's method converges at a quadratic rate.[22] However, $\nabla^2 f(\mathbf{p_m})$ may not be positive semidefinite and $m_n(\mathbf{d})$ may only be sufficiently accurate within a local region. In this case, modifications to the optimization approach are required.

5.2 The Levenberg-Marquardt method

If a trust region around $\mathbf{p_m}^{(n)}$ can be defined in which $m_n(\mathbf{d})$ is considered to be a reasonably accurate model of $f(\mathbf{p_m})$ then the following constrained optimization problem can be solved to find a step towards the minimum $\mathbf{p_m}^*$ at iteration n.

$$\min_{\mathbf{d}} \; m_n(\mathbf{d}) \quad s.t. \; \|\mathbf{d}\| \leq \Delta_r \tag{5.15}$$

where Δ_r is the radius of a hypersphere bordering the trust region. The Lagrange function corresponding to this constrained subproblem is

$$\mathcal{L}(\mathbf{d}, \lambda) = m_n(\mathbf{d}) - \lambda(\Delta_r - \|\mathbf{d}\|) \tag{5.16}$$

For $\mathbf{d}^*$ to be a solution of equation (5.15), the following Karush-Kuhn-Tucker (KKT) conditions must be satisfied at $(\mathbf{d}^*, \lambda^*)$:

$$\begin{aligned} \nabla_{\mathbf{d}}\mathcal{L}(\mathbf{d}^*, \lambda^*) &= \nabla f(\mathbf{p_m}^{(n)}) + \nabla^2 f(\mathbf{p_m}^{(n)})\mathbf{d}^* + \lambda^*\mathbf{d}^* = 0 \\ &\Rightarrow (\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \mathrm{I})\mathbf{d}^* = -\nabla f(\mathbf{p_m}^{(n)}), \qquad (5.17) \\ \lambda^*(\Delta_r - \|\mathbf{d}^*\|) &= 0, \qquad (5.18) \\ \Delta_r - \|\mathbf{d}^*\| &\geq 0, \qquad (5.19) \\ \lambda^* &\geq 0. \qquad (5.20) \end{aligned}$$

Using trust regions to solve least squares problems is the basis of the Levenberg-Marquardt method.[22] Either $\lambda^* > 0$ and a step satisfying equation (5.17) to the boundary of the trust region is taken ($\|\mathbf{d}^*\| = \Delta_r$), or $\lambda^* = 0$ and $\mathbf{d}^*$ is the Newton step to be taken within the trust region. One strategy for choosing the appropriate trust region at each iteration is to adjust λ^* until a step that leads to a lower objective function value is found. This strategy starts by setting λ_0 to some positive value and defining an adjustment factor, $\upsilon > 0$. A trial step is initally solved from the system of equations in (5.17) with $\lambda^* = \lambda_0$. If $\mathbf{d}^*$ leads to a decrease in the objective function, that is,

$$f(\mathbf{p_m}^{(n)} + \mathbf{d}^*) < f(\mathbf{p_m}^{(n)}), \tag{5.21}$$

then the step is accepted and the following updates are applied:

$$\mathbf{d}^0 = \mathbf{d}^*$$

and

$$\mathbf{p_m}^1 = \mathbf{p_m}^0 + \mathbf{d}^0.$$

Otherwise, λ_0 is increased by a factor of υ until, for some $\kappa > 0$, $\lambda^* = \lambda_0 \upsilon^\kappa$ corresponds to a successful step (i.e., satisfies (5.21)). Once λ^*, which leads to a successful step, is found $\lambda_1 = \lambda^*/\upsilon$ is used as the initial trial for the next iteration and the process repeats. With each consecutive success in stepping towards the minimum, λ^* decreases and $\mathbf{d}^*$ converges towards the Newton step. However, when the value of λ^* at iteration n leads to a failed trial, the step size is decreased by increasing λ_n by factors of υ until a new λ^* that results in a successful step is found.

At each iteration, the step, $\mathbf{d}^*$, is in the direction of $-\nabla m_n(\mathbf{d}^*)$ and inversely scaled by λ^* since

$$\lambda^*\mathbf{d}^* = -[\nabla f(\mathbf{p_m}^{(n)}) + \nabla^2 f(\mathbf{p_m}^{(n)})\mathbf{d}^*] = -\nabla m_n(\mathbf{d}^*). \tag{5.22}$$

Thus, λ^* can be considered a damping factor that decreases the values of $\|\mathbf{d}^*\|$ and Δ_r when its value is increased. In finding a step scaled by λ^*, a trust region is directly applied at each iteration.

To account for poorly scaled problems where the magnitudes of the parameters can differ greatly, the trust region can be reshaped to scale appropriately along different dimensions. That is,

$$\|\text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})]\mathbf{d}^*\| \leq \Delta_r \tag{5.23}$$

and

$$(\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})])\mathbf{d}^* = -\nabla f(\mathbf{p_m}^{(n)}) \tag{5.24}$$

where $\text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})]$ is a vector containing the diagonal elements of $\nabla^2 f(\mathbf{p_m}^{(n)})$. Equation (5.24) is now used to find the appropriate step at each iteration. The system of equations in (5.24) is solved for $\mathbf{d}^*$ by decomposing $(\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})])$ into Cholesky factors and using the forward followed by the backward substitution methods.[23]

Referring to equation (5.13), the elements in $\nabla^2 f(\mathbf{p_m})$ are expressed as a summation of inner products for the proposed least squares problem. Therefore, $\nabla^2 f(\mathbf{p_m})$ is guaranteed to be symmetric positive semidefinite and $(\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})])$ is positive definite for $\lambda^* > 0$. Since the Levenberg-Marquardt method only chooses λ^* to be positive, the step accepted at each iteration is a global minimum within the trust region. The Levenberg-Marquardt method behaves like the steepest descent method when λ^* is large and Newton's method when λ^* is small. Therefore, it exhibits local convergence at a rate comparable to Newton's method, but compensates for when the Newton solution is not within the region of trust.

The implementation of FROID is further detailed in the following procedures that include the Levenberg-Marquardt method in determining steps in the argument for minimizing $f(\mathbf{p_m})$.

1. Initialize $\mathbf{p_m}^{(0)}$, λ_0, and υ
2. Calculate $\mathbf{A}$ and $\mathbf{B}$ using equations in Appendix A
3. Calculate $f(\mathbf{p_m}^{(n)})$ from equation (5.8)
4. Calculate $\nabla f(\mathbf{p_m}^{(n)})$ and $\nabla^2 f(\mathbf{p_m}^{(n)})$ from equations (5.12) and (5.13)
5. Set $\lambda^* = \lambda_n$ and solve system of equations in (5.24) using Cholesky factorization
   (a) If $f(\mathbf{p_m}^{(n)} + \mathbf{d}^*) < f(\mathbf{p_m}^{(n)})$ then set $\mathbf{d}^{(n)} = \mathbf{d}^*$ and go to step 6, else proceed
   (b) Set $\lambda^* = \lambda^* \upsilon$
   (c) If $\lambda^* < TOL$ then solve equation (5.24) and go back to 5(a), else FINISH
6. Update $\mathbf{p_m}^{(n+1)} = \mathbf{p_m}^{(n)} + \mathbf{d}^{(n)}$ and $\lambda_{n+1} = \lambda^*/\upsilon$
7. If $\|\mathbf{p_m}^{(n+1)} - \mathbf{p_m}^{(n)}\|_2 > \epsilon_1$ or $\chi^2 > \epsilon_2$ then go back to step 3, else FINISH

To reconstruct images with complex values, replace $\mathbf{p_m}$ with $\mathbf{p_{m_c}}$, $\mathbf{A}$ with $\mathbf{A_c}$, and $\mathbf{B}$ with $\mathbf{B_c}$ in the calculation of $\nabla f(\mathbf{p_m})$ and $\nabla^2 f(\mathbf{p_m})$. The corresponding objective function (equation (5.3)) is minimized using the same optimization routine outlined in procedures 5 to 7.

Chapter 6

Experimental Results and Discussion

FROID has been successful in reconstructing images from optical interferometer data and this work is documented in [15] and [17]. This thesis' primary goal is to apply FROID to reconstructing MR images by adapting and expanding on previously developed methods as discussed in Chapters 4 and 5. The appropriate objective function in the optimization routine is defined and corresponding calculations for reconstructing MR images are implemented. Efforts are made to improve the speed in calculating the constant matrices **A** and **B** required when evaluating model Fourier data. This is done by converting scalar operations to array operations where possible in the IDL code. The results from this thesis and limitations shall be discussed in this chapter.

The objectives of this chapter are to test the performance of FROID, to identify strengths of the algorithm, and to identify areas that require improvement for future development. To accomplish these objectives, FROID is initially restricted to reconstructing real-valued images from simulated data and reconstruction errors with respect to the true images are measured. This restriction is then lifted and an MR image of a bone with complex values is reconstructed. Images are also reconstructed using an FFT and the results are compared to FROID.

6.1 Data simulation

Data are generated using components with analytically derived Fourier transform pairs that are centred at pixel (0,0), which are listed in Table 6.1. The Fourier transform shift property is applied to create off-centred components in the images by multiplying with the appropriate complex exponential. Multiple features are also created by summing the components and their respective signals in the image and Fourier domain. Amplitude data belonging to k-space are created by sampling the Fourier signal. Three different real-valued images, shown in Figure 6.1, are tested and their results shall be discussed in the following

Table 6.1: Fourier Transform Pairs Used for Simulating Data

Description	**Image**	**Fourier Signal**
Point source	$\mathcal{I}(x,y) = A\delta(x,y)$	$S(k_x,k_y) = A$
Rectangle	$\mathcal{I}(x,y) = \begin{cases} A & \text{if } \lvert x\rvert \leq \frac{BW_x}{2} \cap \lvert y\rvert \leq \frac{BW_y}{2} \\ 0 & \text{else} \end{cases}$	$S(k_x,k_y) = \frac{A\sin(BW_x\pi k_x)\sin(BW_y\pi k_y)}{\pi^2 k_x k_y}$
Gaussian	$Ae^{-\frac{x^2}{2\sigma_1^2}-\frac{y^2}{2\sigma_2^2}}$	$2A\sigma_1\sigma_2\pi e^{-2\pi^2(\sigma_1^2k_x^2+\sigma_2^2k_y^2)}$

sections.

The maximum number of points modelled in the image and sampled in k-space is set to 2,500 to fit within memory constraints of IDL. The reason for this is that IDL is only able to allocate up to 1.2 GB of memory to one contiguous block (i.e., one array). This limits the size of a matrix with double precision accuracy, which requires 8 bytes of memory per datum, to approximately $12,250 \times 12,250$ datapoints. Recall from Chapter 5 that the $\mathbf{A}$ and $\mathbf{B}$ matrices are each required to be $N_{samples} \times N_d M_d$ and $\nabla^2 f(\mathbf{p_m})$ is $N_d M_d \times N_d M_d$. Therefore, the image can at most have 110×110 pixels if the $\mathbf{A}$ and $\mathbf{B}$ matrices are to be stored in memory. In addition, there is only a total of 2 to 8 GB of memory addressed to the entire application depending on the operating system and its settings.[24] As a conservative estimate and to avoid memory allocation issues, reconstructed images and data in k-space are modelled using 50×50 samples.

6.2 Testing of uniformly sampled data

Fourier amplitude data sampled at the centres of evenly spaced Cartesian grids are initially tested to ensure that FROID has been implemented correctly; 50×50 samples in k-space are used to reconstruct an image with 50×50 pixels. The three test images are also specifically designed to expose the strengths and weaknesses of FROID, especially when compared to the DFT. Points in the images are modelled to be uniformly spaced, $\Delta x = \Delta y$, and their initial amplitude values are randomly set to avoid bias towards a particular solution (i.e., the initial guess contains no information with respect to the true image). In each test case, FROID is executed with ten different initializations and the reconstruction performance of each trial is measured. Table 6.2 gives the mean and standard deviation of the number of iterations required for FROID to converge and the error measured in the image domain and k-space. The error is defined as the root mean square (RMS) difference between

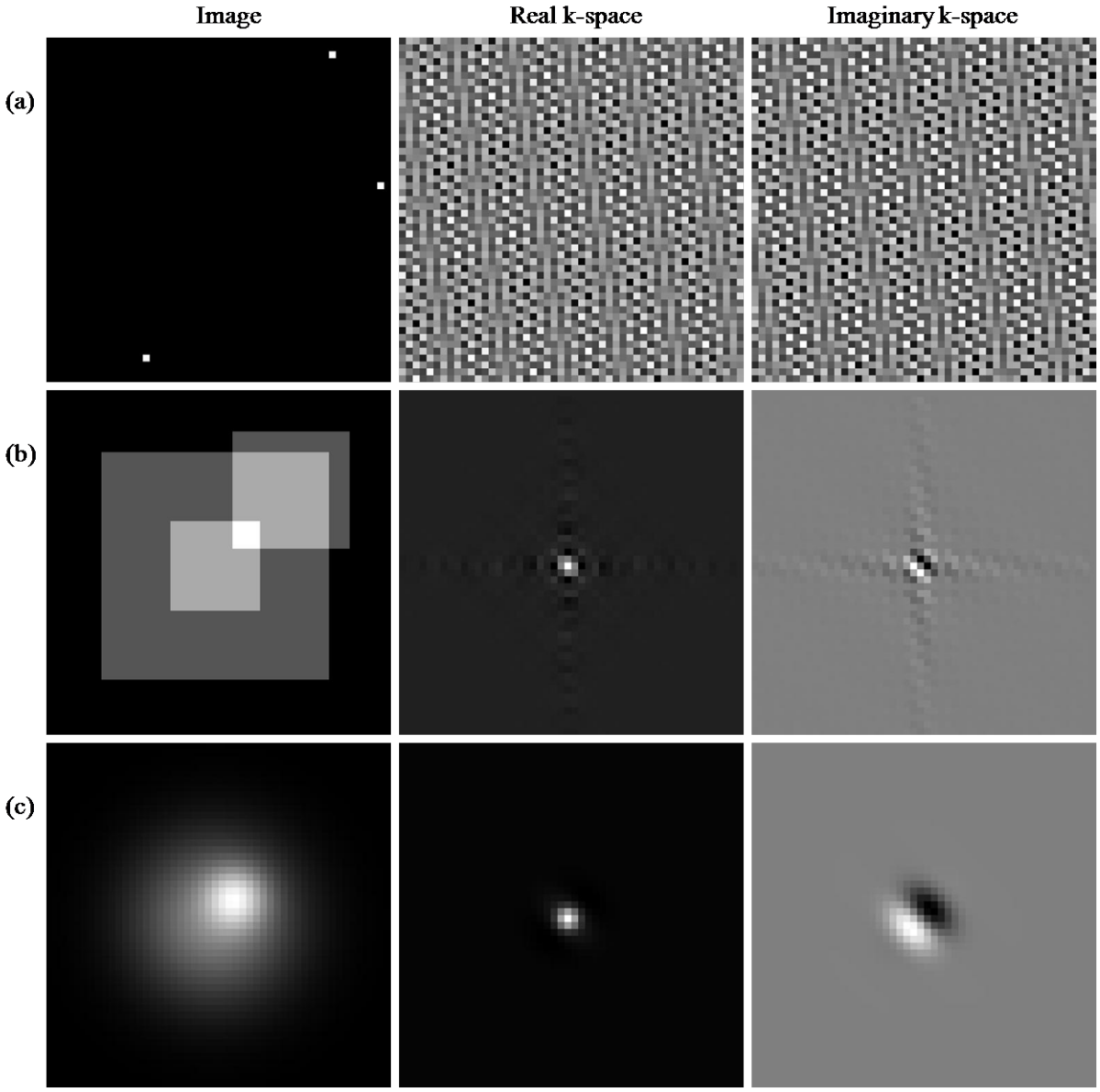


Figure 6.1: Test images: a) 3 point sources, b) 3 rectangles, c) 2 Gaussian functions.

