MOTION MONITOR FOR MOVEMENT IN MRI USING REFLECTIVE MARKER, OPTICAL FIBERS AND WEBCAM

A Thesis in Electrical Engineering
by
Abhijit S. Wander

© 2014 Abhijit S. Wander

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science

December 2014
The thesis of Abhijit S. Wander was reviewed and approved* by the following:

Robert A. Gray
Associate Professor of Electrical Engineering
Thesis Adviser

Mohammad Tofighi
Associate Professor of Electrical Engineering

Seth Wolpert
Associate Professor of Electrical Engineering

Susan Lemieux
Lecturer in Physics

Sedig S. Agili
Professor of Electrical Engineering
Graduate Program Coordinator

*Signatures are on file in the graduate school.
ABSTRACT

Magnetic resonance imaging (MRI) is an imaging technique used as a research modality for the study of human anatomy. Patient motion during MRI scanning remains a severe problem which degrades the diagnostic accuracy, posing significant problems in the acquisition and analysis of the MRI data. In this thesis, three algorithms were explored for real time motion detection of head movement in patients. The current motion detection methods are expensive and, to a degree, scanner specific making it problematic for inclusion in every scanner. The proposed research is a potential alternative, as it is scanner independent and based on low cost hardware namely webcam, non-coherent optical fiber bundles, reflective marker and a personal computer. One end of the optical fiber bundle is above the reflective marker placed on the head of patient and the other end is connected to the webcam. The proposed algorithms use the red, green and blue channels of the color of light emerging from the output end of the optical fiber bundle to determine the range of movement. Bench testing demonstrated that two of the algorithms seemed to have some shortcomings to be usable at this time. The third algorithm passed the bench tests outside the scanner for head-foot linear movement (like straightening of a body when lying on MRI table) and roll rotational movement (like shaking of the head for a no). Head-foot linear movement and pitch rotational movement (like nodding of the head for a yes) were varied to some degrees during calibration inside the scanner. To evaluate the effects of the movements on the resulting MRI images, MRI scan was taken to acquire images with motion. A simulation analysis was performed on an MRI image dataset obtained from internet. The results of the simulation were compared with the results of the images with motion acquired in the MRI scan.

The tests demonstrated that motion monitor was able to detect movements as small as 2.5 mm and 1 degree for the head-foot and roll movements respectively. A feedback system can be developed to inform the patients of movement which will help the patients to remain still during the MRI scan. The merits of the motion monitor device are its real time capability, scanner independent, low cost hardware and patient comfort.
# TABLE OF CONTENTS

LIST OF TABLES........................................................................................................... vii  
LIST OF FIGURES.......................................................................................................... viii  
LIST OF ABBREVIATIONS............................................................................................ xii  

Chapter 1. Introduction  
1.1 Problem................................................................................................................. 1  
1.2 Objectives............................................................................................................... 1  
1.3 Thesis structure...................................................................................................... 3  

Chapter 2. Background  
2.1 K-space matrix..................................................................................................... 4  
2.2 Image processing.................................................................................................. 6  
  2.2.1 Digital images.................................................................................................. 6  
  2.2.2 Image statistics............................................................................................... 7  
  2.2.2.1 Histogram.................................................................................................. 7  
  2.2.2.2 Mean......................................................................................................... 8  
  2.2.2.3 Standard deviation....................................................................................... 8  
  2.2.3 Correlation of images....................................................................................... 8  

Chapter 3. Literature Review  
3.1 Motion effects on MRI.......................................................................................... 9  
  3.1.1 Origin of motion artifacts............................................................................... 9  
  3.1.2 Literature review: motion artifacts................................................................. 9  
    3.1.2.1 An inexpensive scanner-independent motion monitor  
           (old system)................................................................................................. 11  
    3.1.2.2 Dynamic scan plane tracking using MR position monitoring.............. 15  
    3.1.2.3 Prospective real time motion correction using an external optical  
           tracking system......................................................................................... 18  
    3.1.2.4 Measurement and correction of microscopic head motion during  
           magnetic resonance imaging of the brain................................................. 19  

Chapter 4. Motion Detector Hardware and Software Development  
4.1 Hardware development......................................................................................... 24  
  4.1.1 Fiber optics.................................................................................................... 24  
  4.1.2 Webcam......................................................................................................... 26  
  4.1.3 Reflective markers......................................................................................... 27  
  4.1.4 Condenser lens.............................................................................................. 27  
4.2 Software development ....................................................................................... 28  
4.3 Analysis.............................................................................................................. 28  

Chapter 5. Simulation and Techniques for Motion Detection  
5.1 Design of the new system: Simulation of motion effects.................................... 31
5.1.1 Overview......................................................................................31
  5.1.1.1 Image format..............................................................................31
  5.1.1.2 Planes of the section: axial, coronal, sagittal........................31
  5.1.1.3 Type of rotations: pitch, roll, yaw..........................................33
  5.1.1.4 Type of rotations: Head-foot, left-right.................................34
  5.1.1.5 ImageJ......................................................................................35
  5.1.1.6 Testing systems.......................................................................36
  5.1.1.7 Ghosting percentage...............................................................37
  5.1.1.8 Methodology for motion simulation........................................37
    5.1.1.8.1 Stepwise implementation of motion simulation.................40
  5.1.1.9 Methodology for measurement of ghosting percentage............41
    5.1.1.9.1 Stepwise implementation of calculation of ghosting percentage..........................................................42
  5.1.1.10 Algorithm flowchart for motion simulation and measurement of
          ghosting percentage......................................................................43
  5.2 Correlation method I – Correlation using the stationary image as the
          reference......................................................................................44
    5.2.1 Stepwise implementation of correlation method I....................45
    5.2.2 Algorithm flowchart..................................................................47
  5.3 Correlation method II – Correlation using the database of images as
          the reference..................................................................................48
    5.3.1 Stepwise implementation of correlation method II....................49
    5.3.2 Algorithm flowchart..................................................................50
  5.4 Relative luminance method..............................................................51
    5.4.1 Stepwise implementation of red-green relative luminance method..........................................................53
    5.4.2 Algorithm flowchart..................................................................54

Chapter 6. Methods: Simulation and Experimental Measurements for Motion
Detection Techniques
  6.1 Simulation of MRI image dataset..................................................55
  6.2 Measurements using the correlation method I................................55
    6.2.1 Bench test experiments (outside the scanner)............................55
  6.3 Measurements using the correlation method II...............................57
    6.3.1 Bench test experiments (outside the scanner)............................57
  6.4 Measurements using the relative luminance method.......................58
    6.4.1 Bench test experiments (outside the scanner)............................58
    6.4.2 Experiments inside the scanner................................................59

Chapter 7. Results: Observations and Measurements
  7.1 Simulation of MRI image dataset..................................................60
  7.2 Measurement results using the correlation method I........................64
    7.2.1 Bench test experiments (outside the scanner)............................64
  7.3 Measurement results using the correlation method II.........................68
    7.3.1 Bench test experiments (outside the scanner)............................68
  7.4 Measurement results using the relative luminance method..................71
    7.4.1 Bench test experiments (outside the scanner)............................71
    7.4.2 Experiments inside the scanner................................................75
Chapter 8. Discussion, Conclusion and Future Work

8.1 Discussion ........................................................................................................... 79
8.2 Conclusion ........................................................................................................... 80
8.3 Future research work ......................................................................................... 81

Appendix .................................................................................................................... 82

A. Background and Theory of MRI (Magnetic Resonance Imaging) ....................... 82
B. Matlab code for simulation of motion effects and calculation of ghosting percentage.... 104
C. Matlab code for creating the reference images in correlation methods I and II.......... 109
D. Matlab code for calibration of correlation method I ............................................ 109
E. Matlab code for calibration of correlation method II ......................................... 111
F. Matlab code for calibration of relative luminance method .................................. 113
G. Matlab code for testing of relative luminance method ..................................... 113
H. Markers used for testing ................................................................................... 117

References ................................................................................................................. 120
LIST OF TABLES

Table 4.1: Specifications of the optical fiber cable.................................................................25
Table 4.2 Specifications of the webcam..................................................................................26
Table 5.1: Test system I...........................................................................................................36
Table 5.2: Test system II..........................................................................................................36
Table 5.3: The dataset information..........................................................................................37
Table 7.1: Simulation results of the tests for the Coronal SPGR dataset.................................61
Table 7.2: The meaning of positions I-IV................................................................................68
Table 8.1: Comparison of the properties of the Wasil’s system and developed system.........79
Table 8.2: Summary of the properties of the motion tracking methods and developed method.........................................................................................................................80
LIST OF FIGURES

Figure 1.1: x-y-z coordinate system of the face.................................................................2

Figure 2.1: Fourier transformation of raw FID signal (k-space matrix) to create an image.................................................................................................................................4

Figure 2.2: Row by row filling of k-space from top to bottom.........................................5

Figure 2.3: Images obtained through the different regions of k-space matrix..................5

Figure 2.4: Digital image of resolution X*Y........................................................................6

Figure 2.5: Data array of the color components of each pixel.......................................6

Figure 2.6: A test image “Lena” in color, grayscale and binary form............................7

Figure 2.7: Histogram of the grayscale form of ‘Lena’ image in Figure 2.23...............7

Figure 3.1: (a) MR Image with no deliberate motion (b) MR Image with deliberate motion.....9

Figure 3.2: (a) The optical fiber end of the PDA520 detector (b) End of the detector connected to the USB1208FS through coaxial cable..........................................................12

Figure 3.3: The Photodetector Responsivity curve of the PDA520 detector. The important thing to note is the peak response for PDA520 is at 980 nm ........................................12

Figure 3.4: USB-1208FS.................................................................................................13

Figure 3.5: The block diagram of the current system......................................................13

Figure 3.6: Schematic of the hardware setup......................................................................14

Figure 3.7: Prospective scan method (Imaging pulse sequence is locked to the moving brain)...15

Figure 3.8: Photographs in the left column show the assembly of the phantom. The locator coils are placed at the ends of the white rods shown in the photographs. The center column of photographs are images acquired using the original scan plane and the right column contain the images from the updated scan plane .................................................................16

Figure 3.9: The photographs in the left column show the position of the patient inside the bore. The center column of photographs are images acquired using the original scan plane and the right column contain the images from the updated scan plane...........................................17

Figure 3.10: The system setup inside the MRI room.......................................................18
Figure 3.11: (a) 9 out of the 25 corrected images for a single selected slice (b) Standard deviation map of the image intensities across all 25 images (c) Standard deviation map of the image intensities for a different slice (d) Standard deviation map obtained from the retrospectively realigned images for the same first slice.

Figure 3.12: (a) Camera and lighting unit (CLU) (b) 15 mm MPT marker.

Figure 3.13: Image results using gradient echo sequence with (a) no deliberate motion (b) motion.

Figure 3.14: Image results using turbo spin echo sequence (a) without motion correction (b) with motion correction.

Figure 3.15: Image results using MPRAGE sequence (a) without motion correction (b) with motion correction.

Figure 4.1: Hardware setup.

Figure 4.2: The spectral attenuation plot for the borosilicate fiber.

Figure 4.3: Logitech QuickCam Pro 9000.

Figure 4.4 Reflective markers.

Figure 4.5: Converging lens.

Figure 4.6: Schematic of the hardware setup.

Figure 4.7: (a) Optical fiber focused on the center of the marker (b) After movement, optical fiber focus is off-center now.

Figure 5.1: From top to bottom: sagittal, axial and coronal views of a three dimensional object.

Figure 5.2: MR Images of the brain in (a) axial plane (b) coronal plane (c) sagittal plane.

Figure 5.3: x-y-z coordinate system.

Figure 5.4: Pitch, roll and yaw rotations of a head.

Figure 5.5: MRI table in the xyz coordinate system.

Figure 5.6: ImageJ window.

Figure 5.7: Volume viewer plugin of the ImageJ.

Figure 5.8: Position of motion in k-space matrix.
Figure 5.9: Block Diagram of the process to create the image with motion..........................39
Figure 5.10: Block Diagram of the process to create the true signal and pure ghost regions……41
Figure 5.11: Algorithm flowchart: Simulation of motion effects........................................43
Figure 5.12: Block diagram of the construction of the correlation database..........................44
Figure 5.13: Block diagram of the comparison phase of the correlation method I..................44
Figure 5.14: Algorithm flowchart: Correlation method I.....................................................47
Figure 5.15: Block diagram of the working of correlation method II.................................48
Figure 5.16: Algorithm flowchart: Correlation method II.....................................................50
Figure 5.17: Marker used for testing in the relative luminance method ...............................51
Figure 5.18: Image diagram for the different versions of the red and green colors in the marker..................................................................................................................51
Figure 5.19: Block diagram of the working of the relative luminance method ....................52
Figure 5.20: Algorithm flowchart: Relative luminance method...........................................54
Figure 6.1: Reference images obtained in the webcam with the lens on and off for (a) Marker I
(b) Marker II..................................................................................................................56
Figure 6.2: Reference images obtained in the webcam with the lens on and off for Marker I..................................................................................................................57
Figure 6.3: Reference images obtained in the webcam with the lens on and off for Marker I..................................................................................................................58
Figure 7.1: Coronal SPGR scan: MRI image of the brain (a) without motion (b) after introducing
a motion of 4.25 mm in a right-to-left direction at the middle of the k-space (c) after introducing
a motion of 4.25 mm in a right-to-left direction at the edge of the k-space.........................60
Figure 7.2: Coronal SPGR scan: Ghosting percentage for (a) linear movements: head-foot and
left-right (b) rotational movements: yaw and pitch.........................................................62
Figure 7.3: Line plots of the center of the slice number “30” in the coronal SPGR dataset for the
original image, and the image with a left-right movement of 4.25 mm introduced at the middle
of the k-space ..................................................................................................................63
Figure 7.4: Line plots of the center of the slice number “30” in the coronal SPGR dataset for the original image, and the image with a left-right movement of 4.25 mm introduced at the edge of the k-space.

Figures 7.5-7.8 Correlation coefficients data for the correlation method I.

Figure 7.9: The two ends of the optical fiber bundle with different orientation of fibers at each end.

Figures 7.10-7.13 Correlation coefficients data for the correlation method II.

Figure 7.14: Marker with the lens on: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (bench tests).

Figure 7.15: Marker with the lens on: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (bench tests).