Table 6.2: Reconstruction Performance of FROID with Uniform Samples Measured over Ten Different Random Initializations

Description	**Point Source Image**		**Rectangle Image**		**Gaussian Image**	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
No. of iterations to convergence	11	0.816	11.2	1.81	13	2.05
$f(\mathbf{p}_\mathbf{m}^*)$	95.3	1.28×10^{-13}	3.73×10^{15}	7.92	4.27×10^{14}	12.1
RMS error in k-space (real)	0.138	1.11×10^{-16}	1.05×10^{6}	3.06×10^{-8}	3.32×10^{5}	6.81×10^{-9}
RMS error in k-space (imaginary)	0.138	8.19×10^{-17}	6.30×10^{5}	4.98×10^{-8}	2.43×10^{5}	7.25×10^{-9}
RMS error in image	0.0131	7.54×10^{-13}	0.0740	1.16×10^{-14}	0.00210	5.41×10^{-14}

the modelled and true data. Images are also normalized to have a maximum amplitude of 1 before differences between the intensities of the reconstructed and true images are calculated. It is assumed that there are differences throughout the image and a single RMS value of the differences can be used to convey the magnitude of error observed.

Point source image

The magnitudes of the images reconstructed from data corresponding to point sources using FROID and an FFT are shown in Figure 6.2, which are inverted for better visualization. The FFT reconstructed image contains a significant amount of aliasing despite the fact that the image is relatively simple, and the Fourier signal is uniformly and adequately sampled. The aliasing effects are due to the fact that the exact locations of the point sources happen to fall between pixels where the image is reconstructed by an FFT. The locations of the pixels are determined by the samples and assume that the image is composed of a series of impulses at those locations. Pixels are modelled starting at the origin and at fixed intervals of $1/(\Delta k_x N_d)$ and $1/(\Delta k_y M_d)$ along the x and y directions in the image domain. Therefore, the energy lying between these pixels is not well-modelled and leaks into other parts of the image, which produces aliasing. The intensities in the image are redistributed so that Parseval's theorem is satisfied. The aliasing significantly degrades the quality of the image and the RMS difference between the intensities of the FFT reconstructed image and the true image is 0.078.

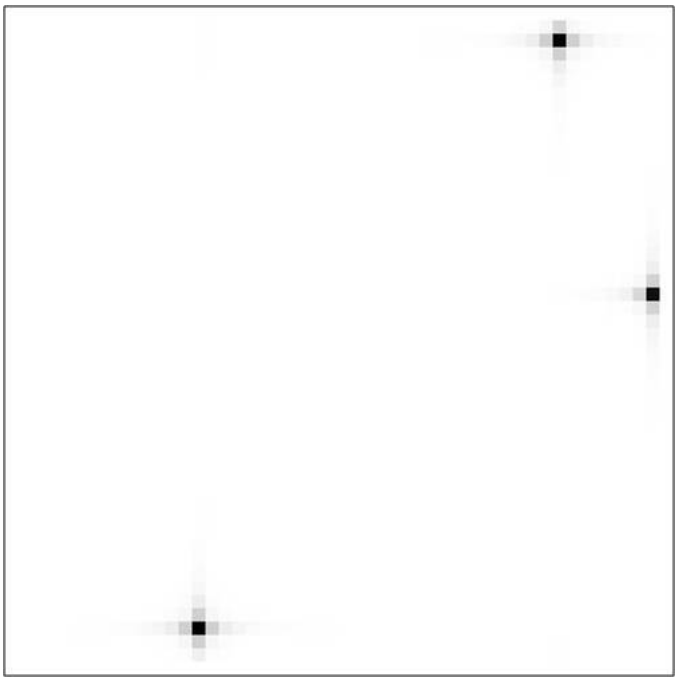
FROID allows arbitrary sampling of the Fourier signal, which is interpolated from amplitudes modelled at independent locations in the image domain. Pixels are strategically

placed in the image to coincide with the locations of the point sources and are not constrained by the locations of the measured samples. This flexibility can be advantageous if locations of important features are estimated ahead of time and are accommodated during image reconstruction using FROID. Even if such information is not available, different images can be produced by modelling pixels at different locations, and through trial and error a reasonable representation of the true image may be found.

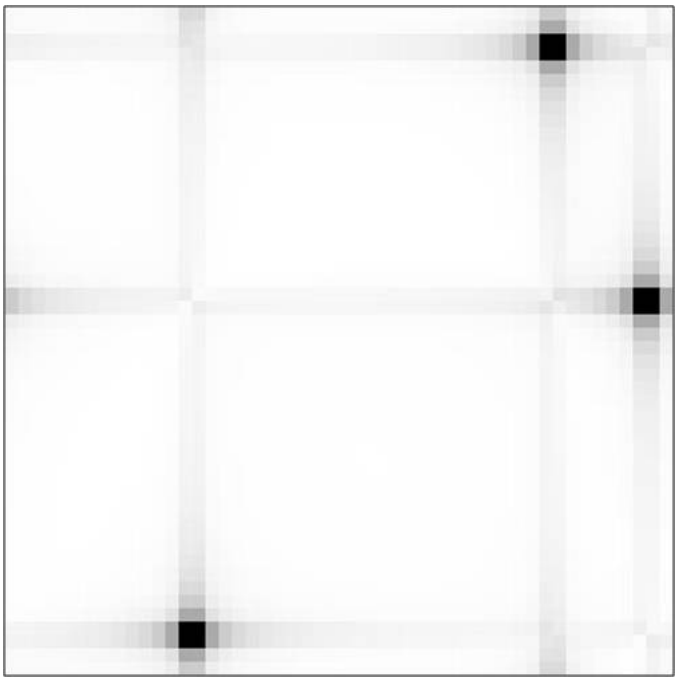
The image reconstructed by FROID does contain some aliasing resulting in an RMS difference of 0.013 when compared to the true image, which is less than that of an FFT. FROID assumes a continuous surface of the image, which is modelled using bilinear interpolation. If the resolution of the modelled points is not fine enough, then intensities interpolated between pixels that are inaccurate would factor into the model Fourier data. When attempting to find a least squares fit to the measured samples, intensities at other locations of the image are introduced in order to cancel out the signal from unwanted components that are mistakenly interpolated. In Figure 6.3, the modelled data does not coincide with the measured samples in k-space and a fair amount of separation between the two datasets are observed when overplotted. The RMS difference is approximately 0.14 for both the real and imaginary data as shown in the first column of Table 6.2, which is more than 10% error when compared to the RMS amplitude of the signal in k-space (1.22 for both real and imaginary).

Rectangle image

Images with discontinuous surfaces such as rectangles experience the Gibbs phenomenon when a limited number of samples in the Fourier domain are used in image reconstruction. It has been discussed before that the Fourier transform of a rectangle is a sinc function with infinite extent. When a limited set of data is sampled in k-space, the sinc function is truncated and ringing in the image occurs with the maximum error at points located near where the image is discontinuous.[6] Figure 6.4 shows the reconstructed magnitude images of three rectangles. Both images reconstructed by FROID and an FFT have crosshatched patterns aliased throughout the image, which are actually ripples caused by Gibbs phenomenon. This results in RMS errors of 0.074 and 0.060 in the images reconstructed by FROID and an FFT respectively. The ripples can be compressed, but not eliminated, by including more Fourier samples in the reconstruction of the image. The crosshatching in the FROID image is more pronounced due to errors induced by interpolation inaccuracies. Just as with the "point source" image, the continuous surface assumed by FROID poorly models sharp edges where the intensities drop to zero. When using bilinear interpolation, a sloped descent from pixels with non-zero intensities to neighbouring pixels with zero intensities is modelled, which is inaccurate for images with discontinous surfaces.

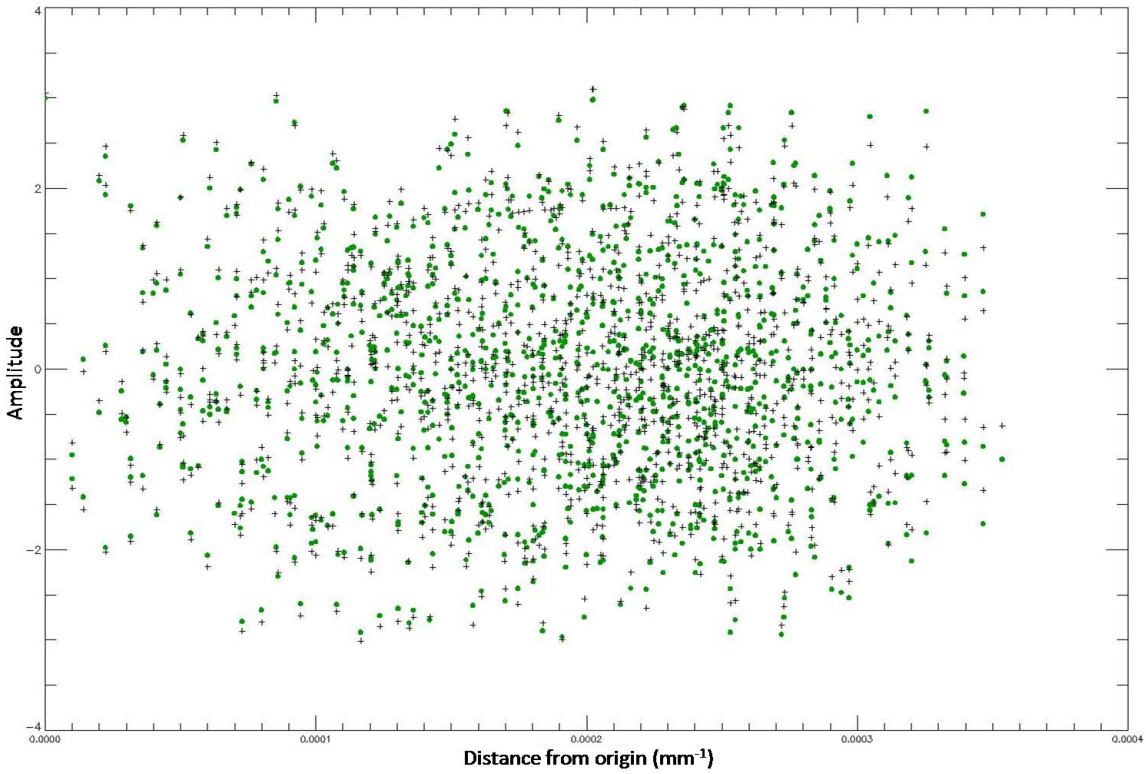


(a) FROID

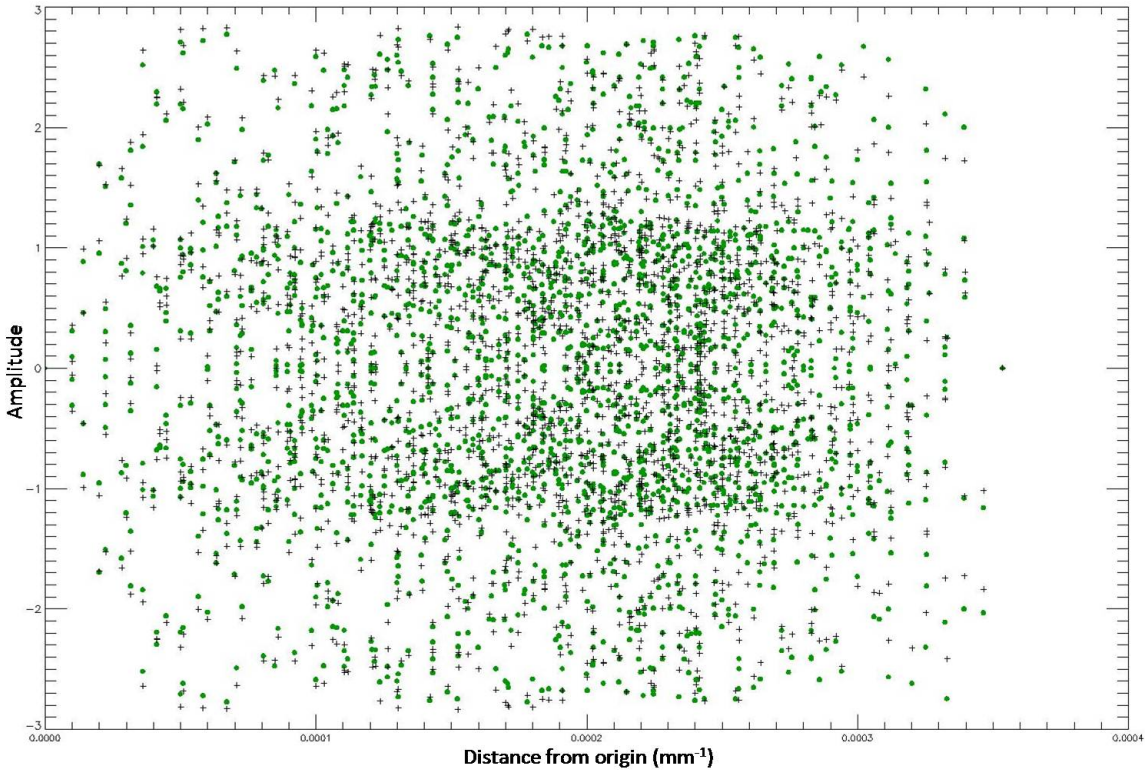


(b) FFT

Figure 6.2: FROID and FFT reconstructed images of 3 point sources.



(a) Real data



(b) Imaginary data

Figure 6.3: Data fit in k-space of the "point source" image: measured data represented by ∘ and modelled data by +.