Figure 7.16: Marker with the lens off: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (bench tests).

Figure 7.17: Marker with the lens off: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (bench tests).

Figure 7.18: Motion monitor output during testing.

Figure 7.19: Marker with the lens off: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (experiments inside the scanner).

Figure 7.20: Marker with the lens off: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (experiments inside the scanner).

Figure 7.21: TSE scan (Axial): Images with motion for 2.5 mm and 5 mm linear head movement at the middle of the k-space and edge of the k-space.

Figure 7.22: TSE scan (Sagittal): Images with motion for 2.5 mm and 5 mm linear head movement at the middle of the k-space and edge of the k-space.

Figure 7.23: TSE scan (Axial): Ghosting percentage for head-foot linear movements at the middle of the k-space and edge of the k-space.

Figure 7.24: TSE scan (Sagittal): Ghosting percentage for head-foot linear movements at the middle of the k-space and edge of the k-space.
**LIST OF ABBREVIATIONS**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRI</td>
<td>Magnetic Resonance Imaging</td>
</tr>
<tr>
<td>LED</td>
<td>Light Emitting Diode</td>
</tr>
<tr>
<td>IR</td>
<td>Infrared</td>
</tr>
<tr>
<td>DAQ</td>
<td>Data Acquisition</td>
</tr>
<tr>
<td>NMR</td>
<td>Nuclear Magnetic Resonance</td>
</tr>
<tr>
<td>SE</td>
<td>Spin Echo</td>
</tr>
<tr>
<td>GE</td>
<td>Gradient Echo</td>
</tr>
<tr>
<td>SPGR</td>
<td>Spoiled Gradient Echo</td>
</tr>
<tr>
<td>FID</td>
<td>Free Induction Delay</td>
</tr>
</tbody>
</table>
Chapter-1

Introduction

1.1 Problem

Magnetic resonance imaging (MRI) is an imaging technique used as a clinical diagnostic tool and as a research modality for the study of human anatomy. Patient motion during MRI scanning remains a severe problem which degrades the images, posing significant problems in the acquisition and analysis of the MRI data. The MRI environment can cause anxiety as the confined space in the scanner makes people nervous resulting in more motion. Some patients, like younger children and older people, face even more problems than young or middle-aged adults staying still for longer periods of time.

There are certain conditions that the motion correction strategy must meet to robustly remove image artifacts. The technique should work without human interference and be automatic. It should not add to the setup time or the data acquisition time. The patient should feel relaxed and be comfortable with the hardware, as often the patients are in pain during the scan. The patients are more likely to feel anxious due to pain, which can again lead to motion in the images. Ideally, it should not require any additional hardware as it may increase the overall cost of the scan. It should not require any post processing techniques to be applied on the image after the completion of the scan as it will increase the reconstruction time. Unfortunately, there is no such ideal technique currently in use and every technique has its own advantages and disadvantages.

1.2 Objectives

Much previous research has been done to track or correct the motion in MRI scanning using video cameras. These systems can be reliable and accurate but they are also very expensive because they must operate in the MRI scanning environment.

The aim of the research presented here, was to develop a method to allow the construction of a monitoring system based on low cost hardware: webcam, optical fibers, reflective marker and a personal computer without using MRI compatible video camera. The system comprised of one reflective marker placed on the temple of the patient and an optical fiber bundle fixed 8 mm away from the reflective marker. This part of the system was inside the MRI room. The other end of the optical fiber bundle was connected to the webcam situated outside the MRI room. The system was designed using optical fibers as they can work in the strong magnetic field without affecting image quality. They are also known for their easy installation.
This thesis describes the simulation, the development and the testing of the three algorithms to detect motion during the MRI scan. The primary goal of the simulation was to quantify the motion artifacts with respect to the amount of motion and position of motion. The simulation aids in setting the threshold for the smallest motion which will be detected by the developed system.

The developed algorithms, analyzed the color or illumination of the color received at the end of the optical fiber connected to the webcam. The algorithms used the red, green and blue channels of the color of light emerging from the output end of the optical fiber bundle to determine the range of movement. The first two algorithms were based on Pearson’s correlation while the third algorithm was based on the relative luminance principle. All three algorithms provide a live feedback to the MRI scanner operator.

Firstly, the calibration and the testing of the algorithms were performed outside the MRI scanner to test the performance of the algorithms before testing them inside the MRI scanner. Experiments involved moving the phantom replica along the z direction, shown in Figure 1.1. The motion in z direction represents the nodding of head or ‘straightening,’ a translational movement in or out of the bore. Patients often make this motion during MRI scanning to “get comfortable”.

**Figure 1.1:** x-y-z coordinate system of the face (taken from the internet source: smartrifle.deviantart.com)
This thesis not only provides the experimental demonstrations of the algorithms but also the simulation results.

1.3 Thesis structure

The thesis is organized as follows. Chapter 2 contains the k-space and image processing concepts. Chapter 3 provides an extensive literature review of the various motion prevention, correction and tracking methods. Chapter 4 explains the hardware and software development of the motion detector. Chapter 5 gives an introduction of methods and explains the working of the implemented algorithms. Chapter 6 contains the experimental measurements for the implemented algorithms for motion detection. Chapter 7 contains the results of the proposed algorithms followed by discussion, conclusion and future work in Chapter 8.
Chapter-2

Background

2.1 K-space matrix

K-space is the matrix where the data from the receiver coils of the MRI scanner gets stored prior to a Fourier transformation to the final reconstructed image. Some method needs to be applied on the raw data to convert them into readable MR images. Fourier Transform is the most common method used in MRI for image reconstruction. The signal obtained in MRI scan is in the frequency domain. The signal obtained from the MRI scanner can be expressed as [1]

\[ s(t) = \int_{x} \int_{y} m(x, y) e^{-2\pi i [k_x(t)x + k_y(t)y]} \, dx \, dy \]  

[2.1]

where \( m(x,y) \) is the 2D image to be reconstructed for a particular slice thickness, and \( k_x \) and \( k_y \) are the spatial frequencies along the x-direction and y-direction respectively. For 2-D image reconstruction, 2D inverse Fourier Transform is used on raw signal \( s(t) \) to reduce it to \( m(x,y) \). Every point in the k-space (raw data) matrix contains part of the image. There is no one-to-one correspondence between the k-space and the MRI image.

![Figure 2.1: Fourier transformation of raw FID signal (k-space matrix) to create an image [1]](image)

Generally, the filling of k-space takes place horizontally, one row after the other from top to bottom.
Figure 2.2: Row by row filling of k-space from top to bottom

In this type of k-space filling, the center of k-space is the low frequency region which represents the general contrast of the image. The outer regions of the k-space are the high frequency regions containing the borders or edges of the image. Figure 2.3 shows the images obtained if you have just the central and outer regions in k-space compared to the whole k-space matrix. This will be important later during the discussion of motion during MRI scanning.

Figure 2.3: Images obtained through the different regions of k-space matrix
2.2 Image processing

The following concepts of pixels, color and correlation are introduced for later use.

2.2.1 Digital images

A digital image is a discrete function of two variables, for example, \( f(x,y) \), which has been quantized over its domain and range \([2]\). The image is composed of discrete set of small rectangles called pixels. A pixel is the smallest component of an image. The value of each pixel is dependent on the nature of the image (binary, grayscale or color). An image of \( X \) rows and \( Y \) columns will have a resolution of \( X*Y \) as shown in Figure 2.4.

![Figure 2.4: Digital image of resolution X*Y](image)

A color image can be represented in \( m \)-by-\( n \)-by-3 data array with red, green and blue color components for each pixel as shown in Figure 2.5.

![Figure 2.5: Data array of the color components of each pixel](image)
A color image measures the chrominance and intensity with each pixel having three values. A grayscale image only measures the light intensity with each pixel representing the brightness. The intensity values for a grayscale image range from 0 to 255 with black having the lowest and white having the highest intensity. The binary image has only two values 0 (Black) and 1 (White). An example of each image has been shown in Figure 2.6.

![Color, Grayscale, and Binary Images](image_url)

**Figure 2.6:** A test image “Lena” in color, grayscale and binary form (taken from the internet source: inperc.com)

### 2.2.2 Image statistics

#### 2.2.2.1 Histogram

The histogram of an image plots the relative frequency of each pixel value in the image. It shows the number of pixels in an image for each intensity level [3] as shown in Figure 2.7. Histogram is one of the most important concepts of Image Processing.

![Histogram of Lena's Grayscale Image](image_url)

**Figure 2.7:** Histogram of the grayscale form of ‘Lena’ image in Figure 2.6
2.2.2.2 Mean

The image mean is the average of all the pixel values in an image. This mean is equal to the average intensity of an image in case of grayscale. Let the image \( a(x,y) \) be a two dimensional image with a mean of \( E[a] \). Equation (2.2) shows the equation for mean of \( a(x,y) \) [4].

\[
E[a] = \frac{1}{X \times Y} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} a(x,y)
\]  

[2.2]

2.2.2.3 Standard deviation

Standard deviation of an image gives a spread of the pixel values around the mean of image. Standard deviation is calculated by squaring all the deviations and then taking the average, followed by square root to compensate for the squaring in the initial step. The mathematical form to calculate the standard deviation [4] is shown in equation (2.3).

\[
\sigma = \sqrt{\frac{1}{X \times Y} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} \left( a(x,y) - E[a] \right)^2}
\]

[2.3]

2.2.3 Correlation of images

The correlation can be used to figure out the similarity between two images. In the correlation between two images, each pixel in the first image is correlated with each pixel in the second image at the same position. The \texttt{corr2} function in Matlab, which will be shown later on in the code section, follows the Pearson correlation for 2D arrays [3] as shown in equation (2.4).

\[
r = \frac{\sum_x \sum_y (A_{xy} - E[A])(B_{xy} - E[B])}{\sqrt{\sum_x \sum_y (A_{xy} - E[A])^2 \sum_x \sum_y (B_{xy} - E[B])^2}}
\]

[2.4]
Chapter-3
Literature Review

3.1 Motion effects on MRI

3.1.1 Origin of motion artifacts

In MRI, the motion of a patient causes artifacts in the images. Motion is normally random or physiological such as cardiac pulsation or respiratory which results in artifacts like ghosting and blurring. The imaging results shown in Figure 3.1 have been obtained using a gradient echo sequence at 1.5 T [5] to show the effect of random motion artifacts on the brain images.

![Figure 3.1: (a) MR Image with no deliberate motion (b) MR Image with deliberate motion [5]](image)

3.1.2 Literature review: motion artifacts

**Motion Prevention**

The most common method used to avoid motion artifacts in the images is physical restraint. Using cushions and bean bags, as physical restraints to stop the motion is the most commonly used method; it is easy to use but not rigorous enough to stop all movement. Custom made bite-bars and head holders can reduce the motion greatly, but they need to perfectly fit the patient. The disadvantage of custom-made products is the inconvenience of constructing personalized...
restraints for each patient. In addition, these physical restraints can cause physical discomfort which can lead to more motion. Sedation of the patient is also an option to reduce motion but it is rarely used due to safety concerns; it also requires the presence of an anesthesiologist for monitoring [6].

Hennig et al. [7] introduced the Fast spin echo, which was the first fast imaging technique. Fast imaging usually results in fewer motion artifacts. In this method, multiple lines of k-space are acquired in a single TR, instead, of acquiring a single line of k-space in every TR in the regular spin echo. This method reduced the acquisition time significantly but it also degraded the overall signal-to-noise ratio because of the reduced imaging time. This did reduce motion artifacts.

**Motion Correction**

Ehman and Felmlee in 1989 proposed the use of “Navigator Echoes” as a motion correction technique [8]. Navigator echoes detect the motion of the patient using an additional MR signal. In its elementary form, a single line of k-space data that passes through the middle of k-space is the navigator echo. This single line of k-space data at the middle of k-space matrix is regularly sampled during the MRI scan, while the other k-space data are being acquired. The middle of k-space contains the low frequency components which represents the overall contrast of the resultant image, so even a small amount of motion alters the data. The detected motion data can be used either to discard the corrupted data or to make corrections in the corrupted data. The disadvantage of the Navigator echo method is the additional time used for calculating navigator echoes which leads to an increased scan time.

Another one of the motion correction methods is PROPELLER (Periodically Overlapping Parallel Lines with Enhanced Reconstruction) [9]. The PROPELLER method is excellent for reducing some of the in-plane and through-plane motion, but the disadvantage with this method is the increased scan time. The reason for this additional time required is due to oversampling of the middle of the k-space region which increases the scan time. Also, extra time is spent for post processing on the image data.

**Motion Tracking**

The motion tracking methods require additional hardware. The additional hardware is used to collect information required for correcting the motion after the scan (retrospective motion correction) and/or adjusting the scanner gradients thereby averting the need for the post processing stage (prospective motion correction). Motion tracking systems for the prospective motion correction have the advantage of not requiring a post processing stage [6].

The concept of adjusting the scanner gradients first started in 1997 [10]. Much progress has been made since then, in developing an effective system.
Prospective motion correction methods for motion tracking related to the developed system are explained in detail below.

### 3.1.2.1 An inexpensive scanner-independent motion monitor (old system)

This motion monitor system, which is the precursor to the current system, based on optical tracking was developed by Muhammad Wasil Wahi-Anwar [11] to track patient or participant movement. The entire system was composed of five hardware components and one software component.

The software and hardware components used in this method has been explained in detail below.

1. **White LED flashlight** – The Monster flashlight pro T6-1000 was used as the white LED source with a 2800mA total output and brightness at 1000 lumens max (Flashlight Freak LLC, Millburn, NJ, USA). The wavelength range of output white light is 400-800 nm.

2. **Optical fibers** – Two borosilicate optical fibers used in the system were of dimensions .250” diameter and 10 m length with .55 NA (custom built by Fiberoptics Systems, Inc. SimiValley, CA, USA). The wavelength for the borosilicate fiber was 400-1600 nm with an approximate consistent attenuation of .9dB/m for the 420-1600 nm range light [11].

3. **Reflector** – Yellow color bicycle reflectors were used as the reflective medium (Fuelbelt Inc., Georgia, VT, USA). These reflectors were cut from an engineering-grade film composed of the microscopic glass spheres that have the ability to reflect the light back to its origin.

4. **PDA520 Detector** - PDA520 was a high precision, high accuracy, low noise, and switchable-gain silicon detector designed for the detection of the light in the wavelength range of 400 to 1100nm (Thorlabs Inc., Newton NJ, USA).
Figure 3.2: (a) The optical fiber end of the PDA520 detector (b) End of the detector connected to the USB1208FS through coaxial cable

Figure 3.3: The Photodetector Responsivity curve of the PDA520 detector. The important thing to note is the peak response for PDA520 is at 980 nm [12].

5. USB1208FS – USB-1208FS from Measurement Computing was a low speed USB device that converted the analog input from the PDA520 detector to a digital output to be fed as input into the Matlab software through the USB 2.0 port (Measurement Computing Corp., Norton MA, USA).
6. Instacal And Matlab – Instacal software was also a product of Measurement Computing (Measurement Computing Corp., Norton MA, USA) used to configure the USB-1208FS Board. The DAQ toolbox of Matlab (Mathworks, Inc., Natick MA, USA) was used to perform the required operations on the digital input from the USB-1208FS to obtain the desired output.

The block diagram of the current system shown in Figure 3.5 is a schematic diagram of the system. Figure 3.6 show a diagram of this system.

**Figure 3.4:** USB-1208FS

**Figure 3.5:** The block diagram of the current system
The system working as used is presented here:

A white LED flashlight was used as the light source to transmit the light through the optical fiber. The light coming out of the optical fiber was illuminated on a reflective marker of 2 cm diameter placed on the temple of the patient. The fibers were focused on the temple area of the patient at a distance of 1 cm [11].

The light reflected from the marker was transmitted to the detector through the receiving optical fiber. The detector was set at a gain of 10 dB and it converted the light to an analog voltage signal. A 50 ohm BNC cable was connected to the output of the detector to feed the analog voltage to the MCC analog-to-digital converter [11].

USB1208FS converted the analog voltage signal received from the detector into a digital voltage level analyzable by Matlab. The data were continuously acquired till the end of scan and sampled every 50 milliseconds. The data were compared to the running mean of the overall data in real time to detect signal variations. Also, the data were saved to perform further analysis on it later [11].

Trials were performed on humans and phantom for nodding and straightening movements. Trials were conducted with the motion monitor running to observe the changes in the magnitude of the digital signal acquired in Matlab. The motion was introduced into the middle of k-space, 2 minutes into the scan, for a total scan time of 4 minutes. In the other case, motion was introduced at the edge of k-space, 30 seconds into the scan [11].

**Figure 3.6:** Schematic of the hardware setup
The system developed in this research was able to detect both the motion movements in real time, but it did not provide data on the distance and direction of motion. There was no linear relationship between the signal received in the device and the motion magnitude. One important result as anticipated from the trials was there is more image artifact when the motion occurs in the middle of the k-space. There was a correlation between the amount of motion and the detected signal change but it was not calibratable [11].

3.1.2.2 Dynamic scan plane tracking using MR position monitoring

The method proposed by Derbyshire [13] was one of the initial motion tracking systems. The patient motion was tracked using three locator coils attached to the patient in a triangular configuration during the entire duration of the MRI scan.

The patient motion was treated as a rigid body transformation. Any motion in the patient also shifted the positions of the locator coils. The updated positions of the coils with respect to their initial positions were fed back to the MRI scanner to make the needed adjustments in the scan plane. The imaging scan plane was updated for every imaging scan, relative to the changing position of the locator coils.

![Diagram](image_url)

**Figure 3.7:** Prospective scan method (Imaging pulse sequence is locked to the moving brain)

Figure 3.7 shows a generalized step by step procedure of the prospective scan method.

Testing was performed on a phantom and human head of a volunteer to test the robustness of the scan update process.

In the experiment involving phantom, image pairs were acquired (in the axial plane) with one image acquired using the original scan plane and the other with the updated scan planes as shown.
in Figure 3.8. Gradient echo imaging was the technique used in the phantom experiment with the scan parameters TR = 33 ms, TE = 8 ms, flip angle = 30 degrees, FOV = 24 cm and 5-mm slice thickness [13].

Figure 3.8: Photographs in the left column show the assembly of the phantom. The locator coils are placed at the ends of the white rods shown in the photographs. The center column of photographs are images acquired using the original scan plane and the right column contain the images from the updated scan plane [13]

For the human experiment, the volunteer wore a swimming cap with a “+” shaped marker on it. The marker was placed on the cap to identify the orientation of the head from the photographs
inside the bore. Locator coils, three, were placed on the forehead and one behind each ear. Gradient echo imaging with the scan parameters TR = 33 ms, TE = 8 ms, flip angle = 30 degrees, FOV = 24 cm and 5-mm slice thickness was used in the experiment [13].

Figure 3.9: The photographs in the left column show the position of the patient inside the bore. The center column of photographs are images acquired using the original scan plane and the right column contain the images from the updated scan plane [13]

Post processing of the images obtained from the phantom and human experiments determined the accuracy of their registration to the images of the original position. The in-plane translational errors, in terms of the sub-pixel precision estimates were 0.00, 0.57, 1.33 and 1.27 pixels respectively for the phantom images shown in Figure 3.8. For the human images shown in Figure 3.9, the errors recorded were 0.157, 1.08, 1.17 and 3.65 respectively [13].
The advantage of using this method over optical tracking methods is that it does not require a clear line of sight from the markers to the light detectors.

### 3.1.2.3 Prospective real time motion correction using an external optical tracking system

Zaitsev demonstrated [14] a prospective real time motion correction system using an external tracking system consisting of two infrared cameras. A 3T Siemens system was used for the experiments. The two IR cameras were positioned in the MRI room as shown in Figure 3.10.

![Figure 3.10: The system setup inside the MRI room [14]](image)

The motion detected by the cameras was sent to a PC located outside the MRI room. The PC was used to calculate the variations in translations and rotations and to send the results to the MRI scanner, at a refresh rate of 60 Hz, to update the scanner gradients (refer to Figure 3.7) [14].
For experiments, the gel or water filled phantoms contained 4 reflective markers out of which 3 had to be visible at all times for determining the position.

The results of one of the imaging methods used on this system are shown in Figure 3.11. A 3D gradient echo sequence with the scan parameters TE = 2.5 ms, TR = 5.8 ms, flip angle = 10 degrees and resolution of 1×1×2 mm was used. In this imaging method, the phantom was scanned in 25 positions each rotated 30° from the last. Figures 3.11(b) and (c) show examples of the standard deviation maps of image intensities across all 25 images. Figure 3.11(d) show the standard deviation map obtained from the retrospectively realigned [14].

![Figure 3.11](image)

**Figure 3.11:** (a) 9 out of the 25 corrected images for a single selected slice (b) Standard deviation map of the image intensities averaged across all 25 images (c) Standard deviation map of the image intensities for a different slice (d) Standard deviation map obtained from the retrospectively realigned images for the same slice as in (d) [14]

The advantages of the real time prospective motion correction can be clearly seen from the results in Figure 3.11. The residual errors can reach .25 mm and .45 degrees for the large scale motion, while for the small scale motion the errors are negligible [14].

The disadvantage with this system was the requirement of a clear view from the cameras down the scanner on the patient. Another disadvantage of this system was the 60 Hz frame rate of cameras which is too slow to be used for fast imaging methods (TR> 17 ms) [14].

### 3.1.2.4 Measurement and correction of microscopic head motion during magnetic resonance imaging of the brain

In another method [5], an MR compatible optical tracking system comprising of a single camera, lighting Unit and 15 mm marker was developed for very small motion correction at different
field strengths. Figure 3.12(a) and (b) show the camera with the lighting unit and 15 mm MPT marker developed for this work.

![Image](image.png)

Figure 3.12: (a) Camera and lighting unit (CLU) (b) 15 mm MPT marker [5]

The Moiré Phase Tracking (MPT) marker used in the system was composed of layers of planar gratings. These layers generate moiré patterns whose phase is dependent on the orientation of marker. The phase was used to compute the through plane parameters using a curve fitting algorithm. Computer vision methods were used to compute the other four degrees of freedom. CLU is attached inside the bore at the top with a clear line of sight to the marker [5].
The experiments were performed at various field strengths using different imaging sequences. In the experiment performed at 1.5 T, a gradient echo sequence with the scan parameters $TE = 4.05$ ms, $TR = 13$ ms, flip angle = 15 degrees, in-plane resolution of $1 \text{ mm} \times 1 \text{ mm}$, through-plane resolution of $2 \text{ mm}$ and acquisition time = 4 min. 40 s was used. The image results for this experiment are shown in Figure 3.1. Results with no deliberate motion shown in Figure 3.1(a) demonstrate that there is no visible improvement in the image quality. For the case of deliberate motion of about 25 mm shown in Figure 3.1(b), this method has significantly improved the image quality. So this method was useful for the larger motion at field strength of 1.5 T [5].

![Figure 3.1: Image results using gradient echo sequence with (a) no deliberate motion (b) motion [5](a)](image)

![Figure 3.13: Image results using gradient echo sequence with (a) no deliberate motion (b) motion [5](b)](image)
The image results for the 2D turbo spin echo sequence at 3 T (TE = 87 ms, TR = 6500 ms, an in-plane resolution of 0.3 mm × 0.3 mm, through-plane resolution of 3 mm, acquisition time = 9 min. 45s) are shown in Figure 3.14. For the experiment, the patient tried to remain as stationary as possible without intentional motion. An improvement in the small features and reduction of ghosting artifacts is clearly visible [5].

![Figure 3.14: Image results using turbo spin echo sequence (a) without motion correction (b) with motion correction [5]](image)

The results for the 3D MPRAGE sequence at 7 T (TE = 2.69 ms, TR = 2500 ms, flip angle = 5 degrees, isotropic resolution of 0.6 mm, acquisition time = 14 min. 3s) are shown in Figure 3.15. For the experiment, the patient although well trained, moved the head by millimeters. Fourteen minute scans are too long for the patient to remain still. An improvement in the image quality is visible in the image with motion correction [5].
Figure 3.15: Image results using MPRAGE sequence (a) without motion correction (b) with motion correction [5]

The results obtained for these three tests indicate that this method can improve the image quality, both for intentional and unintentional movements using the motion tracking data for prospective motion correction.

All the motion tracking methods mentioned above (except the method proposed by Muhammad Wasil) used expensive equipment to gather the real time motion tracking data. An MR compatible camera is the most common hardware in all the optical motion tracking methods. The camera cost is high as it is expensive to make the camera MR compatible. But the objective of the proposed method for this work is to allow the construction of a monitoring system that is sufficiently fast to implement, using low cost hardware, and to detect movements as small as 2.5 mm. So, the proposed system will be more fitting if the cost factor is a major concern.
Chapter-4
Motion Detector Hardware and Software Development

4.1 Hardware development

The proposed system was composed of many different hardware and software elements. The hardware elements include optical fibers, reflective markers, webcam and lens.

![Hardware setup](image)

**Figure 4.1:** Hardware setup

The setup is shown in Figure 4.1. The reflective marker was fixed on the cylindrical phantom. The optical fiber end was fixed 8 mm away from the reflective marker on the phantom. The other end of the optical fiber bundle was connected to the webcam. The phantom is placed inside an RF coil. A brief description of the hardware components, along with their function in the proposed system is given below.

4.1.1 Fiber optics

The optical fiber cable used is an incoherent fiber optic bundle, ordered from Fiberoptics Systems, Inc. SimiValley, CA, USA. It is basically composed of borosilicate fibers arranged in
an incoherent fashion which transmits illumination. The wavelength for the borosilicate fiber is 400-1600 nm with an approximate consistent attenuation of .9dB/m for the 420-1600 nm range light.

**Figure 4.2**: The spectral attenuation plot for the borosilicate fiber (taken from the internet source: fiberopticsystems.com)

The length of the optical fiber cable is 11 m and the outer diameter is .635 cm. The rest of the specifications of the fiber optic cable are shown in Table 4.1.

**Table 4.1**: Specifications of the optical fiber cable

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical Aperture</td>
<td>0.55</td>
</tr>
<tr>
<td>Maximum Operating Temperature</td>
<td>176 F</td>
</tr>
<tr>
<td>Broken Fibers Allowed</td>
<td>5%</td>
</tr>
<tr>
<td>Fiber Count</td>
<td>340</td>
</tr>
<tr>
<td>Scale</td>
<td>1:1</td>
</tr>
<tr>
<td>Core to Cladding Ratio</td>
<td>44.6/50</td>
</tr>
<tr>
<td>Internal Transmission</td>
<td>18.3%</td>
</tr>
<tr>
<td>External Transmission</td>
<td>11.2%</td>
</tr>
</tbody>
</table>
The function of optical fibers in the system is to transfer the light reflected from the reflective marker to the other end connected to the webcam.

### 4.1.2 Webcam

The webcam used in the developed system is Logitech QuickCam Pro 9000. The webcam has an optical resolution of 2 MP and a focal length of 2 mm. The rest of the specifications are shown in Table 4.2.

![Logitech QuickCam Pro 9000](taken from the internet source: logitech.com)

**Figure 4.3:** Logitech QuickCam Pro 9000 (taken from the internet source: logitech.com)

**Table 4.2:** Specifications of the webcam

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB Type</td>
<td>High Speed USB 2.0, UVC</td>
</tr>
<tr>
<td>USB VID_PID</td>
<td>VID_046D&amp;PID_0990</td>
</tr>
<tr>
<td>Lens and Sensor Type</td>
<td>Glass, CMOS</td>
</tr>
<tr>
<td>Focus Type</td>
<td>Auto</td>
</tr>
<tr>
<td>Field of View (FOV)</td>
<td>75 degree Diagonal FOV</td>
</tr>
<tr>
<td>Frame Rate (max)</td>
<td>30 fps @ 800×600, 15 fps @ 960×720</td>
</tr>
</tbody>
</table>

It is connected to the output end of the optical fiber to capture the real time images of the reflective marker placed on the temple of the patient or phantom to track the movement. The images captured through the webcam were processed by the image acquisition toolbox in Matlab.
4.1.3 Reflective markers

The reflective markers used in the old system were bicycle reflectors. To construct markers for the new system, circles with diverse colors were drawn in paint and printed on the glossy paper. Some of the markers used in the testing are shown in Figure 4.4. Different color combination markers were used for testing and implementation. For testing, rings of colors of width 1 mm, 1.5 mm, 2 mm and 2.5 mm were used in the markers (except the central region which was black). The diameter of the central black region was chosen to be the same as the width of the rings of colors on the outside.