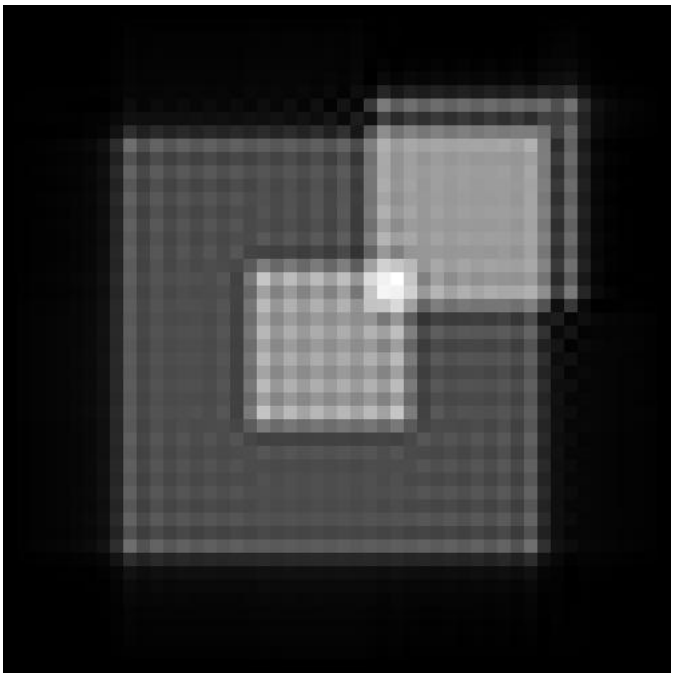
In Figure 6.5, the data modelled by FROID appear to have converged to fit relatively well with the measured samples. Although the RMS errors measured from the real and imaginary amplitudes modelled in k-space (reported in the third column of Table 6.2) seem large, they are only approximately 0.60% and 1.4% of the RMS values of the true real and imaginary signals, which are 1.8×10^8 and 4.5×10^7 respectively. Therefore, the differences between the modelled and measured k-space samples are less significant relative to the magnitude of the k-space signal.

The magnitudes and phases of the modelled and measured data in k-space are also compared in Figure 6.6. Differences between the modelled and measured amplitudes result in the phases of many samples being offset. The error is larger at high spatial frequencies where the magnitudes of the data in k-space are also smaller. Recall from Section 4.3 that an additional term is included in the objective function to fit the power of the reconstructed image to the amplitude at (0,0) in k-space. When determining the step, $\mathbf{d}^{(n)}$, in the argument to take at each iteration to minimize the objective function, the difference between $S_m(0,0)$ and $s_{d_{00}}$ is added to every element in $\nabla f(\mathbf{p_m})$ (refer to equation (5.12)). If the element is much smaller in magnitude, which corresponds to a smaller residual at that point in k-space, the addition causes numerical inaccuracies from rounding errors. As a result, the fit between data with small initial residuals, which are located at high spatial frequencies in this case, is compromised.

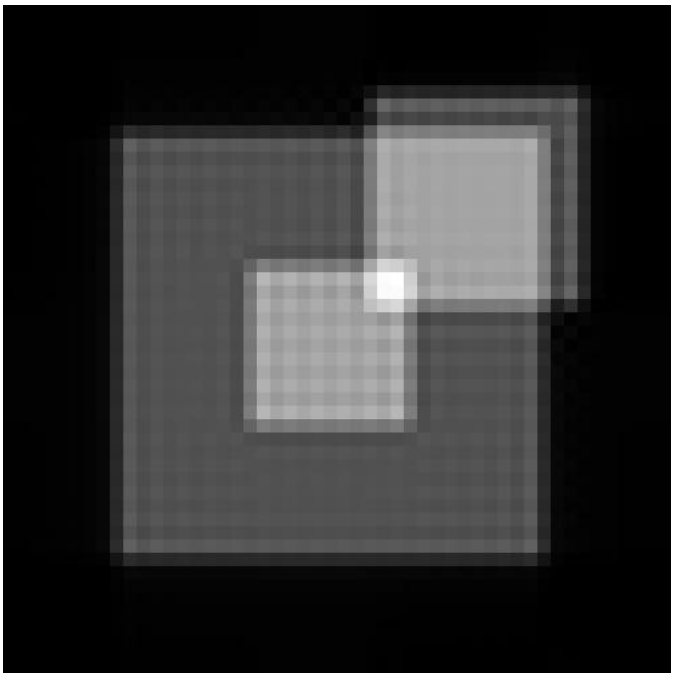
Gaussian image

As shown in Figure 6.7, the images of Gaussian functions reconstructed by FROID and an FFT are very similar. Gaussian functions are smooth and continuous throughout the image domain and approximations made by FROID and an FFT seem to be sufficient in modelling the image. In particular, the intensities between pixels modelled by FROID using bilinear interpolation are valid since the interpolated surface does follow the general trend (i.e., increase or decrease in intensity) of the Gaussian functions. The smoothness of the bilinearly interpolated surface is dependent on the resolution of points modelled in the image.

Taking the difference between the true image and the image reconstructed by FROID reveals that errors occur in two main areas (see Figure 6.8): the edges and the peaks of the Gaussian functions. Recall that in Section 4.3 the image space outside the FOV is assumed to have zero intensity and a zero border is placed around the image. In this case where the Gaussian functions have non-zero values beyond the FOV, FROID inaccurately models the boundaries of the image. The Gaussian functions in the image reconstructed by FROID are also scaled such that the peaks are slightly higher than those of the true image to compensate for the difference in power observed at the edges of the image. However, the

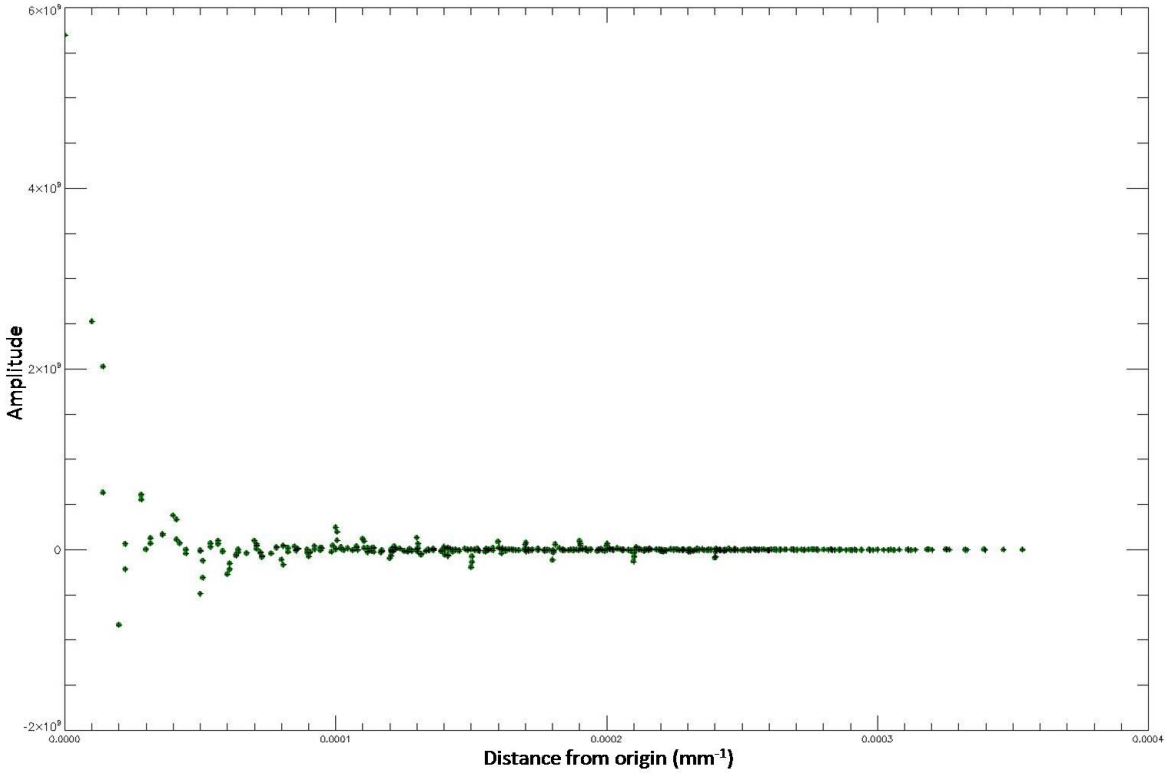


(a) FROID

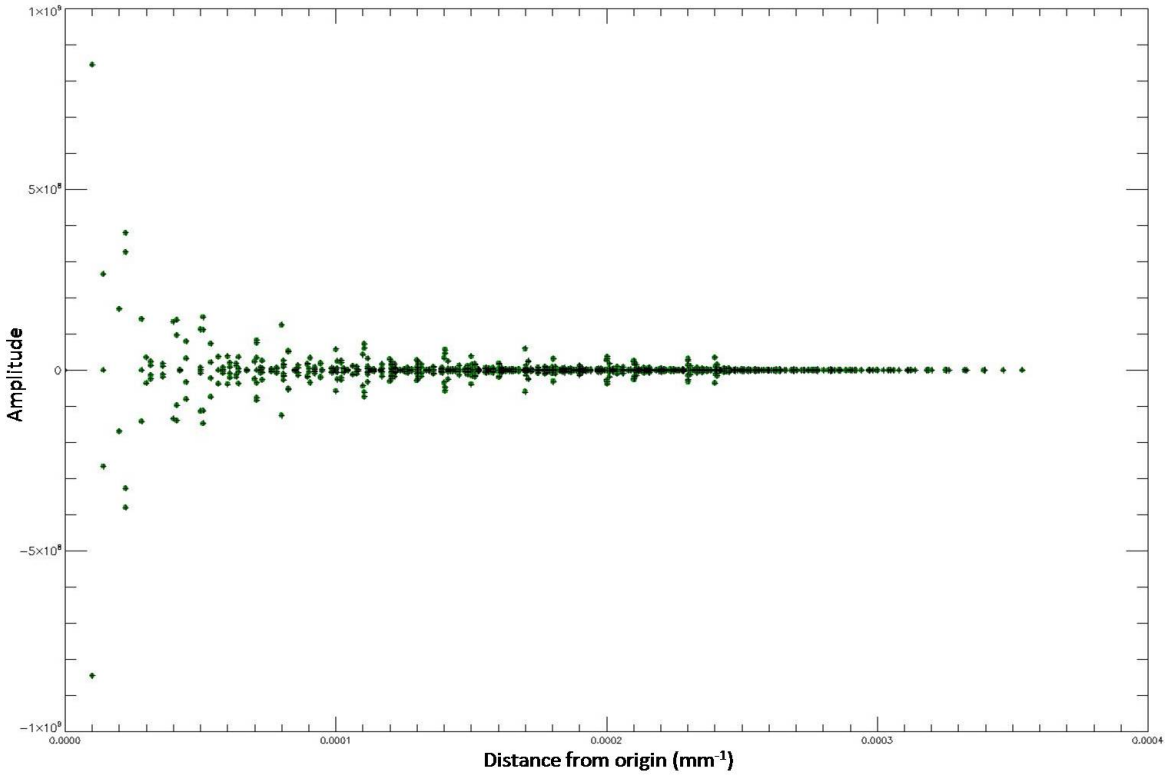


(b) FFT

Figure 6.4: FROID and FFT reconstructed images of 3 superimposed rectangles.

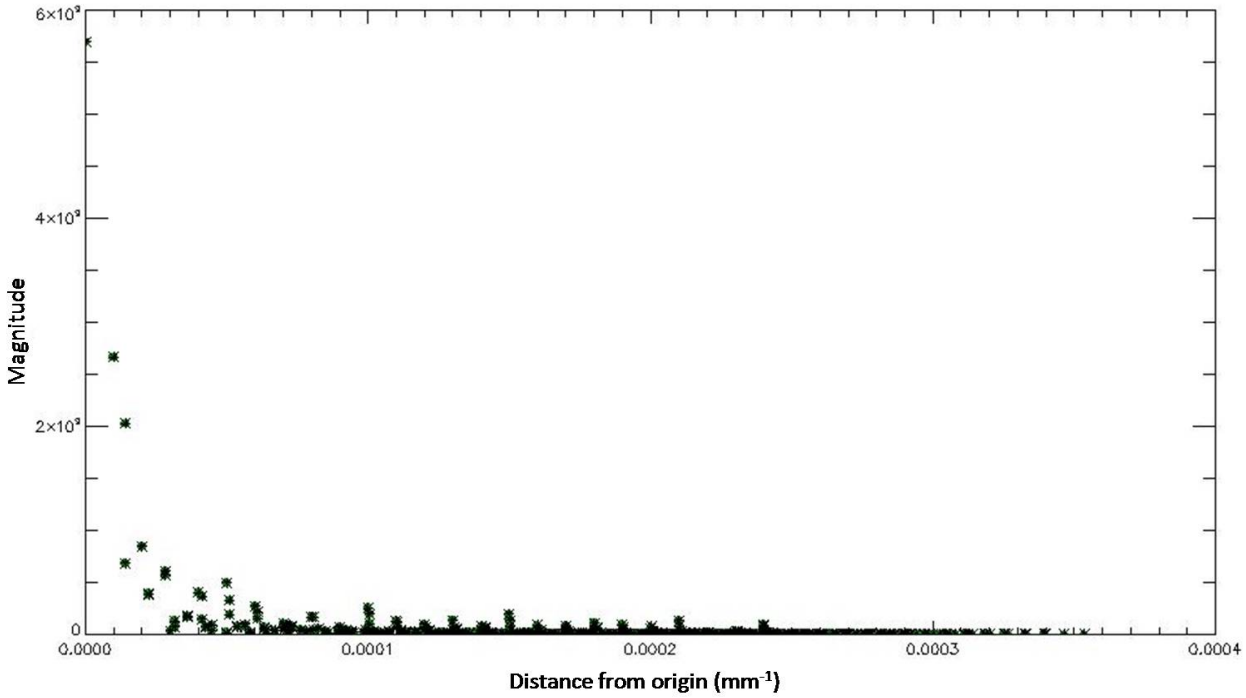


(a) Real data

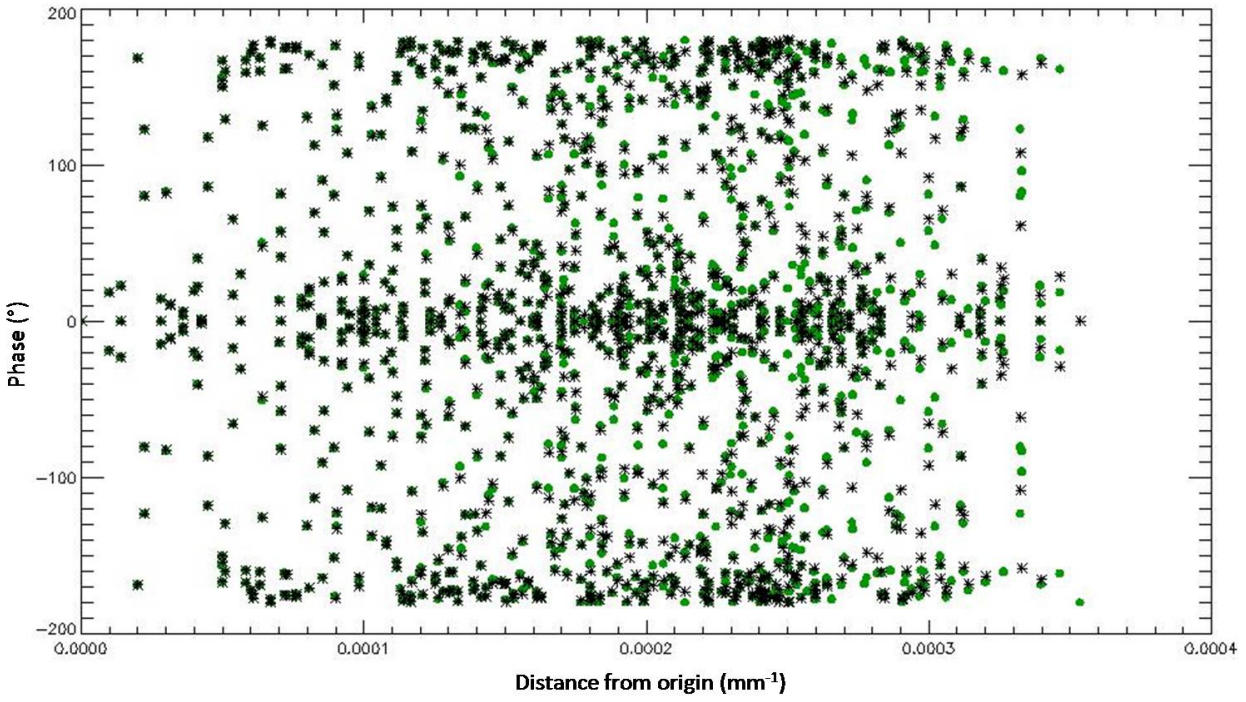


(b) Imaginary data

Figure 6.5: Data fit in k-space of the "rectangle" image: measured data represented by ∘ and modelled data by +.