![Reflective markers](a) ![Reflective markers](b)

**Figure 4.4:** Reflective markers

The 71 markers used for testing are shown in Appendix H.

4.1.4 Converging lens

A converging lens is mainly used in high efficiency illumination applications. These lenses are designed for high light gathering efficiency in projecting, condensing, and other illumination applications. A 7 mm diameter converging lens was chosen for this project to fit perfectly on the end of the optical fiber used (Edmund Optics Inc., Barrington NJ, USA).

![Converging lens](taken from the internet source edmundoptics.com)

**Figure 4.5:** Converging lens (taken from the internet source edmundoptics.com)
The lens was attached to the optical fiber end above the reflective marker using an optical adhesive. The flatter side of the lens faces the fiber to distribute the light received from the reflector equally into all the fiber strands. The equal distribution of the light into every optical fiber strand will make it easier to analyze the color at the other end of the fiber. The experiments have been performed with and without the lens to observe the effects; the lens has not improved the performance of the system.

4.2 Software development

Software versions of the algorithms were created in MATLAB 2013a once the hardware design was complete. The algorithms work on the color images obtained through webcam and analyze the red, green and blue channels of the image to determine the range of movement. The reason for choosing MATLAB for the development of the software version of the algorithms is its image acquisition and image processing toolboxes which are suitable for image acquisition and processing in this research. The large database of the built in algorithms in the image processing toolbox facilitates easy implementation of ideas to operate on the images (represented as matrices in MATLAB) obtained through the webcam.

4.3 Analysis

The work for this thesis is based on the usage of algorithms developed to analyze the color or illumination of the color received at the end of the optical fiber connected to the webcam. Initially, the optical fiber end is exactly on top of the black color region of the marker. Movement in the patient will shift the position of the reflective marker under the optical fiber. The color under the fiber end after movement, if analyzed correctly in the webcam at the other end, will reflect the range of movement (width and position of each color is known).
Figure 4.6: Schematic of the hardware setup
An example is shown in Figure 4.7 to show how the color focused in the optical fiber will give information about the movement of the head.

Ideally, the color of the reflective marker on which the optical fiber is focused should be visible on the other end of the optical fiber and clearly seen in the webcam. Because the signal suffers degradation in the optical fiber due to various losses and dispersion, the algorithms were developed and implemented to circumvent this problem. The image processing toolbox of MATLAB is used to implement the algorithms and to process the images received from the webcam.
5.1 Design of the new system: simulation of motion effects

The effects of motion on the MRI images are distinctive. The defects vary depending on the amount of motion, type of motion, phase encoding steps and field of view. Simulation involved manipulating a dataset of MRI images by introducing motion to analyze the effects of motion on the images. Simulations were performed to observe the effects of each factor (the amount of motion, type of motion, phase encoding steps and the field of view) on the intensity of ghosting introduced into the images.

5.1.1 Overview

5.1.1.1 Image format

The dataset used for the simulation study was obtained from an online source [15]. The dataset of the MRI images used was in DICOM (Digital Imaging and Communications in Medicine) format. DICOM is a medical imaging standard that allows healthcare information to be exchanged, integrated, shared, and retrieved [16]. All the patient information is stored in the metadata in the DICOM images. The metadata information was imported along with the images when the MRI images were loaded into MATLAB software using routines in the image acquisition toolbox.

5.1.1.2 Planes of the section: axial, coronal, sagittal

MRI images can be acquired in three different orientations or planes: axial, coronal and sagittal.

Examples of images in these three different orientations of image views are shown in Figure 5.1 and Figure 5.2.
**Figure 5.1:** From top to bottom: sagittal, axial and coronal views of a three dimensional object [17]

**Figure 5.2:** MR Images of the brain in (a) axial plane (b) coronal plane (c) sagittal plane [17]
5.1.1.3 Type of rotations: pitch, roll, yaw

A rotation can be termed as a circular movement around one of the three rotation axes: x,y,z in the Cartesian coordinate system. People are lying supine (on their backs) in the MRI scanner.

1. **Pitch**: Pitch is rotation around the x-axis, up and down, like shaking of the head for a ‘Yes’.

2. **Roll**: Roll is rotation around the z-axis, like shaking of the head for a ‘No’.

3. **Yaw**: Yaw is the rotation of the head around the anterior posterior axis (y-axis), like shaking of the head for ‘Maybe’.

![Figure 5.3: x-y-z coordinate system](image_url)

**Figure 5.3**: x-y-z coordinate system

![Figure 5.4: Pitch, roll and yaw rotations of a head](image_url)

**Figure 5.4**: Pitch, roll and yaw rotations of a head (taken from the internet source: cnl.web.arizona.edu/imageprops)
5.1.1.4 Type of linear movements: head-foot, left-right

The type of linear movements possible during the MRI scan is along the x and y direction as shown in Figures 5.3 and 5.5.

**Figure 5.5:** MRI table in the xyz coordinate system (taken from the internet source: healthcare.siemens.com)

1. **Translation in z direction:** Translational movement in the head-foot direction (referring to Figures 5.3 and 5.5), like straightening of a body when lying on the MRI table.

2. **Translation in x direction:** Translational movement in the left-right direction (referring to Figures 5.3 and 5.5).
5.1.1.5 ImageJ

ImageJ is an open source program written in the Java programming language. It was created by NIH (National Institutes of Health) to perform powerful image analysis. It is very popular in the scientific community in the field of imaging as it is easily extensible due to its plugin-based design [18] [19].

![ImageJ window](image)

**Figure 5.6:** ImageJ window

It can perform operations such as edit, analyze and process on 8-bit, 16-bit and 32-bit images. It can read the images in standard formats such as JPEG, TIFF, GIF, and also in DICOM and “raw” format, the formats commonly used in the MRI field [18].

![Volume viewer plugin of the ImageJ](image)

**Figure 5.7:** Volume viewer plugin of the ImageJ
Volume viewer is a sub-plugin of plugin “3D” in ImageJ. Volume viewer makes it easy to perform scaling, translation and rotation on the stack of MRI images (datasets) imported into the ImageJ.

The reason to use ImageJ in the simulation is its Volume viewer plugin. Volume viewer allows the rotation of the image around all three rotation axes.

5.1.1.6 Testing systems

The test system I used for development, calibration and testing for the work outside the scanner is shown in Table 5.1. The test system was a HP laptop.

Table 5.1: Test system I

<table>
<thead>
<tr>
<th></th>
<th>Intel(R) Core(TM) i7-3610QM @ 2.30 GHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel(R) Core(TM) i7-3610QM @ 2.30 GHz</td>
</tr>
<tr>
<td>Operating System</td>
<td>Windows 7 Home Premium 64-bit</td>
</tr>
<tr>
<td>System Memory</td>
<td>8 GB</td>
</tr>
<tr>
<td>GPU</td>
<td>Intel(R) HD Graphics 4000</td>
</tr>
</tbody>
</table>

The test system II used for the calibration and testing for the work inside the scanner is shown in Table 5.2. The test system was a desktop computer in Hershey Medical Center.

Table 5.2: Test system II

<table>
<thead>
<tr>
<th></th>
<th>Intel(R) Core(TM) i5 @ 3.33 GHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel(R) Core(TM) i5 @ 3.33 GHz</td>
</tr>
<tr>
<td>Operating System</td>
<td>Windows 7 Professional</td>
</tr>
<tr>
<td>System Memory</td>
<td>4 GB</td>
</tr>
<tr>
<td>GPU</td>
<td>ATI Radeon HD 4550</td>
</tr>
</tbody>
</table>
5.1.1.7 Ghosting percentage

Ghosting percentage is the percentage of the mean signal of the ghost (without background) to the mean signal of the true signal (without background). The mean signal is first calculated in the pure ghost region and then the mean of the true signal is calculated in the same region without ghosts. The next step is to calculate the mean signal in the background, then calculate the ghosting percentage. The higher the ghosting percentage, the brighter the ghost would be in the image [20].

\[
\text{Ghosting percentage} = \frac{\text{mean (ghost)} - \text{mean (background)}}{\text{mean (true brain signal)} - \text{mean (background)}} \times 100 \tag{5.1}
\]

The ghosting percentage is measured using the algorithm in section 5.1.1.9. Calculations of the ghosting percentage are used to quantify the artifacts in the images.

5.1.1.8 Methodology for motion simulation

Coronal SPGR dataset of MRI images was used for the simulation [15].

<table>
<thead>
<tr>
<th>Table 5.3: The dataset information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Image format</strong></td>
</tr>
<tr>
<td>DICOM format</td>
</tr>
</tbody>
</table>

The starting point of the simulation was to import the dataset into an array. MATLAB was used for the simulation. The code is in Appendix B. The next step was to introduce motion. First, select the type of motion to be introduced. The types of motion available in this simulation are explained below.

1. **Translation in z direction**: Translational movement in the head-foot direction along the z-axis (referring to Figures 5.3 and 5.5), like the straightening of a body when lying on the MRI table.
2. **Translation in x direction**: Translational movement in the left-right direction along the x-axis (referring to Figures 5.3 and 5.5).

3. **Pitch**: Rotation around the x-axis (referring to Figure 5.4), like shaking of the head for a ‘Yes’, nodding movement.

4. **Yaw**: Rotation around the y-axis (referring to Figure 5.4), like shaking of the head for ‘Maybe’.

To decide the position and amount of motion to be introduced into the selected slice, input the parameters, repetition time (TR), time of motion and the amount of motion. Repetition time and time of motion will decide the position of motion.

\[
\text{Position of motion in y direction} = \text{round} \left( \frac{\text{Time of motion}}{\text{Repetition time}} \right)
\]

where repetition time is the amount of time taken to acquire one line in k-space.

The position of motion in the k-space matrix is the row in the k-space matrix which is being acquired when motion takes place. Since each row is acquired in a single time, TR, motion during that time is neglected in this simulation.

![Position of motion](image)

**Figure 5.8**: Position of motion in k-space matrix

2D Fourier Transform of the original image was performed. Once the k-space position (denoted as PES in algorithm in section 5.1.1.8.1 and code) and motion was decided, the original image was moved to the new position. 2D Fourier Transform of the image at the new position was performed.
In the MATLAB representation of the k-space after the Fourier Transform of an image, the k-space matrix has high frequencies in the central region. Shifting the quadrants of the FFT matrices of these two images will move the lower frequencies in the center. This is an important correction since that is how rectangular data are acquired in an MRI scanner. These are the k-space matrices of these two images. The section of the k-space past PES (the position of motion) in 2D Fourier Transform of the original image was substituted with 2D Fourier Transform of the image after performing the motion (the shift in head-foot or left-right direction or rotation).

A 2D inverse Fourier Transform of the combined k-space matrix obtained from substitution gave the final image after motion. For the example shown in Figure 5.9, the image at the new position was obtained by moving the original image in the head-foot direction by 10 mm.
5.1.1.8.1 Stepwise implementation of motion simulation

1. Establish the filename convention of .dcm used in the image series of DICOM images.

2. Examine one of the DICOM file headers to extract information such as position, sequence name, the field of view, phase encoding steps and matrix size.

3. Read the DICOM images and store in a 3D array named mri.

4. Read one of the images of array mri and store it in variable im.

5. Input the parameters repetition time (TR), time of motion (TM) and the amount of motion (AM) to decide the position and amount of motion to be introduced into the selected slice.

6. Create a menu to select the type of motion: translation in x direction (left-right movement), translation in y direction (head-foot movement), pitch and yaw. The amount of motion should be inputted in pixels for translations and in degrees for rotations.

7. Introduce a translation of AM pixels along the x direction in im, shift the image by AM pixels in upward direction and add AM rows of zeros at the end of the image im. Store this new image in variable im2.

8. Take Fourier Transform of im, im2 and save it in matrixes imifft, imifft2 respectively.

9. Combine the matrix imifft2 (rows from top to selected phase encoded line) and matrix imifft2 (rows from selected phase encoded line till bottom). Store the value of the combined matrix in a variable named combinedi2.

   Selected phase encoded line = floor(TM/TR).

10. Take Inverse Fourier Transform of combinedi2 to get the image after the head motion.
5.1.1.9 Methodology for measurement of ghosting percentage

The final step after the simulation was to calculate the ghosting percentage of images with motion to observe the effects of motion on the intensity of the ghost introduced into the original image.

![Block Diagram](image)

**Figure 5.10**: Block Diagram of the process to create the true signal and pure ghost regions

Using the original image and the image with motion, the true signal and ghost regions for the left and right sides were obtained. Steps to obtain the true signal and pure ghost regions are explained in detail in the next section. Mean values of the true signal and ghost region of the left and right sides were taken. Average of the mean values for the true signal and pure ghost regions were taken to evaluate the ghosting percentage. Ghosting percentage is the percentage of the mean signal of the ghost (without background) to the mean signal of the true signal (without background) [20].
5.1.1.9.1 Stepwise implementation of calculation of ghosting percentage

1. Take the original image im, shift it by N/2 pixels (where N is the size of the image) to the right to create a mask, mask1. Then, we shift the original image im to left by N/2 pixels to create a mask, mask2. Take the original image and keep the pixels in which mask1 and mask2 are zero to create a new mask called mask3.

2. To evaluate the left-ghost, mask3 is shifted by N/2 to the left to create mask4. The image obtained is the true signal region for the left side. To obtain the pure ghost region, take the ghosted image im2 and wherever the pixel values of mask4 is zero, make im2 also zero. This will result in a pure ghost region for the left side. Similarly do the same steps to get the pure ghost region for right side.

3. Take mean of the pure ghost region, the true signal region and background. The ratio of mean of ghosting (with mean of background subtracted) to mean of true signal (with mean of background subtracted) is ghosting ratio. The mean function used is shown below.

$$E[a] = \frac{1}{XY} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} a(x, y)$$

4. Repeat steps 2 to 3, to calculate the ghosting ratio or percentage for right side.

5. Take average of the ghosting ratio of left and right side.
5.1.1.10 Algorithm flowchart for motion simulation and measurement of ghosting percentage

![Algorithm flowchart](image)

**Figure 5.11:** Algorithm flowchart: Simulation of motion effects
5.2 Correlation method I – Correlation using the stationary image as the reference

In this method, the stationary image was set as the reference image. The stationary image is the image received in the webcam when the patient is stationary (referring to Figure 4.7a, the optical fiber is above the black center when the patient is stationary).

Images are acquired for different positions (colors) and the correlation was performed between these images and the reference image in red, green and blue planes.

![Figure 5.12: Block diagram of the construction of the correlation database](image)

The correlation coefficients corresponding to each position were stored in a database. Once the MRI scan started, the current image was correlated with the reference image, and then compared to all the correlation coefficients in the database. The hypothesis was that if the correlation coefficients of the current image match the correlation coefficients in the database, the motion will be known.

![Figure 5.13: Block diagram of the comparison phase of the correlation method I](image)
5.2.1 Stepwise implementation of correlation method I

1. The first step is to create the reference image.

2. Construct video input object ‘vid’ with resolution 640×480 and select the external webcam as the acquisition source.

3. Set the property values ‘region of interest’ and ‘colorspace’ for the object to 500×460 and rgb, respectively.

4. Preview the video to check if the video input is fine.

5. Place the optical fiber end over the central black color region. Capture the current image and store it as ‘reference.jpg.’

6. Once the reference image is formed, the next step is to create a database of correlation coefficients of each position with reference position.

7. Repeat steps 2 to 4.

8. Move the phantom to a different position. This will change the color of the reflective marker under the optical fiber end.

9. Capture the image of color for this position and correlate with the reference image in each of the three planes: red, green and blue.

10. Store the correlation coefficients for this position in the database.

11. Similarly perform steps 9 and 10 for all positions to calculate and store the correlation coefficients in the database.

12. The database created will be used in the comparison stage during the MRI scan.

13. Repeat steps 2 to 4.

14. Before starting the scan, make sure the optical fiber is above the reference position.

15. Start the comparison process along with the scan. Every 0.8 seconds, the current frame will be correlated with the reference image. The correlation coefficients of the current frame will then be compared to the correlation coefficients of all the positions in the database.
16. As soon as the match is found between the correlation coefficients of the current frame with the correlation coefficients in the database, motion will be known. The process will stop and a beeping sound will alert the technician to stop the scan. Also, a messagebox will open specifying the position of movement.

17. If there is no motion, the comparison process needs to be stopped when the MRI scan is finished.
5.2.2 Algorithm flowchart

Figure 5.14: Algorithm flowchart: Correlation method I
5.3 Correlation method II – Correlation using the database of images as the reference

The second correlation method was based on the correlation between the live current image and the database of images. Images were acquired for the different positions at the optical fiber end, connected to the webcam and stored in the database. The real time image of the marker from the webcam was correlated with the images in the database.

For the ideal case, the hypothesis was that if the correlation coefficients of the live image with one of the images in the database are equal to 1, motion will be known, but in a practical scenario it will be close to 1.

Figure 5.15: Block diagram of the working of correlation method II

Comparing two correlation methods, the main difference between the methods was the parameter in the database formation, and the comparison process. In the first method, the database of the correlation coefficients was created, and then the comparison was between the correlation coefficients of the current frame with the correlation coefficients in the database. The second method employed images directly and created the database of the images. The comparison process in the second method was done using the correlation between the current frame and the images in the database.
5.3.1 Stepwise implementation of correlation method II

1. The first step is to create the database of images.

2. Construct video input object ‘vid’ with resolution 640×480 and select the external webcam as the acquisition source.

3. Set the property values ‘region of interest’ and ‘colorspace’ for the object to 500×460 and rgb, respectively.

4. Preview the video to check if the video input is fine.

5. Move the phantom to each position except the stationary position (black color central region) and capture the image. Create a database of images in which each image corresponds to a specific position.

6. The database created will be used in the Comparison stage during the MRI scan.

7. Repeat steps 2 to 4.

8. Before starting the scan, make sure the optical fiber is above the stationary position.

9. Start the comparison process along with the scan. Every 0.8 seconds, the current frame will be correlated with the database of images.

10. Motion is known if the correlation between the current frame and one of the images in the database results in a correlation coefficient value greater than .98 in all three planes: red, green and blue. The process will stop and a beeping sound will alert the technician to stop the scan. Also, a messagebox will open, specifying the position of movement.

11. If there is no motion, the comparison process needs to be stopped when the MRI scan is finished.
5.3.2 Algorithm flowchart

Figure 5.16: Algorithm flowchart: Correlation method II
5.4 Relative luminance method

Luminance is a measure of the amount of light reflected from a particular area. The marker used for this method has bands of red and green colors of different luminosity in the head-foot and left-right direction, respectively. Red and green were chosen as they contribute most to the intensity in rgb images transmitted through the fiber optic bundles. The marker used for testing is shown in Figure 5.17.

![Figure 5.17: Marker used for testing in the relative luminance method](image)

The marker has the lowest luminance value at the center with black color. So, different versions of red and green colors are set in the marker such that luminance values increase as they move away from the center.

![Figure 5.18: Image diagram for the different versions of the red and green colors in the marker](image)

The mean values of the red and green channels corresponding to each position were stored in a database. The databases for the head-foot and left-right directions were different. With the increase in the luminance value, there was an increase in the mean value of each channel. So the
mean values of the three channels at a width of, for example, 5 mm from the center was more than the mean values of the channels at a width of 2.5 mm.

**Figure 5.19:** Block diagram of the working of the relative luminance method

Once the MRI scan starts, the mean values of the red and green channels of the current image were compared to each other. The hypothesis was that if the mean value of the red channel in the current image was greater than the mean value of the green channel, then the red channel mean value will be compared to the luminance values of the head-foot database. If there is a match, motion will be known.
5.4.1 Stepwise implementation of red-green relative luminance method

1. The first step is to create the database of images.

2. Construct video input object ‘vid’ with resolution 640×480 and select the external webcam as the acquisition source.

3. Set the property values ‘region of interest’ and ‘colorspace’ for the object to 500×460 and rgb respectively.

4. Preview the video to check if the video input is fine.

5. Move the phantom to each position except the stationary position (black color central region) and capture the image. Calculate the mean value of the red and green channels for each position and save the values in the database. Two different databases will be created for head-foot and left-right movements.

6. The databases created will be used in the comparison stage during the MRI scan.

7. Repeat steps 2 to 4.

8. Before starting the scan, make sure the optical fiber is above the stationary position.

9. Start the comparison process along with the scan. Every 0.8 seconds, the mean values of the red and green channels will be compared to each other. The greater value will then be compared with the values in the corresponding database.

10. Motion is known if the mean value of the red channel matches a value in the head-foot database, or a mean value of the green channel matches some value in the left-right database. The process will stop and a beeping sound will alert the technician to stop the scan. Also, a messagebox will open specifying the position of movement.

11. If there is no motion, the comparison process needs to be stopped when the MRI scan is finished.
5.4.2 Algorithm flowchart

Figure 5.20: Algorithm flowchart: Relative luminance method
Chapter-6
Methods: Simulation and Experimental Measurements for Motion Detection Techniques

6.1 Simulation of MRI image dataset

The coronal SPGR scan dataset has been used for simulation. Different types of motion as described in section 5.1.1.8 were introduced into images for the simulations. For linear movements, translation in z and x directions (refer to section 5.1.1.4), the motion introduced was varied from 1-5 pixels for the simulation. For rotational movements, pitch and yaw (refer to section 5.1.1.3), the motion introduced was varied from 1-5 degrees for the simulation. The time taken for introducing the motion depends on

1. Hardware specifications of the computer.

2. Size of the image.

The scan parameters of the MRI dataset used for the simulation were the matrix size: 256*256 and the number of slices: 60.

The motion was introduced in a single slice using the algorithm described in section 5.1.1.8. The image with the motion was the input to the ghosting percentage algorithm. Ghosting percentage steps described in section 5.1.1.9 were used to calculate the ghosting percentage.

6.2 Measurements using the correlation method I

6.2.1 Bench test experiments (outside the scanner)

A bench test is a set of experiments and measurements used to verify the working of the hardware before implementing it in the required workspace. The goal of bench testing here was to test the method before implementing it inside the MRI scanner. The motion monitor was tested on a phantom replica placed on a table inside a room with sufficient light to perform the experiments. The experiments were also performed with the converging lens attached at the optical fiber end over the reflective marker.

In the correlation method I experiments, two markers shown in Figure 6.1a and Figure 6.1b were used. The stationary images used as the reference in the database formation and comparison are
also shown in Figure 6.1. A region of interest of 500×460 was cropped to reduce the computational time.

![Diagram of reference images obtained in the webcam with the lens on and off for (a) Marker I and (b) Marker II.](image)

**Figure 6.1:** Reference images obtained in the webcam with the lens on and off for (a) Marker I (b) Marker II

The experiment involved moving the phantom from the 0 mm reference position to distances of 2 mm and 4 mm in upward and downward directions to emulate ‘straightening,’ a translational movement in and out of the bore. Patients often make this movement during MRI scanning to get comfortable.
6.3 Measurements using the correlation method II

6.3.1 Bench test experiments (outside the scanner)

In the correlation method II experiments, two markers shown in Figures 6.2 and 6.3 were used. The database of images used as the reference in the database formation and comparison are also shown in Figure 6.2 and Figure 6.3. A region of interest of 500×460 was cropped to reduce the computational time.

![Reference images obtained in the webcam with the lens on and off for Marker I](image)

**Figure 6.2:** Reference images obtained in the webcam with the lens on and off for Marker I
The experiment involved moving the phantom from the 0 mm reference position to distances of 2 mm and 4 mm in upward and downward directions to emulate ‘straightening,’ a translational movement in and out of the bore. Patients often make this movement during MRI scanning to get comfortable.

6.4 Measurements using the relative luminance method

6.4.1 Bench test experiments (outside the scanner)

The same hardware setup and environment was used to perform the bench testing for the relative luminance method. The calibrations were performed for the head-foot linear movement (straightening) and roll (rotational movement like a nod for saying ‘no’). The experiment involved moving the phantom from the 0 mm to distances of 2.5 mm, 5 mm, 7.5 mm, and 10 mm for the head-foot movement. For the roll movement, the phantom was moved away 1 degree, 2 degrees, 3 degrees, and 4 degrees in the experiment.
6.4.2 Experiments inside the scanner

For calibration inside the scanner, a cylindrical phantom (composed of materials that can be used in the MRI scanner) and an optical fiber bundle were positioned on the MRI table using a knee rest. The optical fiber end was fixed 8 mm away from the reflective marker on the phantom. A long wooden rod was fixed to the phantom which went directly through the middle of the phantom. The rod was used to move the phantom during the MRI. The Siemens 3.0T MRI scanner (Hershey Medical Center) was used for the calibration.

The measurements were performed for the head-foot movement and pitch while running the motion monitor. Moving the phantom in and out of the MRI table along the axis at distances of 2.5 mm and 5 mm simulated the head-foot movement. The movement made by a person while sneezing and coughing is similar to the pitch. Two movements, small and big, were made for simulating the pitch.

Some trials were also performed to acquire the MRI images with motion, to compare with the results of simulation. Trials involved performing the head-foot movement during the MRI scan at the middle of the k-space and at the edge of the k-space. For reference, motionless trials were also performed. For both the 2.5 mm and 5 mm movements, an 80 s axial TSE scan and a 77 s sagittal TSE scan were acquired. To introduce movement at the center of the k-space, the movement was performed at 40 seconds and 38 seconds into the scan for the axial and sagittal scans respectively. To introduce movement at the edge of the k-space, the movement was performed at 72 seconds and 70 seconds into the scan for the axial and sagittal scans respectively.
Chapter-7
Results: Observations and Measurements

7.1 Simulation of MRI image dataset

The slice number “30” of the dataset and the images with the left-right motion introduced at the middle and edge of the k-space are shown in Figure 7.1. As can be seen in the background of the image in Figure 7.1(b), the image with motion introduced at the middle of the k-space has been greatly corrupted, by motion.

![MRI images](image)

**Figure 7.1**: Coronal SPGR scan: MRI image of the brain (a) without motion (b) after introducing a motion of 4.25 mm in a right-to-left direction at the middle of the k-space (c) after introducing a motion of 4.25 mm in a right-to-left direction at the edge of the k-space
The table below shows, the ghosting percentage for the linear and rotational movements introduced into the middle of the k-space of slice “30” in the coronal SPGR dataset.

**Table 7.1: Simulation results of the tests for the coronal SPGR dataset**

<table>
<thead>
<tr>
<th>Resolution x<em>y (mm</em>mm)</th>
<th>Movement in pixels</th>
<th>Movement in mm</th>
<th>Direction of motion</th>
<th>Movement in degrees</th>
<th>Ghosting percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.85*0.85</td>
<td>1</td>
<td>0.85</td>
<td>Head-foot</td>
<td></td>
<td>0.44 %</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0.85</td>
<td>Right-left</td>
<td></td>
<td>0.87 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(pitch)</td>
<td>1</td>
<td>0.96 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(yaw)</td>
<td>1</td>
<td>0.7 %</td>
</tr>
<tr>
<td>0.85*0.85</td>
<td>2</td>
<td>1.7</td>
<td>Head-foot</td>
<td></td>
<td>0.74 %</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1.7</td>
<td>Right-left</td>
<td></td>
<td>1.91 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(pitch)</td>
<td>2</td>
<td>0.96 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(yaw)</td>
<td>2</td>
<td>1.29 %</td>
</tr>
<tr>
<td>0.85*0.85</td>
<td>3</td>
<td>2.55</td>
<td>Head-foot</td>
<td></td>
<td>1.09 %</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2.55</td>
<td>Right-left</td>
<td></td>
<td>2.79 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(pitch)</td>
<td>3</td>
<td>0.96 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(yaw)</td>
<td>3</td>
<td>1.82 %</td>
</tr>
<tr>
<td>0.85*0.85</td>
<td>4</td>
<td>3.4</td>
<td>Head-foot</td>
<td></td>
<td>1.36 %</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3.4</td>
<td>Right-left</td>
<td></td>
<td>3.72 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(pitch)</td>
<td>4</td>
<td>0.96 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(yaw)</td>
<td>4</td>
<td>2.31 %</td>
</tr>
<tr>
<td>0.85*0.85</td>
<td>5</td>
<td>4.25</td>
<td>Head-foot</td>
<td></td>
<td>1.64 %</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4.25</td>
<td>Right-left</td>
<td></td>
<td>4.77 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(pitch)</td>
<td>5</td>
<td>0.96 %</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rotation(yaw)</td>
<td>5</td>
<td>2.64 %</td>
</tr>
</tbody>
</table>
Figure 7.2: Coronal SPGR scan: Ghosting percentage for (a) linear movements: head-foot and left-right (b) rotational movements: yaw and pitch
As observed in Figure 7.2, the ghosting percentage is small but even this small ghosting percentage can affect the readability of the MRI images as it is a percentage of the signal level in the true image. Also, it can be observed that the left-right movement has the highest ghosting percentage. To visualize the change even this small percentage of ghosting can have on the image, line plots of the signal intensity for the center of slice “30” of the original image and the image with a 4.25 mm motion in the left-right direction introduced at the middle of the k-space have been shown below.