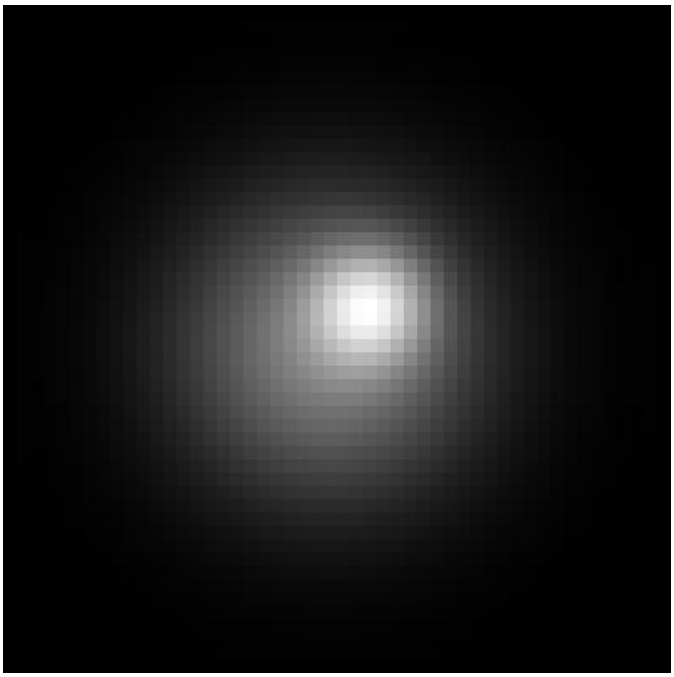


(a) Magnitudes of data

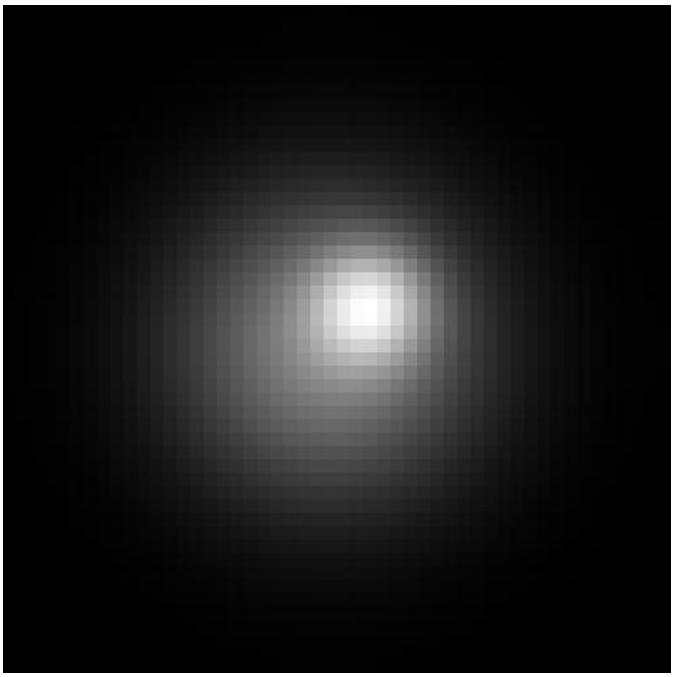


(b) Phases of data

Figure 6.6: Data fit of magnitudes and phases in k-space of "rectangle" image: measured data represented by ∘ and modelled data by +.



(a) FROID



(b) FFT

Figure 6.7: FROID and FFT reconstructed images of 2 superimposed Gaussian functions.

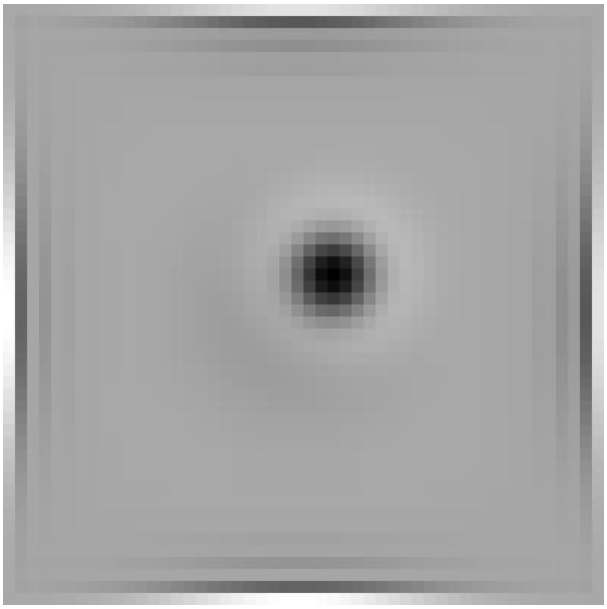


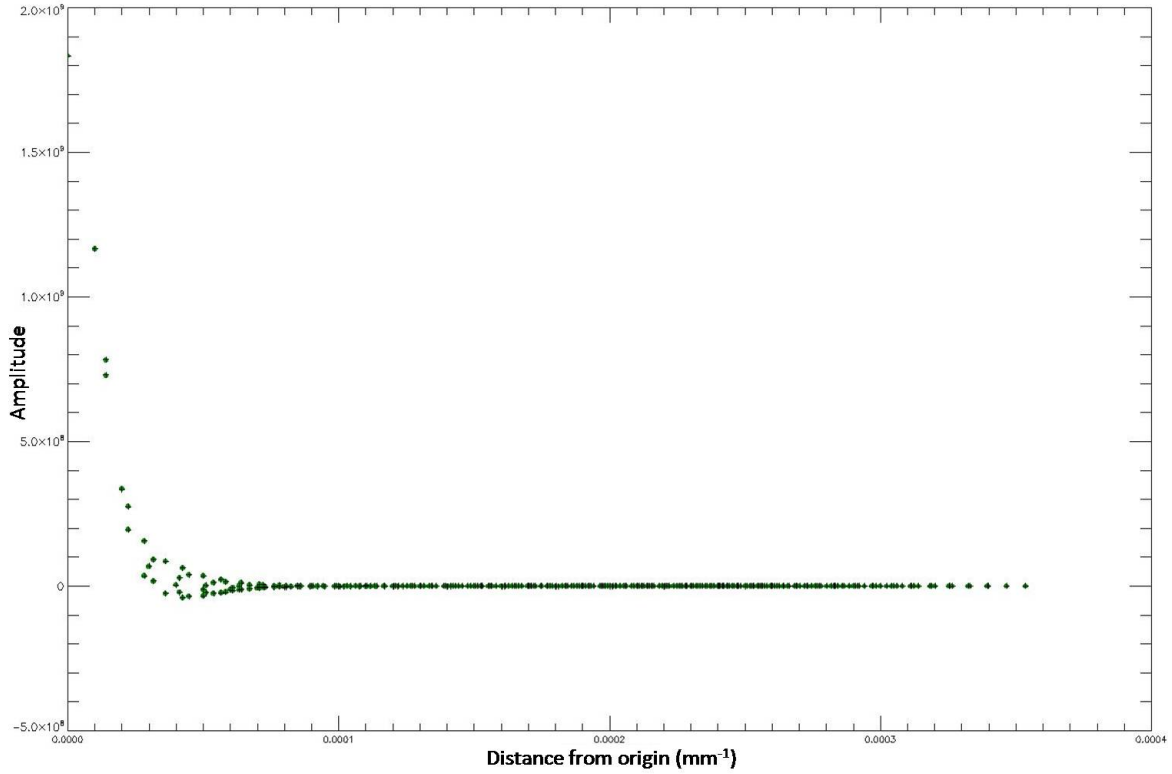
Figure 6.8: Difference between image reconstructed by FROID and true image.

Gaussian functions are sufficiently contained within the FOV that the intensities at the edges are low and the errors incurred (RMS error of 2.1×10^{-3}) are not noticeable in the magnitude image shown in Figure 6.7. The RMS error in the image reconstructed by an FFT (6.9×10^{-4}) is lower since it does not have issues dealing with modelling intensities at the boundaries of the image.

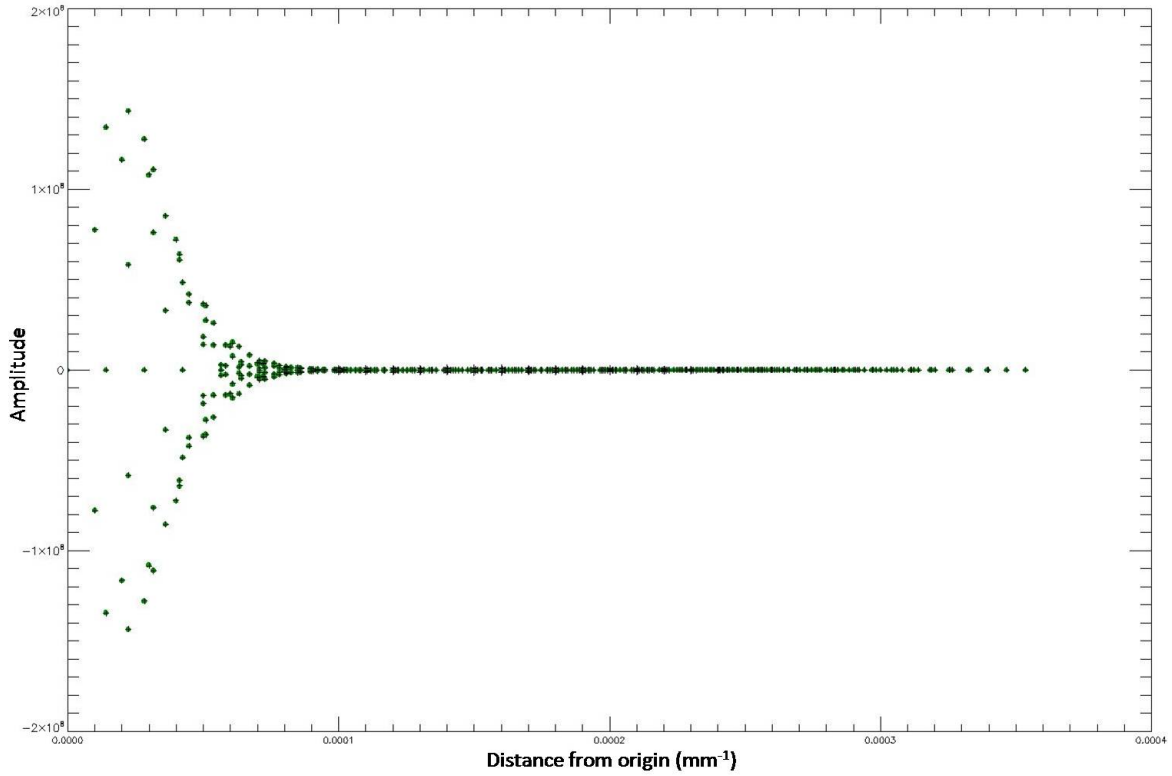
The amplitudes modelled in k-space have converged to fit relatively well with the measured samples as shown in Figure 6.9. Just as discussed with the previous "rectangle" image, the RMS differences observed in the real and imaginary data modelled in k-space are large (refer to the fifth column in Table 6.2), but are only approximately 0.48% and 1.7% of the RMS amplitudes of the real and imaginary signal respectively. Among the three test images reconstructed by FROID, the "Gaussian" image had the lowest RMS error when compared to its respective true image. The "Gaussian" image shall be further tested with nonuniform samples in the next section.

6.3 Testing of randomly sampled data

Amplitudes are randomly sampled from the Fourier signal of the "Gaussian" image and reconstructed using FROID. For a dense dataset (i.e., 2,500 samples), the mean RMS error of the reconstructed image over five different tests with five different randomly sampled datasets is 0.0025. The performance is comparable to the previous analysis when uniform datasets (i.e., evenly sampled data conforming to a Cartesian grid) were used. Figure

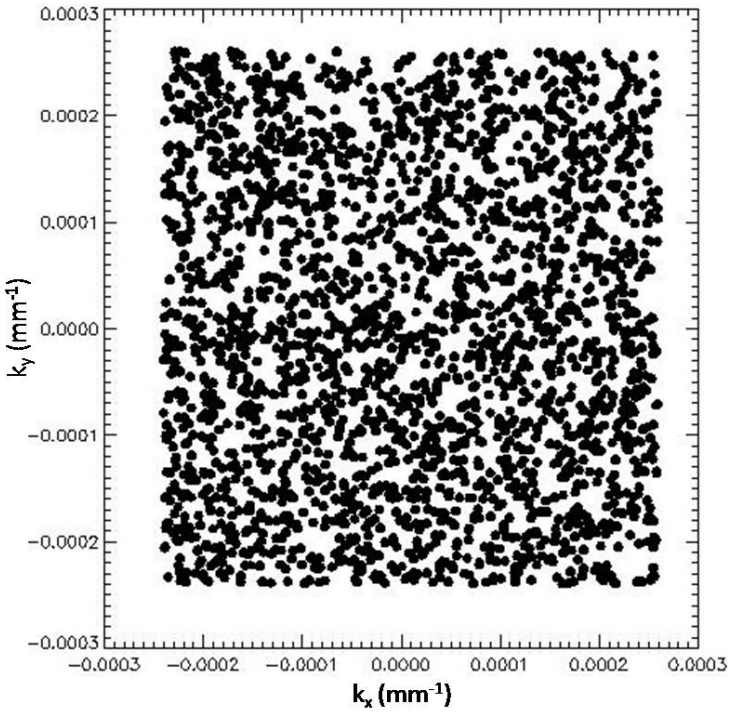


(a) Real data

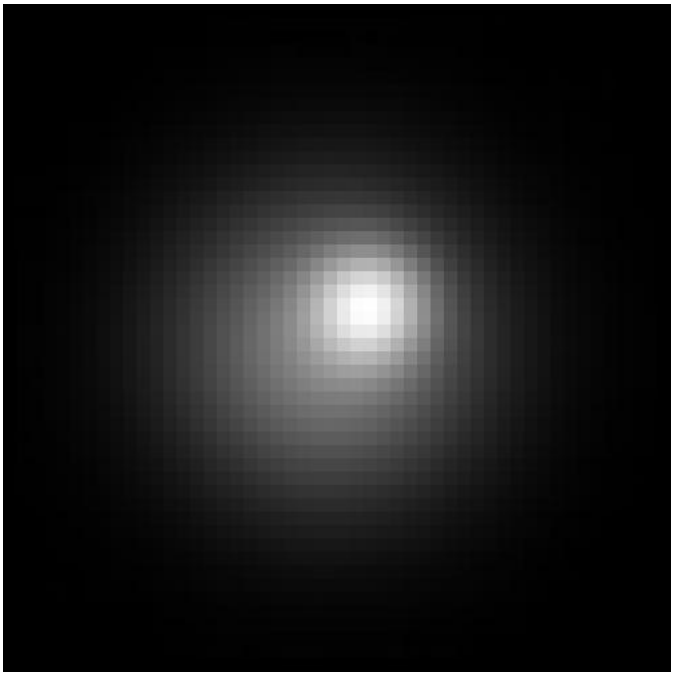


(b) Imaginary data

Figure 6.9: Date fit in k-space of the "Gaussian" image: measured data represented by ∘ and modelled data by +.