![Line plots of the center of slice number “30” in coronal SPGR dataset](image)

**Figure 7.3:** Line plots of the center of slice number “30” in the coronal SPGR dataset for the original image, and the image with a left-right movement of 4.25 mm introduced at the middle of the k-space

Figure 7.3 demonstrates that the mean of the signal intensities for the line profile of the image with motion is more than the original image. So, even a ghosting percentage as low as 5% can have a significant effect on the image. The image with motion shown in Figure 7.1b displays evidence of the effects of ghosting.
Figure 7.4: Line plots of the center of the slice number “30” in the coronal SPGR dataset for the original image, and the image with a left-right movement of 4.25 mm introduced at the edge of the k-space.

Figure 7.4 demonstrates that the line profile of an image with motion introduced at the edge of k-space is almost similar to the line profile of the original image except some small variations. Comparing the signal intensity of the line plots (Figures 7.3 and 7.4), the motion during the middle of the k-space affects the image more than the motion at the edge of the k-space. Figure 7.1(b) & (c) also display the evidence of increased ghosting with motion at the middle of the k-space than at the edge.

7.2 Measurement results using the correlation method I

7.2.1 Bench test experiments (outside the scanner)

Figures 7.5-7.8 were generated from the data obtained from the correlation of each position with the reference position for the markers I and II with the lens on and off.
7.2.1.1 Marker I with the lens on – correlation method I

Figure 7.5: Correlation method I: Correlation coefficients for the positions (colors) in the vertical direction in the marker I when the lens is on

<table>
<thead>
<tr>
<th></th>
<th>-4</th>
<th>-2</th>
<th>0</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc in red space</td>
<td>9.82E-01</td>
<td>9.78E-01</td>
<td>9.83E-01</td>
<td>9.61E-01</td>
<td>9.73E-01</td>
</tr>
<tr>
<td>cc in green space</td>
<td>9.85E-01</td>
<td>9.81E-01</td>
<td>9.86E-01</td>
<td>9.87E-01</td>
<td>9.85E-01</td>
</tr>
<tr>
<td>cc in blue space</td>
<td>9.01E-01</td>
<td>8.65E-01</td>
<td>8.39E-01</td>
<td>8.72E-01</td>
<td>8.96E-01</td>
</tr>
</tbody>
</table>

7.2.1.2 Marker I with the lens off – correlation method I

Figure 7.6: Correlation method I: Correlation coefficients for the positions (colors) in the vertical direction in the marker I when the lens is off

<table>
<thead>
<tr>
<th></th>
<th>-4</th>
<th>-2</th>
<th>0</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc in blue space</td>
<td>9.90E-01</td>
<td>9.91E-01</td>
<td>9.91E-01</td>
<td>9.89E-01</td>
<td>9.84E-01</td>
</tr>
</tbody>
</table>
7.2.1.3 Marker II with the lens on – correlation method I

**Figure 7.7:** Correlation method I: Correlation coefficients for the positions (colors) in the vertical direction in the marker II when the lens is on

### Table 7.1: Correlation method I

<table>
<thead>
<tr>
<th>mm</th>
<th>cc in red space</th>
<th>cc in green space</th>
<th>cc in blue space</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>9.48E-01</td>
<td>9.64E-01</td>
<td>9.73E-01</td>
</tr>
<tr>
<td>-2</td>
<td>9.94E-01</td>
<td>9.94E-01</td>
<td>9.68E-01</td>
</tr>
<tr>
<td>0</td>
<td>9.98E-01</td>
<td>9.99E-01</td>
<td>9.76E-01</td>
</tr>
<tr>
<td>2</td>
<td>9.90E-01</td>
<td>9.98E-01</td>
<td>9.76E-01</td>
</tr>
<tr>
<td>4</td>
<td>9.97E-01</td>
<td>9.85E-01</td>
<td>9.57E-01</td>
</tr>
</tbody>
</table>

7.2.1.4 Marker II with the lens off – correlation method I

**Figure 7.8:** Correlation method I: Correlation coefficients for the positions (colors) in the vertical direction in the marker II when the lens is off

### Table 7.2: Correlation method I

<table>
<thead>
<tr>
<th>mm</th>
<th>cc in red space</th>
<th>cc in green space</th>
<th>cc in blue space</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>9.82E-01</td>
<td>9.92E-01</td>
<td>9.80E-01</td>
</tr>
<tr>
<td>-2</td>
<td>9.89E-01</td>
<td>9.96E-01</td>
<td>9.81E-01</td>
</tr>
<tr>
<td>0</td>
<td>9.98E-01</td>
<td>9.99E-01</td>
<td>9.81E-01</td>
</tr>
<tr>
<td>2</td>
<td>9.89E-01</td>
<td>9.97E-01</td>
<td>9.80E-01</td>
</tr>
<tr>
<td>4</td>
<td>9.92E-01</td>
<td>9.92E-01</td>
<td>9.77E-01</td>
</tr>
</tbody>
</table>
Three trials were conducted to create the database of correlation coefficients for each position with the reference stationary image in the correlation method I using the two markers. The reproducibility was excellent and the maximum standard deviation observed between the data values was 0.01.

From the data in Figures 7.5-7.8, it was evident that the correlation coefficients obtained for each position are not as sensitive to the movement. The reason is the alignment of fibers in the bundle is not the same at both the ends. The marker image modifies by the time, it reaches the other end of the fiber. So, the image acquired by the webcam is a random orientation of the image from the input end of the fiber.

![Diagram](image)

**Figure 7.9:** The two ends of the optical fiber bundle with different orientation of fibers at each end

This method shows a change in the correlation coefficients for the movement, but it is small and not linear with the motion magnitude. So, this method was not pursued with the testing inside the MRI scanner.
7.3 Measurement results using the correlation method II

7.3.1 Bench test experiments (outside the scanner)

Figures 7.10-7.13 were generated from the data obtained from the correlation of each position with itself and the other three positions in the database for the marker I with the lens on.

The experiments were conducted for the marker I with the lens off, and marker II with the lens on and off as well. The results of these experiments have not been shown as the data obtained were similar to the data obtained for the marker I with the lens on.

The meaning of the positions I-IV used in the Figures 7.10-7.13 have been described in Table 7.2.

Table 7.2: The meaning of positions I-IV

<table>
<thead>
<tr>
<th>Position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position I</td>
<td>4 mm in upward direction from the centre</td>
</tr>
<tr>
<td>Position II</td>
<td>2 mm in upward direction from the centre</td>
</tr>
<tr>
<td>Position III</td>
<td>2 mm in downward direction from the centre</td>
</tr>
<tr>
<td>Position IV</td>
<td>4 mm in downward direction from the centre</td>
</tr>
</tbody>
</table>
7.3.1.1 Marker I with the lens on – correlation method II

\[ \text{c.c. in red, green and blue planes} \]

<table>
<thead>
<tr>
<th>c.c.</th>
<th>1.02E+00</th>
<th>1.00E+00</th>
<th>9.80E-01</th>
<th>9.60E-01</th>
<th>9.40E-01</th>
<th>9.20E-01</th>
<th>9.00E-01</th>
<th>8.80E-01</th>
<th>8.60E-01</th>
<th>8.40E-01</th>
<th>8.20E-01</th>
<th>8.00E-01</th>
</tr>
</thead>
</table>

**Figure 7.10:** Correlation method II: Correlation coefficients of position I with itself and the other three positions in the database for the marker I with the lens on

\[ \text{c.c. in red, green and blue planes} \]

<table>
<thead>
<tr>
<th>c.c.</th>
<th>1.00E+00</th>
<th>9.80E-01</th>
<th>9.60E-01</th>
<th>9.40E-01</th>
<th>9.20E-01</th>
<th>9.00E-01</th>
<th>8.80E-01</th>
<th>8.60E-01</th>
<th>8.40E-01</th>
<th>8.20E-01</th>
<th>8.00E-01</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2</td>
<td>9.73E-01</td>
<td>9.84E-01</td>
<td>9.85E-01</td>
<td>9.83E-01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8.71E-01</td>
<td>8.83E-01</td>
<td>9.00E-01</td>
<td>8.98E-01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>8.60E-01</td>
<td>8.80E-01</td>
<td>9.00E-01</td>
<td>9.20E-01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 7.11:** Correlation method II: Correlation coefficients of position II with itself and the other three positions in the database for the marker I with the lens on
**Figure 7.12:** Correlation method II: Correlation coefficients of position III with itself and the other three positions in the database for the marker I with the lens on.

**Figure 7.13:** Correlation method II: Correlation coefficients of position IV with itself and the other three positions in the database for the marker I with the lens on.
Three trials were conducted in this experiment for the marker I with the lens on. From the data shown in the Figures 7.10-7.13, it can be observed that not only the correlation coefficients of a position with itself are closer to 1.0 but also with the other three positions. Similar results were obtained for the maker I with the lens off, and marker II with the lens on and off.

So, the results from the experiments are not in good agreement with the results predicted and this method was not pursued with the testing inside the MRI scanner.

7.4 Measurement results using the relative luminance method

7.4.1 Bench test experiments (outside the scanner)

Graphs corresponding to the database values calibrated for the different positions by performing the head-foot linear movement and roll (left-right rotational movement, refer Figure 5.4) are shown in the Figures 7.14-7.17.
7.4.1.1 Marker with the lens on – relative luminance method

**Figure 7.14:** Marker with the lens on: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (bench tests)

**Figure 7.15:** Marker with the lens on: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (bench tests)
7.4.1.2 Marker with lens off – relative luminance method

**Figure 7.16:** Marker with the lens off: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (bench tests)

**Figure 7.17:** Marker with the lens off: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (bench tests)
Three trials were conducted to calibrate the mean values of the red, green and blue channels for each position. The values obtained for each channel in the case of lens on, are almost the same with high error bars. So, the testing was not performed for this case.

For the case of lens off, the mean values of the red channel showed a change in value of 25% to 116% when moved to positions 2.5 mm and 10 mm away from the center during the head-foot movement. During calibration of the left-right movement, the mean values of the red channel showed a change in value of 10% to 30% when moved to positions 1 degree and 4 degree away from the center. Since this case gave the expected results in calibration, this case was tested on the bench.

In testing, the motion monitor was moved randomly within the range of the head-foot and left-right directions. The motion monitor was able to detect the correct movement every time in 20 times, the testing was performed. The output of the motion monitor informs about the motion in real time in Matlab as shown in Figure 7.18. The motion monitor updated the position of the reflective marker every 0.8 seconds.

![Figure 7.18: Motion monitor output during testing](image)

As is evident from Figure 7.18, the testing was performed for a period of 20 seconds. The time duration of the testing can be inputted at the start of the testing. This method passes the benchmark testing, so it was pursued with the testing inside the MRI scanner.
7.4.2 Experiments inside the scanner

**Figure 7.19:** Marker with the lens off: Mean values of red, green and blue channels for different positions in head-foot movement for the relative luminance method (experiments inside the scanner)

**Figure 7.20:** Marker with the lens off: Mean values of red, green and blue channels for different positions in roll for the relative luminance method (experiments inside the scanner)
Mean values of the red channel showed a change in value of 90% to 420% when moved to positions 2.5 mm and 10 mm away from the center during the head-foot movement. During the calibration of pitch, the mean values of the red channel showed a change in value of 160% to 278% when moved to positions 1 degree and 4 degree away from the center. The change of mean value of the red channel is representative of the luminance of light reflected; the mean values of the red channel increase with the distance from the center.

The images with motion, acquired in the MRI scanner, were simulated to evaluate the amount of ghosting in them. Figures 7.21 and 7.22, shows the images with motion for the axial and sagittal scans. The results of the simulations are shown in Figures 7.23 and 7.24.
Figure 7.21: TSE scan (Axial): Images with motion for 2.5 mm and 5 mm linear head movement at the middle of the k-space and edge of the k-space

Figure 7.22: TSE scan (Sagittal): Images with motion for a 2.5 mm and 5 mm linear head movement at the middle of the k-space and edge of the k-space
Figures 7.23 and 7.24 demonstrate that even a small linear movement at the middle of the k-space will affect the image quality more than the larger movement at the edge of the k-space. Figures 7.21 and 7.22, also displays the evidence of an increased ghosting with motion at the middle of the k-space than at the edge.
Chapter-8
Discussion, Conclusion and Future Work

8.1 Discussion

Initially, the objective of the research was to improve the performance of the system developed by Wasil (discussed in section 3.1.2.1). On the way to improving the system by Wasil, a lot of changes to the hardware setup and the approach towards motion detection took place.

The earlier system used a detector and an analog to digital converter to convert the light received into voltage to be analyzed by the data acquisition toolbox in MATLAB. The problem with this system was there was no direct relationship between the signal received and the motion magnitude. The new system does not perform any conversion and grabs the light signal or image directly from the optical fiber bundle using the webcam. The image acquisition toolbox in MATLAB is used to analyze the images obtained by the webcam. This provides a more direct method of motion observation.

Table 8.1: Comparison of the properties of the Wasil’s system and developed system

<table>
<thead>
<tr>
<th>System Characteristics</th>
<th>Wasil’s System</th>
<th>Developed System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>Designed for analyzing voltage signal to detect motion</td>
<td>Designed for analyzing color images to detect motion</td>
</tr>
<tr>
<td>Software</td>
<td>Data acquisition toolbox in MATLAB and Instacal</td>
<td>Image acquisition toolbox in MATLAB</td>
</tr>
<tr>
<td>Hardware</td>
<td>Reflective marker, optical fibers, PDA520 detector, analog to digital converter</td>
<td>Reflective marker, optical fibers, webcam</td>
</tr>
<tr>
<td>Calibratable</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The other motion detection methods presented in section 3.1.2 have their own distinct advantages and disadvantages compared to each other and to the developed system. Table 8.2 summarizes these motion tracking methods and also the developed system versus the properties required in a motion detection method.
Table 8.2: Summary of the properties of the motion detection methods and developed method

<table>
<thead>
<tr>
<th>Motion Detection Method</th>
<th>Prospective</th>
<th>Scanner Dependent</th>
<th>Cost</th>
<th>Patient Comfort</th>
<th>Require clear line of sight from camera to marker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic scan plane tracking using MR position monitoring</td>
<td>Yes</td>
<td>Yes</td>
<td>Expensive</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Prospective real time motion correction using an external optical tracking system</td>
<td>Yes</td>
<td>Yes</td>
<td>Expensive</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Measurement and correction of microscopic head motion during magnetic resonance imaging of the brain</td>
<td>Yes</td>
<td>No</td>
<td>Expensive</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Developed Method</td>
<td>Yes</td>
<td>No</td>
<td>Inexpensive</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

8.2 Conclusion

The objective of the research was to develop a method based on low cost hardware namely optical fiber bundle, webcam and personal computer, which will be able to reliably detect the head movement in the patient during the MRI scan.

Three algorithms were explored for real time motion detection of the head movement in patients. The calibration results showed that the first two algorithms based on correlation seemed to have some shortcomings to be usable at this time. The reason, being the use of non-coherent optical fiber bundles in which there is no particular order of individual fibers. The third algorithm based on relative luminance gave significant results in the calibration and testing during the bench testing outside the scanner. It was able to detect the straightening and rotating motions of the head as small as 2.5 mm and 1 degree respectively. The calibrated results obtained inside the scanner also demonstrated that this method can detect movement and provide scanner independent information about the head movement in a patient.

Simulation was performed on an MRI dataset to compare the amount of artifacts introduced in the MRI images with the different amount of motion and position of motion introduced into the
k-space. Simulation results showed that the left-right movement introduced the maximum artifact in the image and if the motion occurred in the middle of the k-space, the artifact will be even more severe. TSE scans were acquired inside the MRI scanner for the different amounts of the linear head-foot movement. Analysis was also performed on the MRI images with motion obtained from the TSE scans. The results of the simulation and analysis were in good agreement and showed that the artifacts introduced in the MRI images are more severe if motion occurs in the middle of the k-space.

Other current motion detection methods are to a degree, scanner specific making it problematic for inclusion in every scanner. The proposed research will be a potential alternative, which will be scanner independent and based on inexpensive components.

### 8.3 Future research work

The research work in this thesis led to the development of a motion monitor that can detect movement as small as 2.5 mm and 1 degree in the head-foot and left-right directions. The motion detected can be fed back to the patient to inform about the motion. A feedback system can be developed to inform the patients of the movement which will help the patients to remain still during the MRI scan by informing them how well they are remaining still during the scan.

The effect of the change of the reflective marker to improve the system’s performance in MRI rooms with less room light can be researched in the future. 3D printer can be used to create a reflective marker with more luminance in colors. Bioluminescence ink can be used to create a marker which will be extremely bright and could work without any room light. A high resolution camera or a frame grabber could be used instead of the webcam to improve the quality of the images obtained in the PC. Therefore, the use of new and improved system components for a high performance motion detection system is envisaged in the future. Further testing in the MRI should be conducted.
Appendix

A. Background and theory of MRI (Magnetic Resonance Imaging)

A.1 History of MRI

MRI has a fascinating history. This brief section will serve as a quick introduction for the reader who does not have any previous experience in this field.

The first contribution to the field of MRI was by Joseph Fourier, a French mathematician and physicist. He developed an all-powerful tool called Fourier Transform which served as the basis for MRI. Fourier Transform takes a signal and expresses it in terms of frequencies of waves that forms that signal [2].

In 1895, a new era of medical imaging began with the invention of X-rays. This invention made scientists to start thinking about alternative technologies which indirectly led to MRI [6].

The discovery of the electron spin in 1924 was the first step towards MRI. After six years, Dirac predicted the property of spin angular momentum $\mathbf{J}$ in the quantum mechanical description of atomic nuclei. Later in 1938, an Austrian scientist named Isidor Isaac Rabi discovered the magnetic moment $\mathbf{\mu}$ [6].

In 1946, Felix Bloch and Edward Mills Purcell independently described Nuclear Magnetic Resonance (NMR) for which they both won the Nobel Physics Prize in 1952 [22] [23]. In late 1970s, Paul Lauterbur and Peter Mansfield independently acquired the first images of the brain using NMR signals which won them the Nobel Prize in physiology and medicine [24]. This imaging technique was based on something known as spatial encoding. Since then, MRI has experienced striking improvements in image quality and an explosion of MRI methods.

A.2 Physics of NMR

Atoms are made of electrons and nuclei with electrons rotating around the nucleus in orbits. Nuclei with an odd number of protons and neutrons have a net spin or angular moment. These protons have an electric charge and spin around the axes which create a magnetic field around them. This results in protons possessing magnetic moment along with angular moment. The angular moment $\mathbf{J}$ and magnetic moment $\mathbf{\mu}$ are related through a constant, called gyromagnetic ratio $\gamma$, as shown below.

$$\mathbf{\mu} = \gamma \mathbf{J}$$  \[A.1\]
The hydrogen nucleus has only one proton and thus exhibits a magnetic moment which is a vector quantity. Hydrogen atoms are found in abundance in the form of water and fat in human tissue which forms the basis for using NMR in medical MRI. In the absence of an externally applied magnetic field, the direction of the magnetic moments of spinning protons is randomly oriented. Thus the net sum of the magnetic moments referred to as the net magnetization denoted by $\mathbf{M}_0$, is practically zero. In the presence of an external magnetic field, $\mathbf{H}_0$, the magnetic moment vectors of hydrogen protons align themselves parallel or anti-parallel to the applied magnetic field [17][25].

![Diagram A.1](image)

**Figure A.1:** (a) Under no external magnetic field, proton spins are randomly oriented (b) Under influence of magnetic field; the spins attain one of the two orientations: parallel or anti-parallel [25].

Two energy states can be measured according to the interaction of the external magnetic field with spins in different orientations. The high energy state is when the proton spin is aligned anti-parallel to $\mathbf{H}_0$, while the low energy proton state is spin aligned parallel to the field. When thermal equilibrium is reached in the presence of external magnetic field, the number of protons aligned along the external magnetic is greater than the number of protons aligned against the external magnetic field. A net magnetization vector $\mathbf{M}$ will be created due to this imbalance in the direction of magnetic field. The transverse vector is still zero as the precession phase is still random [17].

### A.2.1 MR signal as a function of time

According to the classical description of a spin moving in a moving field, the rate of change of angular momentum is given by equation (A.2) [26] [27].
The solution to the above equation is the known as Larmor equation. This equation gives us the precession frequency for certain type of nuclei with a specific gyromagnetic ratio and the strength of external magnetic field.

\[ \omega = \gamma H_0 \]  

Classically, the addition of energy to protons causes the net magnetization vector to be pushed out of alignment with the external magnetic field direction. The net longitudinal magnetization vector is no longer in the direction of external magnetic field. The nuclei also start to precess together (in phase) resulting in a nonzero transverse vector that rotates in the x-y plane. The RF energy pulse required to flip the net longitudinal magnetization vector by 90-degree and 180-degree are known as the “90-degree pulse” and the “180 degree pulse” respectively.

After the RF pulse, the net longitudinal vector relaxes back to its original state by releasing energy and the net transverse magnetization vector dephases until it becomes zero. An electrical signal is induced in the RF coil during this relaxation phase at the Larmor frequency. The free induction delay (FID) of the electrical signal is the basic signal used in the creation of MR images.

Considering the nuclear relaxation phase, the rate of change of magnetization vector can be expressed as follows (Bloch’s equation):

\[ \frac{d\vec{M}}{dt} = \gamma \vec{M} \times \vec{H} - \frac{M_x I + M_y J}{T_2} - \frac{(M_z - M_z^0)k}{T_1} \]  

where $\vec{H}$ is the net magnetic field, $M_z^0$ is the net magnetization vector along longitudinal direction in thermal equilibrium state, and $T_1$ and $T_2$ are the longitudinal and transverse relaxation times respectively.

The transverse and longitudinal magnetization vector solutions to equation (A.4) can be obtained. The longitudinal magnetization solution equation (A.5) with respect to relaxation time $T_2$ is obtained through spin-lattice relaxation process with a decay rate of $1/T_1$ until thermal equilibrium position is obtained [1].

\[ M_z(t) = M_z^0 \left( 1 - e^{-t/T_1} \right) + M_z(0) e^{-t/T_2} \]  

[A.5]
where $M_z(0)$ is the longitudinal magnetization vector at time $t = 0$.

The transverse magnetization vector decays at a rate of $1/T_2$ until the transverse magnetization vector dephases completely and approaches zero.

$$M_{x,y}(t) = M_{x,y}(0) e^{-t/T_2} e^{-i\omega t} \quad [A.6]$$

where $M_{x,y}(0)$ is the longitudinal magnetization vector at time $t = 0$.

**Figure A.2:** (a) Transverse magnetization vector returning to zero value after RF pulse (b) Longitudinal relaxation returns to its original state after a 90-degree RF pulse (c) Longitudinal relaxation returns to its original state after a 180-degree pulse [17]
According to Faraday’s law, the voltage generated in the receiver coil, $V(t)$, is dependent on the rate of change of flux in the RF coil.

$$V(t) = -\frac{\partial \Phi}{\partial t}$$  \[A.7\]

If $\mathbf{a}(x,y,z)$ is the spatial location of a point in spinning nuclei system with a net magnetic field intensity $\mathbf{H}_a(x,y,z)$ and net magnetization $\mathbf{M}(x,y,z,t)$, it can be shown that the voltage induced in the RF coil is

$$V(t) = -\frac{\partial}{\partial t} \int_{\text{vol}} [\mathbf{H}_a(x,y,z) \cdot \mathbf{M}(x,y,z,t)] \, dv$$  \[A.8\]

The mapping of the MR response of the nuclei in a particular location requires a spatial location dependent signal to form an MR image. From equation (A.3), it is known that the Larmor frequency is related to the net magnetic field. If the external magnetic field is superimposed with an additional magnetic gradient field, the spatial locations can be excited using different Larmor frequencies. The different RF frequencies can be utilized to decode the different NMR signals generated by different spatial locations.

The NMR signals are mapped into images using different pulse sequences. The features associated with the image reconstructed are dependent on the pulse sequence used. Several version of the image can be reconstructed based on the weightings of the longitudinal relaxation time $T_1$, transverse relaxation time $T_2$ and spin density.

Considering that a time varying gradient field is applied, the equation can be expressed as

$$s(t) = \int \int M^0(x,y,z) \, e^{-i\omega_0 t} \, \exp \left( -i \gamma \int_0^t G(\tau) \cdot r \, d\tau \right) \, dx \, dy \, dz$$  \[A.9\]

where $G(\tau)$ is the time varying gradient vector and $r$ is the position vector.

For example, if a time varying gradient field is applied along the x-direction and y-direction,

$$s(t) = \int \int M^0(x,y,z) \, e^{-i\omega_0 t} \, \exp \left[ -i \gamma x \left( \int_0^t G_x(\tau) \, d\tau \right) \right] \exp \left[ -i \gamma y \left( \int_0^t G_y(\tau) \, d\tau \right) \right] \, dx \, dy$$  \[A.10\]

Suppose a plane centered at $z_0$ and of width $\Delta z$ is selected, so that signals from only that plane are considered. The signal will reduce to

$$s(t) = \int \int m(x,y) \, e^{-i\omega_0 t} \exp(i \gamma \int G(\tau) \cdot r \, d\tau) \, dx \, dy$$  \[A.11\]

where $m(x,y)$ is the 2D image to be reconstructed for a particular slice thickness.

The signal in equation (A.11) is demodulated in frequency using phase sensitive detection which results in a baseband signal

$$s(t) = \int \int m(x,y) \exp(i \gamma \int G(\tau) \cdot r \, d\tau) \, dx \, dy$$  \[A.12\]
The above equation simplifies further as the spatial localization is only in x and y directions, so only the x and y direction gradients will be considered [1].

\[
    s(t) = \int_x \int_y m(x, y)e^{-2\pi [k_x(t)x + k_y(t)y]} dx \, dy
\]  

[A.13]

where

\[
    k_x = \frac{\gamma}{2\pi} \int_0^t G_x(\tau) \, d\tau \quad \text{[A.14a]}
\]

\[
    k_y = \frac{\gamma}{2\pi} \int_0^t G_y(\tau) \, d\tau \quad \text{[A.14b]}
\]

Equation (A.13) is known as the signal equation. As is evident from equation (A.13), the signal \( s(t) \) is equal to the integral of the magnetization multiplied by a spatially dependent phase vector.

The 2D Fourier Transform of magnetization \( m(x,y) \) can be expressed as [1]

\[
    M(k_x, k_y) = \int_x \int_y m(x, y)e^{-i2\pi(k_x x + k_y y)} \, dx \, dy
\]  

[A.15]

Comparing the signal equation (A.13) with equation (A.14), the signal can be expressed as

\[
    s(t) = M(k_x(t), k_y(t))
\]  

[A.16]

or

\[
    s(t) = M\left(\frac{\gamma}{2\pi} \int_0^t G_x(\tau) \, d\tau, \frac{\gamma}{2\pi} \int_0^t G_y(\tau) \, d\tau\right)\]

[A.17]

\( k_x(t) \) and \( k_y(t) \) have units of spatial frequency, cycles/cm or cycles/mm. So, the signal \( s(t) \) equals the 2D Fourier Transform of \( m(x,y) \) at spatial frequencies \( k_x(t) \) and \( k_y(t) \).

Here, it becomes evident how important Fourier Transform mathematics were in the development of MRI.