(a) Points sampled in k-space



(b) Reconstructed image

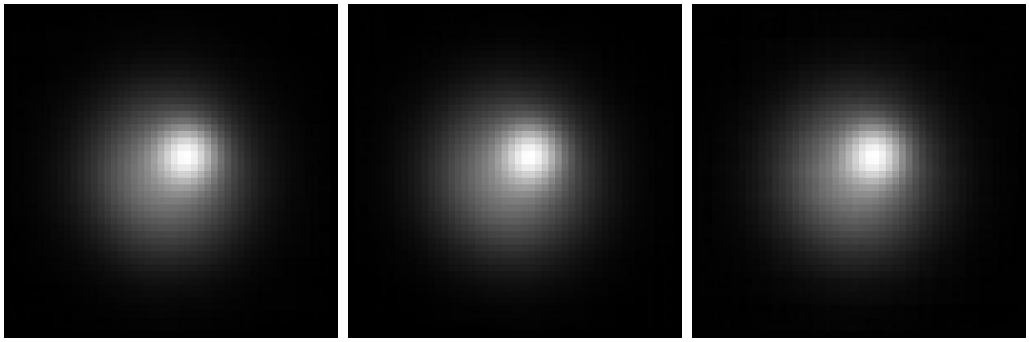
Figure 6.10: Reconstruction of the "Gaussian" image from nonuniform samples using FROID from one of the five tests that were performed.

6.10 shows the reconstructed image and the corresponding locations in k-space that are sampled. The ability to account for nonuniform samples in reconstructing images is an advantage that FROID posseses as demonstrated here.

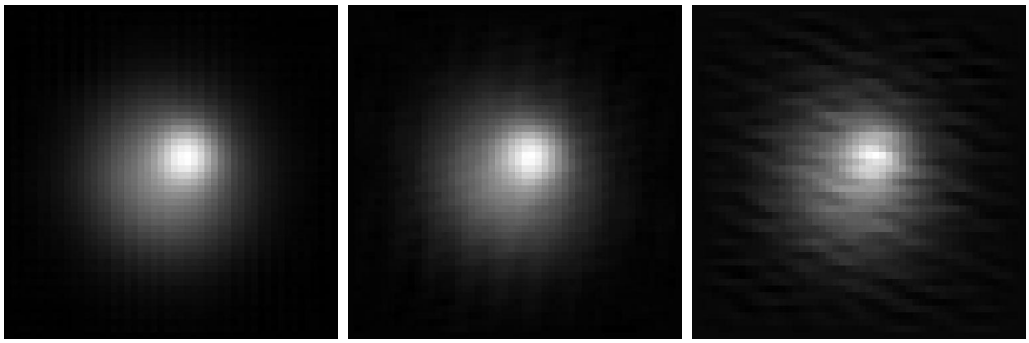
The standard deviation (calculated over the five different tests) of the RMS difference between the modelled and measured amplitudes in k-space increases with random sampling since the data coverage in k-space varies. The amount of data that is sampled at low spatial frequencies where the magnitudes are larger, versus at higher spatial frequencies where the magnitudes are smaller, can significantly change χ^2 and affect the behaviour of the optimization. In some instances, nonuniform sampling results in lower RMS errors measured in the real and imaginary Fourier data. However, accurate modelling of amplitudes in k-space does not necessarily mean better image reconstruction as shown when k-space is undersampled. Table 6.3 shows the results of FROID on reconstructing the "Gaussian" image from nonuniform datasets that were undersampled by different amounts; 100% of data refers to 2,500 samples. As the number of samples measured in k-space decreases, and are less than the number of pixels in the reconstructed image (kept constant at 2,500 pixels), the number of possible solutions that FROID can converge to increases; many of these solutions do not match the true image and are undesirable. When the data is undersampled the system of equations in (5.1) becomes undetermined where the degrees of freedom exceed the number of known measurements and more than one solution exists with corresponding Fourier data that fits. The image can not be uniquely and fully determined when the objective function to be minimized remains unconstrained. Recall from Section 3.2 on compressed sensing that if the image is sparse or a sparsifying transform is found, and k-space can be undersampled such that it gives incoherent measurements, then the image can be reconstructed by finding the sparsest vector. Compressed sensing solves for a unique solution by assuming that the desired image is represented by the sparsest vector among all possible solutions. In this example, no constraints are imposed on FROID to account for redundant information that can be used to assist in reconstructing images from undersampled data. Figure 6.11 shows images reconstructed by FROID from datasets of differing density. After the data coverage in k-space has been reduced to less than 50% of the original 2,500 points, the background and one of the Gaussian functions are no longer distinguishable. When only 250 (10%), 500 (20%), 750 (30%) or 1000 (40%) datapoints are sampled in k-space, χ^2 and the RMS error in k-space are notably low since there are fewer data to fit. So, FROID is successful in minimizing the objective function and finding a least squares fit between the modelled and measured amplitudes in k-space, but the solution to which it converged is not the desired reconstruction. Figure 6.12 shows the trend of the mean and standard deviation of the RMS errors measured in the reconstructed images as data are successively undersampled. In this case, up to 20-30% of data missing due

Table 6.3: Reconstruction Performance of FROID on the "Gaussian" Image with Nonuniform Samples Measured over Five Tests

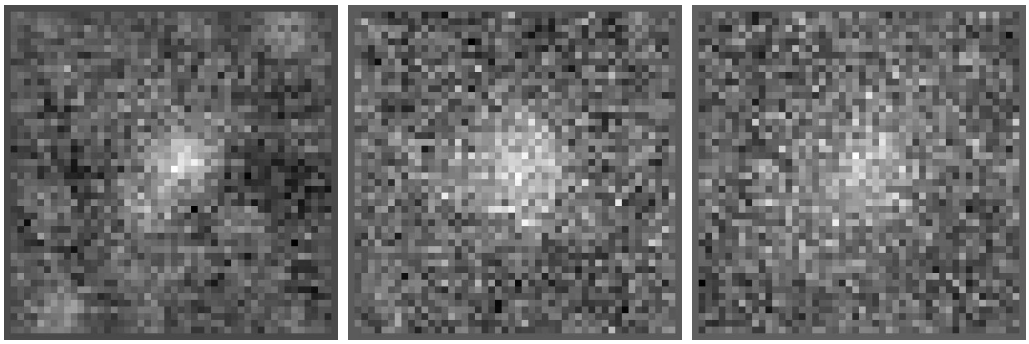
Percentage of Data Sampled		**No. of Iterations to Converge**	$f(\mathbf{p}_\mathbf{m}^*)$	**RMS Error in K-space**		**RMS Error in Image**
				Real	**Imaginary**	
100%	mean	12.8	1.33×10^{14}	1.99×10^{5}	1.06×10^{5}	0.00253
	std. dev.	0.837	3.23×10^{13}	2.12×10^{4}	1.35×10^{4}	2.38×10^{-4}
90%	mean	11.8	1.42×10^{14}	2.16×10^{5}	1.16×10^{5}	0.00280
	std. dev.	2.28	5.19×10^{13}	4.08×10^{4}	1.92×10^{4}	5.68×10^{-4}
80%	mean	13.8	7.24×10^{13}	1.62×10^{5}	8.78×10^{4}	0.00325
	std. dev.	2.39	1.91×10^{13}	1.77×10^{4}	1.20×10^{4}	8.29×10^{-4}
70%	mean	14	6.23×10^{13}	1.59×10^{5}	8.56×10^{4}	0.00513
	std. dev.	1.87	3.63×10^{13}	4.39×10^{4}	2.42×10^{4}	8.24×10^{-4}
60%	mean	16	3.76×10^{13}	1.34×10^{5}	7.97×10^{4}	0.0132
	std. dev.	1.22	6.59×10^{12}	1.39×10^{4}	5.98×10^{3}	0.00395
50%	mean	21	3.81×10^{12}	3.74×10^{4}	2.79×10^{4}	0.0400
	std. dev.	3.32	3.43×10^{12}	2.73×10^{4}	1.78×10^{4}	0.0364
40%	mean	29.4	2.12×10^{-12}	3.17×10^{-8}	2.20×10^{-8}	0.120
	std. dev.	5.41	2.89×10^{-12}	2.61×10^{-8}	6.81×10^{-9}	0.0246
30%	mean	26.6	7.90×10^{-13}	2.23×10^{-8}	2.18×10^{-8}	0.167
	std. dev.	3.97	3.29×10^{-13}	4.94×10^{-9}	4.43×10^{-9}	0.0188
20%	mean	23.6	7.25×10^{-13}	2.82×10^{-8}	2.43×10^{-8}	0.190
	std. dev.	3.65	2.37×10^{-13}	5.16×10^{-9}	4.47×10^{-9}	0.0238
10%	mean	24.8	2.09×10^{-13}	2.01×10^{-8}	2.05×10^{-8}	0.227
	std. dev.	7.19	5.39×10^{-14}	2.90×10^{-9}	3.10×10^{-9}	0.00998



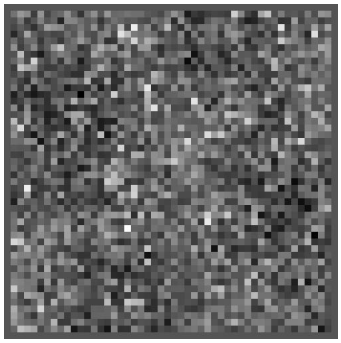
(a) 100% of data sampled (b) 90% of data sampled (c) 80% of data sampled



(d) 70% of data sampled (e) 60% of data sampled (f) 50% of data sampled



(g) 40% of data sampled (h) 30% of data sampled (i) 20% of data sampled



(j) 10% of data sampled

Figure 6.11: "Gaussian" images reconstructed by FROID from nonuniform data.

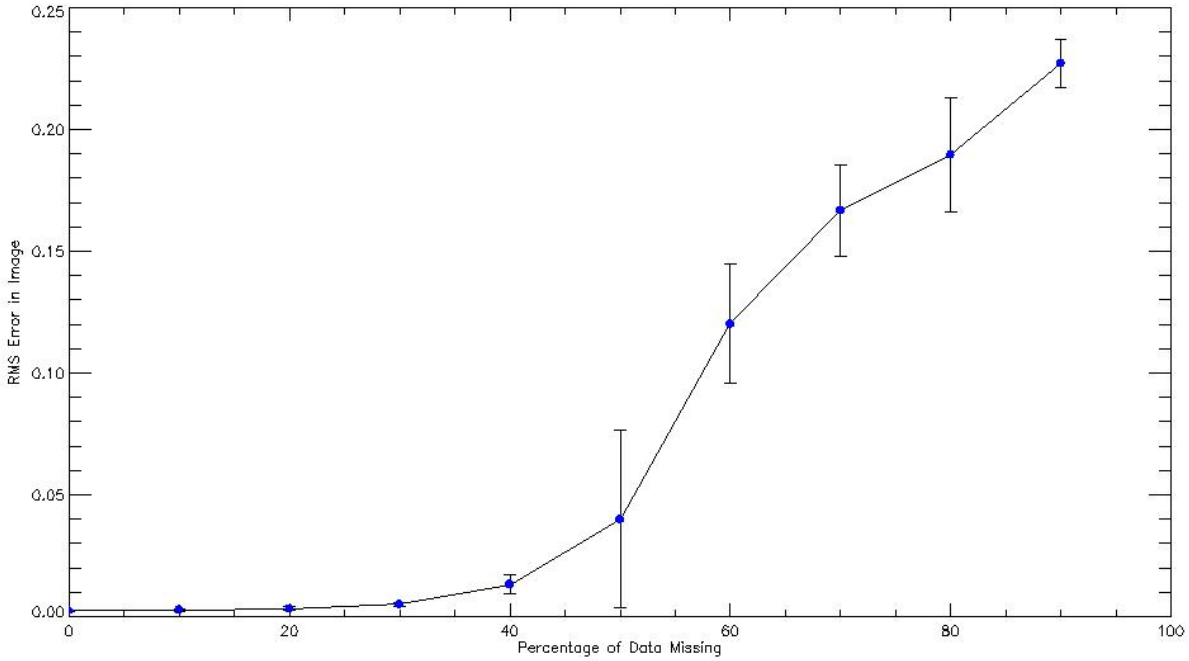


Figure 6.12: Mean RMS error and standard deviation in the "Gaussian" image reconstructed by FROID on sparse and nonuniform data.

to undersampling is tolerable and FROID still reconstructs relatively good images consistently. In addition, the number of iterations required for FROID to converge significantly increases when the number of undetermined degrees of freedom increases.

6.4 Reconstruction of a bone image

A sample of a cow's bone is imaged using a 1.5T MRI scanner at Sunnybrook Hospital (Toronto) and the data corresponding to an image with 256×256 pixels, shown in Figure 6.13, are captured in k-space. The dataset is cropped such that only 2,500 uniform samples of low spatial frequency data (at the centre of k-space) are kept, which allows the IDL implementation of FROID to be used without issues in memory allocation. It is important to note that shortening the length of the signal in k-space is not the same as undersampling, which increases the interval between samples for a fixed bandwidth. Reducing the length of the signal in k-space to 50×50 samples results in lowering the resolution of the image to 50×50 pixels where the high frequency components are thrown away (i.e., the image has been low-pass filtered). Since the peak signal is not necessarily centred properly, an image with complex values is reconstructed using the complex version of FROID so that a magnitude image can be computed. Reconstructing complex images doubles the number of

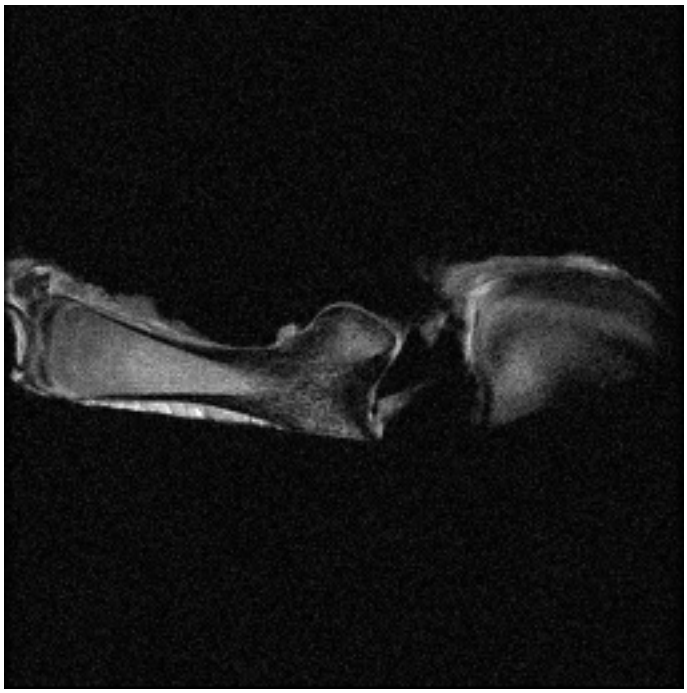


Figure 6.13: 256×256 MR image of bone reconstructed using an FFT.

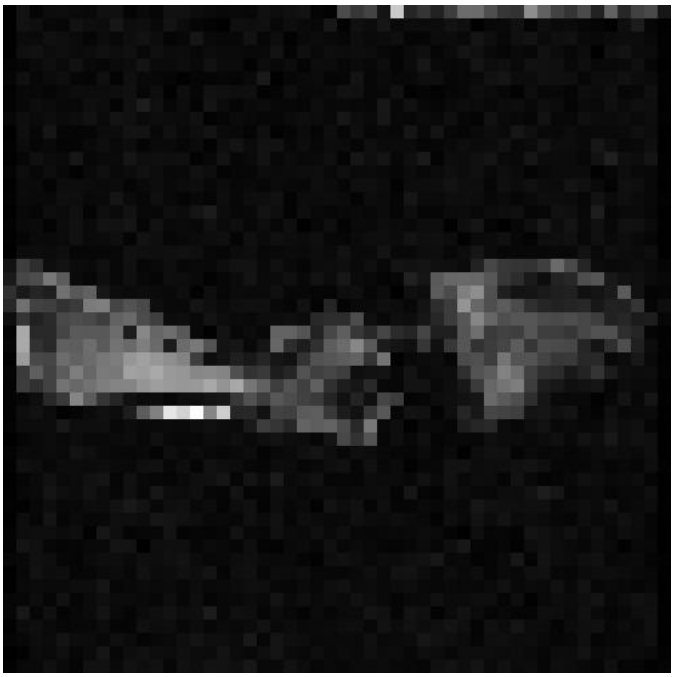
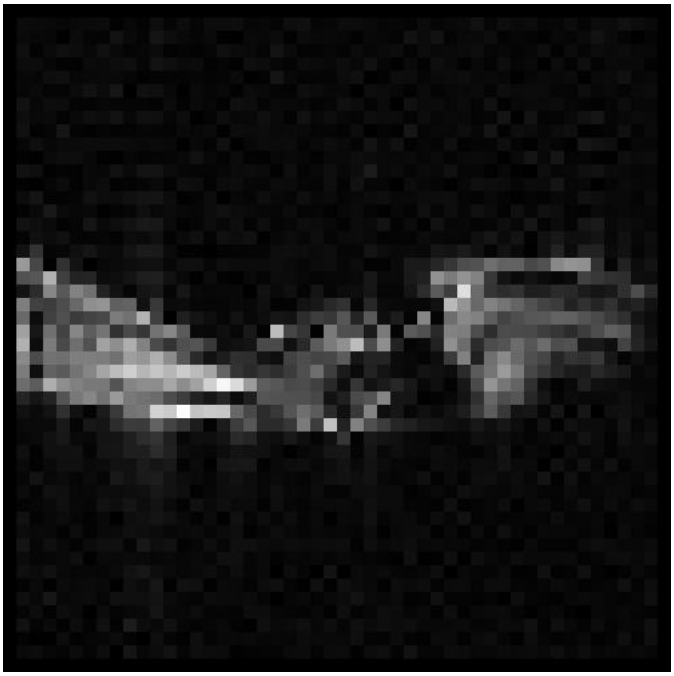
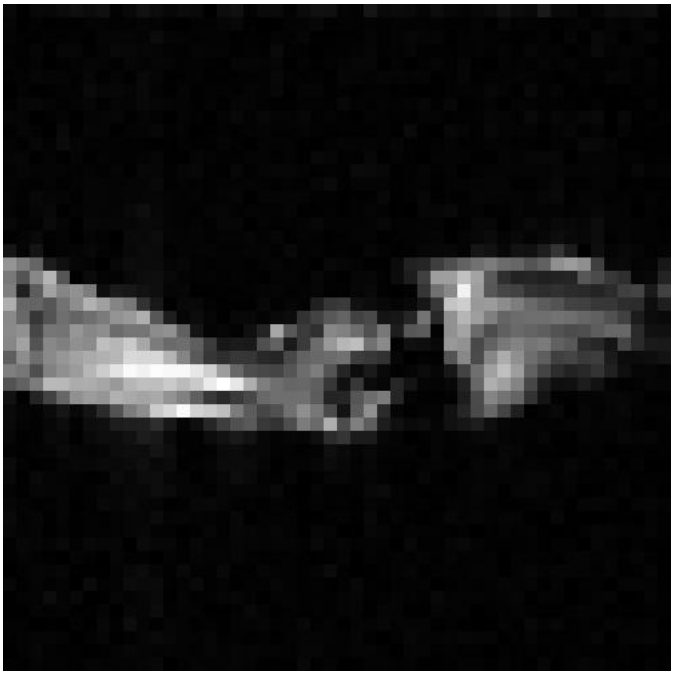


Figure 6.14: 50×50 pixels interpolated from high resolution image of bone.



(a) FROID



(b) FFT

Figure 6.15: 50×50 MR image of bone reconstructed by FROID and an FFT.

arguments in the objective function; for every point modelled in the image there is a real and imaginary component. As such, the reconstruction becomes more computationally expensive and more memory for data storage is required.

The magnitude images reconstructed by FROID and an FFT are shown in Figure 6.15, which are normalized to a maximum amplitude of 1. The image reconstructed by FROID contains more background noise, but these may be real features as the high resolution image also has background noise. Although the true image is not known, the high resolution image reconstructed by an FFT using the full dataset in k-space is interpolated to 50×50 pixels (see Figure 6.14) for comparison and RMS error calculations. It can be seen that the image reconstructed by FROID using the cropped dataset contained more of the features found in the high resolution image of the bone and this results in an RMS difference of 0.074. The image reconstructed by an FFT from the cropped dataset seems to be more blurred and there is a larger RMS difference of 0.095.

6.5 Performance constraints and limitations

With the current implementation of FROID in IDL, further testing with complex images and other more interesting k-space sampling trajectories (e.g., radial and spiral) can not be efficiently performed and therefore, lies outside the scope of this thesis. The IDL code is limited to handling data no greater than 110×110 points for real-valued images and 78×78 points for complex images due to constraints previously discussed in Section 6.1. This restricts the testing of high resolution MRI data and the ability to reconstruct images on the same scale as current MRI reconstruction engines.

The testing of FROID for this thesis is performed on a Sony VAIO VGN-FW275D laptop with Intel Centrino Core 2 Duo processor (2.26 GHz processing speed and 4 GB RAM), and Windows 7, 64-bit operating system. Table 6.4 lists the amount of time approximately required to perform the major tasks in FROID for reconstructing real-valued and complex images. The time required to reconstruct complex images significantly increases since the size of $\nabla^2 f(\mathbf{p_m})$ is increased by a factor of 4 (i.e., the matrix is doubled in both

Table 6.4: The Amount of Time required to Run FROID

Task	**Time Required for Real-Valued Images**	**Time Required for Complex Images**
Calculating constants	~ 58 s	~ 193 s
Solving for $\mathbf{d}^*_\mathbf{m}$	~ 4 s	~ 31 s
Minimizing $f(\mathbf{p_m})$ in 11 iterations	~ 140 s	~ 982 s

dimensions when accounting for real and imaginary values in the image) and the system of equations in (5.24) is doubled. Recall from Section 5.2 that the Hessian of the quadratic model, $(\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})])$, is factored using Cholesky decomposition at each iteration to solve for the optimal step. Cholesky factorization requires less memory than other factorization methods since the expression is decomposed into a lower triangular matrix and its transpose, and only one of the factors need to be stored. However, the factorization needs to be performed at every iteration, which can be computationally expensive; the number of operations required to compute the Cholesky factors is $\mathcal{O}(n^3)$. From iteration to iteration, $(\nabla^2 f(\mathbf{p_m}^{(n)}) + \lambda^* \text{diag}[\nabla^2 f(\mathbf{p_m}^{(n)})])$ is only modified in the diagonal elements and a more efficient factorization method that reuses some of the information from previous factors should be adopted at each iteration. Algorithms that perform rank 1 updates of Cholesky factors where only 1 diagonal element has changed exist and require $\mathcal{O}(n^2)$ operations to compute.[25] To modify the Hessian, n rank 1 updates need to be performed for changes in n diagonal elements, which can add to $\mathcal{O}(n^3)$ operations per iteration. Therefore, the use of Cholesky factorization is suitable only when the number of iterations required for the optimization routine to converge is relatively small or the size of the problem is small.

In the example of the "Gaussian" image (discussed in Section 6.3), FROID can reconstruct images from data that are undersampled to a certain amount without including any constraints in the optimization. The ability to reconstruct images from radial and spiral sampling trajectories has yet to be demonstrated. Besides the locations of the samples, other *a priori* information may be required to restrict the degrees of freedom and the number of possible solutions that FROID can converge to while still achieving a least squares fit in the Fourier data. Constraints can be incorporated into the objective function by including an additional cost(s) that increases the function value when a constraint(s) is violated.

Chapter 7

Future Work and Conclusions

In this thesis, FROID has been implemented for MR image reconstruction. The results presented in Chapter 6 demonstrates that FROID can reconstruct real-valued and complex images from data acquired in k-space, both simulated and experimental. Under certain circumstances, FROID outperforms an FFT. In particular, FROID possesses the flexibility to arbitrarily model points in the image and k-space independent of one another. Pixels in the image can be strategically positioned to correspond to locations of features to be reconstructed. Samples do not need to be modelled uniformly in k-space and FROID can accommodate irregular sampling trajectories. The experiment using random sampling was successful in that images could be reconstructed with similar quality to those reconstructed from evenly sampled data. Initial results using undersampled data shows promise, but more work implementing constraints to restrict the degrees of freedom in the image is required.

A significant amount of testing can still be performed using constrained optimization techniques. The reconstruction of MR images from radial, spiral, and other common nonuniform sampling trajectories using FROID can be further explored. Comparative analyses with images reconstructed from other algorithms described in the background of this thesis (e.g., compressed sensing and the gridding method) should also be performed.

The limitations discussed in Section 6.5 can be resolved by modifying the implementation of FROID. It was noted that as the number of iterations required for the optimization routine to converge increases, the use of Cholesky factorization becomes more computationally expensive. When solving for the step in the argument, $\mathbf{d_m}^{(n)}$, in equation (5.24), other factorization techniques where the computed factors can be updated more efficiently in successive iterations can be explored. For example, QR factorization or singular value decomposition (SVD) can be used. The SVD is more computationally expensive to compute initially, but updating the factors when only the diagonal elements of the Hessian are modified is trivial. One of the factors given by the SVD is a diagonal matrix containing the eigenvalues of the Hessian. Modifications to the diagonal elements of the Hessian are

reflected by modifying the diagonal matrix in the same manner. In [22], a method for updating QR factors related to the Hessian using rotational matrices (Givens rotation) is also proposed for implementing the Levenberg-Marquardt algorithm.

Implementing FROID using a lower level programming language such as C^{++} can also improve performance speed by avoiding overheads that come with programs such as IDL. Memory addressing and allocation can be better controlled and is not be limited to 1.2 GB per variable. Resolving speed and memory issues allows reconstruction of MR images with complex values to be further tested using FROID.

FROID is not restricted to modelling the surface of an image using bilinear interpolation. The formula discussed in Chapter 4 is merely an example of a working model of FROID. The underlying concept is to map a continuous surface of the image so that the integral Fourier transform can be applied and data in the Fourier domain can be sampled accordingly. Depending on the interpolation method used, an analytical expression of the Fourier transform may be derived. For future development, other models of FROID can be derived by using different interpolation techniques. Although bilinear interpolation has proven to be effective even when the image is smooth like a Gaussian function, other techniques that produce smoother surfaces, such as the use of Bezier curves, may be required and can be explored in future work.

The ability to reconstruct images from nonuniform samples through the use of FROID leads to different applications in MRI. As discussed earlier in this thesis, many accelerated imaging techniques use non-Cartesian sampling trajectories. It is expected that with further development in FROID, images can be reconstructed from a variety of sampling schemes commonly employed in MRI. Known sampling imperfections can also be accounted for when reconstructing images. For example, magnetic field gradients used in MRI are susceptible to perturbations caused by eddy currents. This may result in errors in the sampling locations of data in k-space. So, when performing rectilinear sampling, data may not be exactly equidistant. If these sampling irregularities can be measured, FROID can then be used to reconstruct images where the locations of the Fourier data modelled are adjusted accordingly. FROID can potentially be implemented using digital circuits and integrated into a reconstruction engine for an MRI scanner.

APPENDICES

Appendix A

Analytic Solution to Fourier Transform in FROID

The Fourier transform to be solved is

$$\begin{aligned} S_m(u,v) =& \mathcal{F}[\mathcal{I}(x,y)] \\ &= \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\int_{x_j}^{x_{j+1}}\int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(1)} e^{-i2\pi(ux+vy)}\,\mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\int_{x_j}^{x_{j+1}}\int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(2)} e^{-i2\pi(ux+vy)}\,\mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\int_{x_j}^{x_{j+1}}\int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(3)} e^{-i2\pi(ux+vy)}\,\mathrm{d}x\mathrm{d}y \\ &+ \sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\int_{x_j}^{x_{j+1}}\int_{y_k}^{y_{k+1}} \mathrm{T}_{j,k}^{(4)} e^{-i2\pi(ux+vy)}\,\mathrm{d}x\mathrm{d}y \end{aligned} \tag{A.1}$$

where the terms are defined as

$$\begin{aligned} \mathrm{T}_{j,k}^{(1)} &= p_{m_{j,k}}\left[1-\frac{x-x_j}{\Delta x}\right]\left[1-\frac{y-y_k}{\Delta y}\right] \\ \mathrm{T}_{j,k}^{(2)} &= p_{m_{j+1,k}}\left[\frac{x-x_j}{\Delta x}\right]\left[1-\frac{y-y_k}{\Delta y}\right] \\ \mathrm{T}_{j,k}^{(3)} &= p_{m_{j,k+1}}\left[1-\frac{x-x_j}{\Delta x}\right]\left[\frac{y-y_k}{\Delta y}\right] \\ \mathrm{T}_{j,k}^{(4)} &= p_{m_{j+1,k+1}}\left[\frac{x-x_j}{\Delta x}\right]\left[\frac{y-y_k}{\Delta y}\right]. \end{aligned} \tag{A.2}$$

Term 1

$$\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\int_{x_j}^{x_{j+1}}\int_{y_k}^{y_{k+1}} p_{m_{j,k}}\left[1-\frac{x-x_j}{\Delta x}\right]\left[1-\frac{y-y_k}{\Delta y}\right]e^{-i2\pi(ux+vy)}\,\mathrm{d}x\mathrm{d}y. \tag{A.3}$$

By isolating the constants and separating the x and y factors the following is given

$$\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\frac{p_{m_{j,k}}}{\Delta x\Delta y}\int_{x_j}^{x_{j+1}}(\Delta x+x_j-x)e^{-i2\pi ux}\,\mathrm{d}x\int_{y_k}^{y_{k+1}}(\Delta y+y_k-y)e^{-i2\pi vy}\,\mathrm{d}y. \tag{A.4}$$

Noting that

$$\int_{x_j}^{x_{j+1}} e^{-i2\pi ux}\,\mathrm{d}x=\frac{i(e^{-i\vartheta_{j+1}}-e^{-i\vartheta_j})}{2\pi u} \tag{A.5}$$

and

$$\int_{x_j}^{x_{j+1}} xe^{-i2\pi ux}\,\mathrm{d}x=\frac{(e^{-i\vartheta_{j+1}}[i\vartheta_{j+1}+1]-e^{-i\vartheta_j}[i\vartheta_j+1])}{4\pi^2u^2} \tag{A.6}$$

for $\vartheta_j=2\pi x_j u$, the integral over x can be solved. Similarly, the integral over y can be evaluated using $\varepsilon_k=2\pi y_k v$. After solving the x and y integrals separately, term 1 becomes

$$\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\frac{p_{m_{j,k}}}{\Delta x\Delta y}\left\{(\Delta x+x_j)(\Delta y+y_k)z_1-(\Delta x+x_j)z_2-(\Delta y+y_k)z_3+z_4\right\} \tag{A.7}$$

where

$$z_1=-\frac{(e^{-i\vartheta_{j+1}}-e^{-i\vartheta_j})(e^{-i\varepsilon_{k+1}}-e^{-i\varepsilon_k})}{4\pi^2uv}, \tag{A.8}$$

$$z_2=\frac{i(e^{-i\vartheta_{j+1}}-e^{-i\vartheta_j})(e^{-i\varepsilon_{k+1}}[i\varepsilon_{k+1}+1]-e^{-i\varepsilon_k}[i\varepsilon_k+1])}{8\pi^3uv^2}, \tag{A.9}$$

$$z_3=\frac{i(e^{-i\varepsilon_{k+1}}-e^{-i\varepsilon_k})(e^{-i\vartheta_{j+1}}[i\vartheta_{j+1}+1]-e^{-i\vartheta_j}[i\vartheta_j+1])}{8\pi^3u^2v}, \tag{A.10}$$

and

$$z_4=\frac{(e^{-i\vartheta_{j+1}}[i\vartheta_{j+1}+1]-e^{-i\vartheta_j}[i\vartheta_j+1])(e^{-i\varepsilon_{k+1}}[i\varepsilon_{k+1}+1]-e^{-i\varepsilon_k}[i\varepsilon_k+1])}{16\pi^4u^2v^2}. \tag{A.11}$$

Terms 2, 3, and 4

Using similar means as above, the Fourier transform of terms 2, 3 and 4 are evaluated as

$$\mathcal{F}[\mathrm{T}^{(2)}_{j,k}]=\frac{p_{m_{j+1,k}}}{\Delta x\Delta y}\left\{-x_j(\Delta y+y_k)z_1+x_jz_2+(\Delta y+y_k)z_3-z_4\right\}, \tag{A.12}$$

$$\mathcal{F}[\mathrm{T}^{(3)}_{j,k}]=\frac{p_{m_{j,k+1}}}{\Delta x\Delta y}\left\{-(\Delta x+x_j)y_kz_1+(\Delta x+x_j)z_2+y_kz_3-z_4\right\}, \tag{A.13}$$

and

$$\mathcal{F}[\mathrm{T}^{(4)}_{j,k}] = \frac{p_{m_{j+1,k+1}}}{\Delta x \Delta y} \left\{ x_j y_k z_1 - x_j z_2 - y_k z_3 + z_4 \right\}. \tag{A.14}$$

Separating the real and imaginary terms

Combining terms 1, 2, 3, and 4, $S_m(u,v)$ is equal to

$$\begin{aligned}
\sum_{j=1}^{N_d-1} \sum_{k=1}^{M_d-1} \Big[& p_{m_{j,k}} \left\{ (\Delta x + x_j)(\Delta y + y_k) z_1 - (\Delta x + x_j) z_2 - (\Delta y + y_k) z_3 + z_4 \right\} / (\Delta x \Delta y) \\
& + p_{m_{j+1,k}} \left\{ -x_j (\Delta y + y_k) z_1 + x_j z_2 + (\Delta y + y_k) z_3 - z_4 \right\} / (\Delta x \Delta y) \\
& + p_{m_{j,k+1}} \left\{ -(\Delta x + x_j) y_k z_1 + (\Delta x + x_j) z_2 + y_k z_3 - z_4 \right\} / (\Delta x \Delta y) \\
& + p_{m_{j+1,k+1}} \left\{ x_j y_k z_1 - x_j z_2 - y_k z_3 + z_4 \right\} / (\Delta x \Delta y) \Big].
\end{aligned} \tag{A.15}$$

The z factors can be further broken down into real and imaginary parts.

$$\begin{aligned}
\Re[z_1] = [& -\cos(\vartheta_{j+1} + \varepsilon_{k+1}) + \cos(\vartheta_{j+1} + \varepsilon_k) \\
& + \cos(\vartheta_j + \varepsilon_{k+1}) - \cos(\vartheta_j + \varepsilon_k)] / (4\pi^2 uv),
\end{aligned} \tag{A.16}$$

$$\begin{aligned}
\Re[z_2] = [& \sin(\vartheta_{j+1} + \varepsilon_{k+1}) - \varepsilon_{k+1} \cos(\vartheta_{j+1} + \varepsilon_{k+1}) \\
& - \sin(\vartheta_{j+1} + \varepsilon_k) + \varepsilon_k \cos(\vartheta_{j+1} + \varepsilon_k) \\
& - \sin(\vartheta_j + \varepsilon_{k+1}) + \varepsilon_{k+1} \cos(\vartheta_j + \varepsilon_{k+1}) \\
& + \sin(\vartheta_j + \varepsilon_k) - \varepsilon_k \cos(\vartheta_j + \varepsilon_k)] / (8\pi^3 uv^2),
\end{aligned} \tag{A.17}$$

$$\begin{aligned}
\Re[z_3] = [& \sin(\vartheta_{j+1} + \varepsilon_{k+1}) - \vartheta_{j+1} \cos(\vartheta_{j+1} + \varepsilon_{k+1}) \\
& - \sin(\vartheta_{j+1} + \varepsilon_k) + \vartheta_{j+1} \cos(\vartheta_{j+1} + \varepsilon_k) \\
& - \sin(\vartheta_j + \varepsilon_{k+1}) + \vartheta_j \cos(\vartheta_j + \varepsilon_{k+1}) \\
& + \sin(\vartheta_j + \varepsilon_k) - \vartheta_j \cos(\vartheta_j + \varepsilon_k)] / (8\pi^3 u^2 v),
\end{aligned} \tag{A.18}$$

$$
\begin{aligned}
\Re[z_4] = [&-\vartheta_{j+1}\varepsilon_{k+1}\cos(\vartheta_{j+1}+\varepsilon_{k+1}) + \vartheta_{j+1}\sin(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&+ \varepsilon_{k+1}\sin(\vartheta_{j+1}+\varepsilon_{k+1}) + \cos(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&+ \vartheta_{j+1}\varepsilon_k\cos(\vartheta_{j+1}+\varepsilon_k) - \vartheta_{j+1}\sin(\vartheta_{j+1}+\varepsilon_k) \\
&- \varepsilon_k\sin(\vartheta_{j+1}+\varepsilon_k) - \cos(\vartheta_{j+1}+\varepsilon_k) \\
&+ \vartheta_j\varepsilon_{k+1}\cos(\vartheta_j+\varepsilon_{k+1}) - \vartheta_j\sin(\vartheta_j+\varepsilon_{k+1}) \\
&- \varepsilon_{k+1}\sin(\vartheta_j+\varepsilon_{k+1}) - \cos(\vartheta_j+\varepsilon_{k+1}) \\
&- \vartheta_j\varepsilon_k\cos(\vartheta_j+\varepsilon_k) + \vartheta_j\sin(\vartheta_j+\varepsilon_k) \\
&+ \varepsilon_k\sin(\vartheta_j+\varepsilon_k) + \cos(\vartheta_j+\varepsilon_k),
\end{aligned}
\tag{A.19}
$$

$$
\begin{aligned}
\Im[z_1] = [&\sin(\vartheta_{j+1}+\varepsilon_{k+1}) - \sin(\vartheta_{j+1}+\varepsilon_k) \\
&- \sin(\vartheta_j+\varepsilon_{k+1}) + \sin(\vartheta_j+\varepsilon_k)]/(4\pi^2uv),
\end{aligned}
\tag{A.20}
$$

$$
\begin{aligned}
\Im[z_2] = [&\cos(\vartheta_{j+1}+\varepsilon_{k+1}) + \varepsilon_{k+1}\sin(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&- \cos(\vartheta_{j+1}+\varepsilon_k) - \varepsilon_k\sin(\vartheta_{j+1}+\varepsilon_k) \\
&- \cos(\vartheta_j+\varepsilon_{k+1}) - \varepsilon_{k+1}\sin(\vartheta_j+\varepsilon_{k+1}) \\
&+ \cos(\vartheta_j+\varepsilon_k) + \varepsilon_k\sin(\vartheta_j+\varepsilon_k)]/(8\pi^3uv^2),
\end{aligned}
\tag{A.21}
$$

$$
\begin{aligned}
\Im[z_3] = [&\cos(\vartheta_{j+1}+\varepsilon_{k+1}) + \vartheta_{j+1}\sin(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&- \cos(\vartheta_{j+1}+\varepsilon_k) - \vartheta_{j+1}\sin(\vartheta_{j+1}+\varepsilon_k) \\
&- \cos(\vartheta_j+\varepsilon_{k+1}) - \vartheta_j\sin(\vartheta_j+\varepsilon_{k+1}) \\
&+ \cos(\vartheta_j+\varepsilon_k) + \vartheta_j\sin(\vartheta_j+\varepsilon_k)]/(8\pi^3u^2v),
\end{aligned}
\tag{A.22}
$$

and

$$
\begin{aligned}
\Im[z_4] = [&\vartheta_{j+1}\varepsilon_{k+1}\sin(\vartheta_{j+1}+\varepsilon_{k+1}) + \vartheta_{j+1}\cos(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&+ \varepsilon_{k+1}\cos(\vartheta_{j+1}+\varepsilon_{k+1}) - \sin(\vartheta_{j+1}+\varepsilon_{k+1}) \\
&- \vartheta_{j+1}\varepsilon_k\sin(\vartheta_{j+1}+\varepsilon_k) - \vartheta_{j+1}\cos(\vartheta_{j+1}+\varepsilon_k) \\
&- \varepsilon_k\cos(\vartheta_{j+1}+\varepsilon_k) + \sin(\vartheta_{j+1}+\varepsilon_k) \\
&- \vartheta_j\varepsilon_{k+1}\sin(\vartheta_j+\varepsilon_{k+1}) - \vartheta_j\cos(\vartheta_j+\varepsilon_{k+1}) \\
&- \varepsilon_{k+1}\cos(\vartheta_j+\varepsilon_{k+1}) + \sin(\vartheta_j+\varepsilon_{k+1}) \\
&+ \vartheta_j\varepsilon_k\sin(\vartheta_j+\varepsilon_k) + \vartheta_j\cos(\vartheta_j+\varepsilon_k) \\
&+ \varepsilon_k\cos(\vartheta_j+\varepsilon_k) - \sin(\vartheta_j+\varepsilon_k).
\end{aligned}
\tag{A.23}
$$

All the real and imaginary constants for each term in equation (A.15) are collected to give

$$\begin{aligned} S_m(u,v) = &\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\left[p_{m_{j,k}}a_{j,k}^{(1)} + p_{m_{j+1,k}}a_{j,k}^{(2)} + p_{m_{j,k+1}}a_{j,k}^{(3)} + p_{m_{j+1,k+1}}a_{j,k}^{(4)}\right] \\ &+ i\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\left[p_{m_{j,k}}b_{j,k}^{(1)} + p_{m_{j+1,k}}b_{j,k}^{(2)} + p_{m_{j,k+1}}b_{j,k}^{(3)} + p_{m_{j+1,k+1}}b_{j,k}^{(4)}\right], \end{aligned} \tag{A.24}$$

where $a_{j,k}^{(n)}$ is the linear combination of the real components of z_1, z_2, z_3, and z_4 for term n at iteration j and k, and $b_{j,k}^{(n)}$ is the imaginary. Assuming that $p_{m_{j,k}}$ is equal to zero when $j = 1, N_d$ or $k = 1, M_d$ (i.e. the image has a zero intensity border), the summation can be shifted such that constants corresponding to the same points in the image are grouped together.

$$\begin{aligned} S_m(u,v) = &\sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1}\left[p_{m_{j,k}}a_{j,k}^{(1)} + p_{m_{j,k}}a_{j-1,k}^{(2)} + p_{m_{j,k}}a_{j,k-1}^{(3)} + p_{m_{j,k}}a_{j-1,k-1}^{(4)}\right] \\ &+ i\sum_{j=1}^{N_d-1}\sum_{k=1}^{M_d-1}\left[p_{m_{j,k}}b_{j,k}^{(1)} + p_{m_{j,k}}b_{j-1,k}^{(2)} + p_{m_{j,k}}b_{j,k-1}^{(3)} + p_{m_{j,k}}b_{j-1,k-1}^{(4)}\right] \end{aligned} \tag{A.25}$$

and

$$S_m(u_l,v_l) = \sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1} A_{j,k,l}p_{m_{j,k}} + i\sum_{j=2}^{N_d-1}\sum_{k=2}^{M_d-1} B_{j,k,l}p_{m_{j,k}}, \tag{A.26}$$

where

$$A_{j,k,l} = a_{j,k}^{(1)} + a_{j-1,k}^{(2)} + a_{j,k-1}^{(3)} + a_{j-1,k-1}^{(4)} \tag{A.27}$$

and

$$B_{j,k,l} = b_{j,k}^{(1)} + b_{j-1,k}^{(2)} + b_{j,k-1}^{(3)} + b_{j-1,k-1}^{(4)} \tag{A.28}$$

for points sampled in k-space at locations (u_l,v_l), $l = 1, ..., N_{samples}$.

Calculating the limits of the coefficients

Using l'Hôpital's rule the following limits are evaluated:

$$\lim_{u\to 0}\Re[z_1] = [x_{j+1}\sin(\varepsilon_{k+1}) - x_{j+1}\sin(\varepsilon_k) - x_j\sin(\varepsilon_{k+1}) + x_j\sin(\varepsilon_k)]/(2\pi v), \tag{A.29}$$

$$\begin{aligned} \lim_{u\to 0}\Re[z_2] = &[x_{j+1}\cos(\varepsilon_{k+1}) + x_{j+1}\varepsilon_{k+1}\sin(\varepsilon_{k+1}) \\ &- x_{j+1}\cos(\varepsilon_k) - x_{j+1}\varepsilon_k\sin(\varepsilon_k) \\ &- x_j\cos(\varepsilon_{k+1}) - x_j\varepsilon_{k+1}\sin(\varepsilon_{k+1}) \\ &+ x_j cos(\varepsilon_k) + x_j\varepsilon_k\sin(\varepsilon_k)]/(4\pi^2v^2), \end{aligned} \tag{A.30}$$

$$\lim_{u\to 0} \Re[z_3] = [x_{j+1}^2 \sin(\varepsilon_{k+1}) - x_j^2 \sin(\varepsilon_{k+1}) - x_{j+1}^2 \sin(\varepsilon_k) + x_j^2 \sin(\varepsilon_k)]/(4\pi v), \tag{A.31}$$

$$\begin{aligned}\lim_{u\to 0} \Re[z_4] = [&x_{j+1}^2 \varepsilon_{k+1} \sin(\varepsilon_{k+1}) + x_{j+1}^2 \cos(\varepsilon_{k+1}) \\ -\ &x_{j+1}^2 \varepsilon_k \sin(\varepsilon_k) - x_{j+1}^2 \cos(\varepsilon_k) \\ -\ &x_j^2 \varepsilon_{k+1} \sin(\varepsilon_{k+1}) - x_j^2 \cos(\varepsilon_{k+1}) \\ +\ &x_j^2 \varepsilon_k \sin(\varepsilon_k) + x_j^2 \cos(\varepsilon_k)]/(8\pi^2 v^2),\end{aligned} \tag{A.32}$$

$$\lim_{u\to 0} \Im[z_1] = [x_{j+1} \cos(\varepsilon_{k+1}) - x_{j+1} \cos(\varepsilon_k) - x_j \cos(\varepsilon_{k+1}) + x_j \cos(\varepsilon_k)]/(2\pi v), \tag{A.33}$$

$$\begin{aligned}\lim_{u\to 0} \Im[z_2] = [&-x_{j+1} \sin(\varepsilon_{k+1}) + x_{j+1} \varepsilon_{k+1} \cos(\varepsilon_{k+1}) \\ +\ &x_{j+1} \sin(\varepsilon_k) - x_{j+1} \varepsilon_k \cos(\varepsilon_k) \\ +\ &x_j \sin(\varepsilon_{k+1}) - x_j \varepsilon_{k+1} \cos(\varepsilon_{k+1}) \\ -\ &x_j \sin(\varepsilon_k) + x_j \varepsilon_k \cos(\varepsilon_k)]/(4\pi^2 v^2),\end{aligned} \tag{A.34}$$

$$\lim_{u\to 0} \Im[z_3] = [x_{j+1}^2 \cos(\varepsilon_{k+1}) - x_j^2 \cos(\varepsilon_{k+1}) - x_{j+1}^2 \cos(\varepsilon_k) + x_j^2 \cos(\varepsilon_k)]/(4\pi v), \tag{A.35}$$

$$\begin{aligned}\lim_{u\to 0} \Im[z_4] = [&x_{j+1}^2 \varepsilon_{k+1} \cos(\varepsilon_{k+1}) - x_{j+1}^2 \sin(\varepsilon_{k+1}) \\ -\ &x_{j+1}^2 \varepsilon_k \cos(\varepsilon_k) + x_{j+1}^2 \sin(\varepsilon_k) \\ -\ &x_j^2 \varepsilon_{k+1} \cos(\varepsilon_{k+1}) + x_j^2 \sin(\varepsilon_{k+1}) \\ +\ &x_j^2 \varepsilon_k \cos(\varepsilon_k) - x_j^2 \sin(\varepsilon_k)]/(8\pi^2 v^2),\end{aligned} \tag{A.36}$$

$$\lim_{v\to 0} \Re[z_1] = [y_{k+1} \sin(\vartheta_{j+1}) - y_{k+1} \sin(\vartheta_j) - y_k \sin(\vartheta_{j+1}) + y_k \sin(\vartheta_j)]/(2\pi u), \tag{A.37}$$

$$\lim_{v\to 0} \Re[z_2] = [y_{k+1}^2 \sin(\vartheta_{j+1}) - y_k^2 \sin(\vartheta_{j+1}) - y_{k+1}^2 \sin(\vartheta_j) + y_k^2 \sin(\vartheta_j)]/(4\pi u), \tag{A.38}$$

$$\begin{aligned}\lim_{v\to 0} \Re[z_3] = [&y_{k+1} \cos(\vartheta_{j+1}) + y_{k+1} \vartheta_{j+1} \sin(\vartheta_{j+1}) \\ -\ &y_{k+1} \cos(\vartheta_j) - y_{k+1} \vartheta_j \sin(\vartheta_j) \\ -\ &y_k \cos(\vartheta_{j+1}) - y_k \vartheta_{j+1} \sin(\vartheta_{j+1}) \\ +\ &y_k cos(\vartheta_j) + y_k \vartheta_j \sin(\vartheta_j)]/(4\pi^2 u^2),\end{aligned} \tag{A.39}$$

$$\begin{aligned}\lim_{v\to 0}\Re[z_4] = [&y_{k+1}^2\vartheta_{j+1}\sin(\vartheta_{j+1}) + y_{k+1}^2\cos(\vartheta_{j+1})\\ -\ &y_{k+1}^2\vartheta_j\sin(\vartheta_j) - y_{k+1}^2\cos(\vartheta_j)\\ -\ &y_k^2\vartheta_{j+1}\sin(\vartheta_{j+1}) - y_k^2\cos(\vartheta_{j+1})\\ +\ &y_k^2\vartheta_j\sin(\vartheta_j) + y_k^2\cos(\vartheta_j)]/(8\pi^2u^2),\end{aligned} \tag{A.40}$$

$$\lim_{v\to 0}\Im[z_1] = [y_{k+1}\cos(\vartheta_{j+1}) - y_{k+1}\cos(\vartheta_j) - y_k\cos(\vartheta_{j+1}) + y_k\cos(\vartheta_j)]/(2\pi u), \tag{A.41}$$

$$\lim_{v\to 0}\Im[z_2] = [y_{k+1}^2\cos(\vartheta_{j+1}) - y_k^2\cos(\vartheta_{j+1}) - y_{k+1}^2\cos(\vartheta_j) + y_k^2\cos(\vartheta_j)]/(4\pi u), \tag{A.42}$$

$$\begin{aligned}\lim_{v\to 0}\Im[z_3] = [&-y_{k+1}\sin(\vartheta_{j+1}) + y_{k+1}\vartheta_{j+1}\cos(\vartheta_{j+1})\\ +\ &y_{k+1}\sin(\vartheta_j) - y_{k+1}\vartheta_j\cos(\vartheta_j)\\ +\ &y_k\sin(\vartheta_{j+1}) - y_k\vartheta_{j+1}\cos(\vartheta_{j+1})\\ -\ &y_k\sin(\vartheta_j) + y_k\vartheta_j\cos(\vartheta_j)]/(4\pi^2u^2),\end{aligned} \tag{A.43}$$

$$\begin{aligned}\lim_{v\to 0}\Im[z_4] = [&y_{k+1}^2\vartheta_{j+1}\cos(\vartheta_{j+1}) - y_{k+1}^2\sin(\vartheta_{j+1})\\ -\ &y_{k+1}^2\vartheta_j\cos(\vartheta_j) + y_{k+1}^2\sin(\vartheta_j)\\ -\ &y_k^2\vartheta_{j+1}\cos(\vartheta_{j+1}) + y_k^2\sin(\vartheta_{j+1})\\ +\ &y_k^2\vartheta_j\cos(\vartheta_j) - y_k^2\sin(\vartheta_j)]/(8\pi^2u^2),\end{aligned} \tag{A.44}$$

$$\lim_{u,v\to 0}\Re[z_1] = \Delta x\Delta y, \tag{A.45}$$

$$\lim_{u,v\to 0}\Re[z_2] = \Delta x(y_{k+1}^2 - y_k^2)/2, \tag{A.46}$$

$$\lim_{u,v\to 0}\Re[z_3] = \Delta y(x_{j+1}^2 - x_j^2)/2, \tag{A.47}$$

$$\lim_{u,v\to 0}\Re[z_4] = (x_{j+1}^2 - x_j^2)(y_{k+1}^2 - y_k^2)/4, \tag{A.48}$$

$$\lim_{u,v\to 0}\Im[z_1] = 0, \tag{A.49}$$

$$\lim_{u,v\to 0}\Im[z_2] = 0, \tag{A.50}$$

$$\lim_{u,v\to 0}\Im[z_3] = 0, \tag{A.51}$$

and

$$\lim_{u,v\to 0}\Im[z_4] = 0. \tag{A.52}$$

References

[1] M. H. Levitt, *Spin Dynamics: Basics of Nuclear Magnetic Resonance*. West Sussex, England: John Wiley & Sons Ltd, second ed., 2008.

[2] Z. P. Liang and P. C. Lauterbur, *Principles of Magnetic Resonance Imaging - A Signal Processing Perspective*. New York, NY: IEEE Press, 2000.

[3] R. H. Hashemi, W. G. B. Jr., and C. J. Lisanti, *MRI - The Basics*. Philadelphia, PA: Lippincott Williams & Wilkins, 2004.

[4] J. P. Hornak, "The basics of MRI," 1996. http://www.cis.rit.edu/htbooks/mri/inside.htm.

[5] M. A. Bernstein, K. F. King, and X. J. Zhou, *Handbook of MRI Pulse Sequences*. London, UK: Elsevier Inc., 2004.

[6] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab, *Signals & Systems*. Upper Saddle River, NJ: Prentice Hall, Pearson Education Inc., second ed., 1997.

[7] E. Chu, *Discrete and Continuous Fourier Transforms - Analysis, Applications and Fast Algorithms*. Boca Raton, FL: Chapman & Hall/CRC, Taylor & Francis Group, 2008.

[8] M. Blaimer, F. Breuer, M. Mueller, R. M. Heidemann, M. A. Griswold, and P. M. Jakob, "SMASH, SENSE, PILS, GRAPPA: How to choose the optimal method," *Topics in Magnetic Resonance Imaging*, vol. 15, no. 4, 2004.

[9] E. G. Larsson, D. Erdogmus, R. Yan, J. C. Principe, and J. R. Fitzsimmons, "SNR-optimality of sum-of-squares reconstruction for phased-array magnetic resonance imaging," *Journal of Magnetic Resonance*, vol. 163, pp. 121–123, 2003.

[10] M. Griswold, P. M. Jakob, M. Nittka, J. W. Goldfarb, and A. Haase, "Partially Parallel Imaging With Localized Sensitivities (PILS)," *Magnetic Resonance in Medicine*, vol. 44, pp. 602–609, 2000.

[11] K. P. Pruessman, M. Weiger, M. B. Scheidegger, and P. Boesiger, "SENSE: Sensitivity Encoding for fast MRI," *Magnetic Resonance in Medicine*, vol. 42, pp. 952–962, 1999.

[12] M. A. Griswold, P. M. Jakob, R. M. Heidemann, M. Nittka, V. Jellus, J. Wang, B. Kiefer, and A. Haase, "Generalized Autocalibrating Partially Parallel Acquisitions (GRAPPA)," *Magnetic Resonance in Medicine*, vol. 47, pp. 1202–1210, 2002.

[13] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, 2006.

[14] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI." Magnetic Resonance Systems Research Laboratory, Department of Electric Engineering, Stanford University. http://www.stanford.edu/∼mlustig/CSMRI.pdf.

[15] A. R. Hajian, B. B. Behr, A. T. Cenko, R. P. Olling, D. Mozurkewich, J. T. Armstrong, B. Pohl, S. Petrossian, K. H. Knuth, R. B. Hindsley, M. Murison, M. Efroimsky, R. Dantowitz, M. Kozubal, D. G. Currie, T. E. Nordgren, C. Tycner, and R. S. McMillan, "Initial results from the USNO dispersed fourier transform spectrograph," *The Astrophysical Journal*, Jan 2007.

[16] A. R. Hajian, B. B. Behr, A. T. Cenko, K. H. Knuth, and D. G. Currie, "The FROID algorithm for spectral reconstruction," in *Fourier Transform Spectroscopy/Hyperspectral Imaging and Sounding of the Environment*, 2007. OSA Technical Digest Series, Optical Society of America, paper FWC3.

[17] C. Tycner and A. R. Hajian, "Continuous surface fitting to spatial optical interferometer data." United States Naval Observatory, Washington, DC, September 2006.

[18] J. A. Högbom, "Aperture synthesis with a non-regular distribution of interferometer baselines," *Astronomy Astrophysics Supplement*, vol. 15, pp. 417–426, 1974.

[19] T. J. Cornwell, "Hogboms CLEAN algorithm: Impact on astronomy and beyond," *Astronomy & Astrophysics*, vol. 500, pp. 65–66, 2009.

[20] J. Skilling and R. K. Bryan, "Maximum entropy image reconstruction: General algorithm," *Royal Astronomical Society Monthly Notices*, vol. 211, pp. 111–124, 1984.

[21] "IDL: Application programming," May 2009. ITT Visual Information Solutions. http://www.ittvis.com/ProductServices/IDL/ProductDocumentation.aspx.

[22] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, NY: Springer Science+Business Media, LLC., second ed., 2006.

[23] T. Sauer, *Numerical Analysis*. USA: Pearson Education Inc., Addison-Wesley., 2006.

[24] "Overcoming windows memory allocation limitations," November 2009. ITT Visual Information Solutions. http://www.ittvis.com/services/techtip.asp?ttid=3346.

[25] P. E. Gill, G. H. Golub, W. Murray, and M. A. Saunders, "Methods for modifying matrix factorizations," *Mathematics of Computation*, pp. 505–535, April 1974.