**A.2.2 Signal contrast**

MRI is more flexible than x-ray CT as it has the ability to provide high contrast between soft tissue structures. The primary parameters responsible for the signal strength or the image contrast in the images are \( T_1 \), \( T_2 \) and \( \rho \) (spin density). The MR images acquired are weighted by these MR physical parameters. The physical parameters are highlighted by the values of imaging sequence parameters – TE, TR and flip angle. So, the MR images are a function of the physical and imaging sequence parameters. A brief summary of the MR physical parameters is given below [1].
Transverse magnetization \( M_{xy} \) has a significant property of decaying over time after its inception by a RF pulse.

There are precisely two types of decay which occur:

(a) Longitudinal relaxation time constant (spin-lattice) – \( T_1 \) decay
(b) Transverse relaxation time constant (spin-spin) - \( T_2 \) decay

The longitudinal relaxation, also known as spin-lattice relaxation process involves exchange of energy between the surroundings and the spin system to come back to equilibrium. This process is characterized by the spin-lattice or longitudinal relaxation time \( T_1 \) where \( T_1 \) is the time taken for the longitudinal magnetization \( M_z \) to return to its initial value of longitudinal magnetization \( M_0 \) after the operation of the RF pulse. The relaxation can be expressed as shown in equation (A.5) [6].

The exchange of energy not only takes place with the surrounding lattice, but also among the spins themselves. This process, known as transverse relaxation or spin-spin relaxation is characterized by the time constant \( T_2 \). The spin-spin relaxation time \( T_2 \) cites to the dephasing of the transverse magnetization \( M_{xy} \) due to the unavoidable thermal variations of the magnetic environment. This exponential decay of the transverse signal can be expressed as shown in equation (A.6) [6].

Spin density is the number of hydrogen protons per unit volume of tissue. When the weighting caused by the \( T_1 \) and \( T_2 \) relaxation times are low, the image contrast is primarily caused by the spin density [6].

\( T_1 \)-weighted scans generally use a pulse sequence with short repetition time (TR) and short echo time (TE). Keeping TR short, the longitudinal relaxation does not have enough time to recover 100 percent between the excitations. Keeping TE short makes sure that \( T_2 \) has little influence on the image contrast. An image weighted by \( T_1 \) is formed as the signal strength is governed by the \( T_1 \) of the tissue when the longitudinal relaxation does not fully recover [6].

\( T_2 \)-weighted scans normally use a pulse sequence with long repetition time (TR) and long echo time (TE). Keeping TR long ensures that \( T_1 \) does not influence image contrast as a long TR means all longitudinal magnetization recovers between the excitations. A long TE means that there is significant dephasing and the image contrast depends on the \( T_2 \) of the tissues [6].

Spin-density weighted images are generated using a pulse sequence with long repetition time (TR) and short echo time (TE). This results in spin density images with minimal weightings caused by \( T_1 \) and \( T_2 \) [6].

Contrast mechanisms are generally based on the differences in the \( T_1 \) and \( T_2 \) values for different tissues. Fat has a short \( T_1 \) and water has a long \( T_1 \), so longitudinal magnetization has time to
recover for fat and it appears bright in the final image but the effect of water on the final image is the opposite. Longitudinal magnetization does not recover fully for water if a short TR is used; it appears dark on image.

Figure A.3: T₁ weighted image: Water appears dark and fat appears bright (taken from the UC San Diego School of Medicine site)

As water has a long T₂, it will appear bright in the T₂ weighted image on the other hand fat, with a shorter T₂, appears dark.

Figure A.4: T₂ weighted image: Water appears bright and fat appears dark (taken from the UC San Diego School of Medicine site)
Table A.1 gives a summary of the relationship between the TE, TR and the image weighting with examples as shown below [6].

<table>
<thead>
<tr>
<th>Image Weighting</th>
<th>TR</th>
<th>TE</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>short</td>
<td>short</td>
<td>Fat</td>
</tr>
<tr>
<td>T2</td>
<td>long</td>
<td>Long</td>
<td>Water</td>
</tr>
</tbody>
</table>

Variable image contrast can be achieved using different pulse sequences, by changing the imaging sequence parameters and MR physical parameters. Every pulse sequence has specific timing of the RF and gradient pulses affecting the contrast on the MR image. This is explained in brief below using different pulse sequences.

### A.2.3 Echoes

There are some off-resonance sources that are responsible for phase accumulation. The phase accumulation can result from field inhomogeneity, chemical shift and gradient fields. Phase accumulation causes the signal amplitude to decay at a faster rate. The solution to this problem is the formation of echoes, where these phase shifts are removed thereby improving the strength of the signal. The two most common types of echoes are gradient echoes and spin echoes [1].

#### A.2.3.1 Gradient echoes

The gradient echoes are generated after gradient pulses are used for undoing of phase shifts introduced by gradient fields. In gradient echo, there is no 180 degree rephasing pulse and a bipolar readout gradient is used for refocusing the free induction delay (FID) [1].

![Gradient Echo Diagram](image)

**Figure A.5:** The bipolar gradient first dephases the spins and follows it up with rephasing to create the FID signal in the form of an echo [1].
From time $t=0$ to $\tau$, during the negative lobe stage of readout gradient, the gradient warps the spin phases, thus causing systematic dephasing. At time $t=\tau$, the gradient field reverses the direction. During the data acquisition or positive lobe stage, from time $t=\tau$ to $2\tau$, the phase warping gathered in negative lobe unwinds. At time $t = 2\tau$, the spins rephase completely, and the FID signal is obtained in the form of an echo [1] [28].

![Diagram of gradient echo](image)

**Figure A.6:** Phase plot of gradient echo: phase progression of spins at $x_1$, $-x_1$ and 0. From time 0 to $\tau$, the spins at $x_1$ gains phase and loses phase from $\tau$ to $2\tau$. The spins rephase at time $2\tau$, where the echo is formed [1].

$T_2^*$ relaxation is a combination of $T_2$ relaxation and magnetic field inhomogeneities. As the gradient echo method does not use a 180 degree rephasing pulse, it does not compensate for field inhomogeneities, therefore the transverse relaxation in gradient echo sequences is $T_2^*$. So, the signal obtained in gradient echo sequences are $T_2^*$ weighted rather than $T_2$ weighted [29].

### A.2.3.2 Spin echoes

Spin echoes are responsible for undoing of phase shifts introduced due to field inhomogeneities and chemical shift. A spin echo is the refocusing of spin magnetization by a series of RF pulses: 90° pulse and 180° pulse.
Looking into the spin diagram shown in Figure A.7(a), a 90 degree excitation pulse along x’ axis tilts the magnetization vectors from z to y’ axis. The spins experience phase dispersion and the NMR signal decays after the initial excitation pulse due to field inhomogeneity and chemical shift. The rate of phase accumulation due to inhomogeneities is constant as shown in Figure A.7(c). A 180 degree excitation pulse is applied to rotate about the spins x’ axis. At time $2\tau$, spin echo is formed and the spins are refocused [1] [28].

**Figure A.7:** Spin diagram showing the spin echo generation [1].

**Figure A.8:** Phase plot of spin echo. At time $\tau$, the 180 degree pulse reverses the phase by $\pi$. At time $2\tau$, the spins rephase [1].
Analyzing the phase plot (Figure A.8), it can be observed that, at time $\tau$, the 180 degree pulse reverses the phase by $\pi$. The phase accrual from time $\tau$ to $2\tau$, is the same as it did from time $t \to 0$ to $\tau$, and at time $t = 2\tau$, the spins rephase.

![Phase Plot Diagram](image)

**Figure A.9:** The signal decays for a period of $\tau$ after 90 degree RF pulse. 180 degree pulse reverses the dephasing effects to create the FID signal in the form of an echo at time $2\tau$ [1].

As is observed from Figure A.9, the 180 degree pulse rephases the spins and the NMR signal is obtained in the form of an echo. But the signal amplitude obtained at $2\tau$, is dependent on the intrinsic $T_2$ of the subject [1].

Comparing gradient echo (GE) and spin echo (SE), the most significant difference between the echoes is speed. Spin echo (SE) is a series of pulses while gradient echo (GE) only employs one RF pulse and does not have the 180 degree pulse, resulting in a shorter echo time. In gradient echo (GE), even low flip angles can be used, so, repetition time (TR) can also be shorter which makes gradient echo (GE), the best option for fast imaging. Another difference is that signal obtained in gradient echo (GE) is $T_2^\ast$ weighted while on the other hand the image is $T_2$ weighted in spin echo (SE) [30].

The MR signals can be acquired in the form of echoes without gradients but the information obtained only gives information about the material and not the location. Gradients are required to know the definite location of the source (explained in the next section). During the scanning process, if only an excitation pulse and data are acquired (without gradients); the acquisition of MR signals is independent of patient motion in the midst of scan. The combination of MR signals and spatial encoding (gradients) is termed as Magnetic Resonance Imaging.
A.3 Magnetic Resonance Imaging

A.3.1 Gradients to localize signals

In MRI, spatial encoding is the technique used to generate an image. The NMR signal, along with signals $T_2$ and $T_2^*$, can give the information about the material but it cannot give the definite location of the signal source. Spatial encoding was the missing piece to the puzzle of using NMR to form images.

Spatial encoding matches the received signal to the location of its origin by introducing gradients in the magnetic field. The spatial encoding is achieved by introducing three gradient coils inside the scanner each capable of generating a magnetic field along the z direction which can vary along any of the three direction x, y and z. The field gradients denoted by $G_x$, $G_y$ and $G_z$ are mathematically defined as:

\[
G_x = \frac{\partial B_z}{\partial x} \\
G_y = \frac{\partial B_z}{\partial y} \\
G_z = \frac{\partial B_z}{\partial z}
\]

The first step involved in spatial encoding is slice selection. This involves localizing or isolating to a two-dimensional slab of body in the object being imaged by only exciting the spins in that isolated single plane. This is accomplished by applying a RF pulse which only affects a specific part of the object being imaged in the presence of $G_z$ (linear field gradient along the direction along which the slice is to be selected) that adds to the static magnetic field $B_0$ as expressed [6]

\[
B_z(z) = B_0 + G_z z
\]  \[A.18\]

So a change in gradient will result in a change in the strength of the magnetic field. As given by the Larmor Equation, the precession frequency is directly proportional to the strength of the magnetic field. Therefore a variation in the Larmor frequency is created along the direction of the gradient. This results in excitation of only those spins which possess the same resonant frequency. If another slice needs to be selected, the RF pulse frequency is changed accordingly [6].
Figure A.10: Slice selection: Magnitude of the gradient field and bandwidth of the RF pulse determines the thickness of the slice. The excitations produced by the range of the magnetic field strengths are determined by the range of frequencies of the RF pulse as stated by the Larmor equation [6].

The thickness of the excited slice can be adjusted using the magnitude of the gradient field and bandwidth of the RF pulse as:

$$\Delta z = \frac{\Delta \omega}{\gamma G_z}$$  

[A.19]

where $G_z$ is the magnitude of the gradient field, $\gamma$ is the gyromagnetic ratio, $\Delta z$ is the thickness of slice and $\Delta \omega$ is the bandwidth of the RF pulse.

Since the gradient coils can be used to produce the linear field gradient in any of the three directions, the slice acquisition can be done in any direction.

A.3.2 Spatial resolution

The spatial resolution is defined as the minimum distance that can be seen distinctively in an image. Given Field of View (FOV) and number of sample points, spatial resolution can be defined in the x and y directions as [1]
where $N_{pe}$ is the number of phase encodes, $N_{read}$ is the number of readout samples, $\Delta k_x$ and $\Delta k_y$ are the sampling periods along x and y directions respectively. So, spatial resolution is dependent on the widths of k-space coverage in x and y directions [1].

Consider a 2DFT sequence with its k-space sampling as shown in Figure A.11. The sampling periods are $\Delta k_x$ and $\Delta k_y$ while $k_{x_{max}}$ and $k_{y_{max}}$ are the highest spatial frequencies sampled. It can be noticed from Figure A.11, that the widths traversed in k-space in x and y directions can be expressed [1]

$$W_{k_x} = 2 \left( k_{x_{max}} + \frac{\Delta k_x}{2} \right)$$  \hspace{1cm} [A.21a]

$$W_{k_y} = 2 \left( k_{y_{max}} + \frac{\Delta k_y}{2} \right)$$  \hspace{1cm} [A.21b]

**Figure A.11:** Sampling in k-space [1]
To proceed further, we refer to Figure A.12 which displays the timing and amplitude parameters for readout and phase encode gradients.

**Figure A.12:** Timing and amplitude parameters [1]

The k-space position depends on the amplitude of the gradients. So, the k-space sampling period in each direction will be dependent on the incremental gradient area as [1]

\[ \Delta k_x = \frac{v}{2\pi} G_{xr} \Delta t \]  \hspace{1cm} [A.22a]

\[ \Delta k_y = \frac{v}{2\pi} G_{yr} \Delta t \]  \hspace{1cm} [A.22b]

Also the highest spatial frequencies, \( k_{xmax} \) and \( k_{ymax} \), can be expressed as functions of maximum gradient areas attained by \( G_x \) and \( G_y \) (refer to Figure A.12).

\[ k_{xmax} = \frac{v}{2\pi} G_{xr} \frac{\tau_x}{2} \]  \hspace{1cm} [A.23a]

\[ k_{ymax} = \frac{v}{2\pi} G_{yp} \tau_y \]  \hspace{1cm} [A.23b]

where \( G_{xr} \frac{\tau_x}{2} \) and \( G_{yp} \tau_y \) are the maximum gradient area for the readout gradient \( (G_x) \) and phase encoding gradient \( (G_y) \) respectively [1].

Combining equation (A.20) with equations (A.21), (A.22) and (A.23), the spatial resolution can be expressed as [1]

\[ \delta_x = \frac{1}{W_{kx}} = \frac{1}{\frac{v}{2\pi} G_{xr}(\tau_x + \Delta t)} \]  \hspace{1cm} [A.24a]

\[ \delta_y = \frac{1}{W_{ky}} = \frac{1}{\frac{v}{2\pi} (2G_{yp} + G_{yr}) \tau_p} \]  \hspace{1cm} [A.24b]
For 3D imaging, the spatial resolution along the third dimension also needs to be expressed. The spatial resolution along z direction equals the slice thickness as expressed in equation (A.19).

### A.3.3 Signal-to-noise ratio

The Signal-to-Noise Ratio (SNR) is defined as the ratio of the power of original signal to power of noise. The original signal can be denoted by the mean of the signal and the noise can be considered by the variations in background as standard deviation. If the powers of signal and noise are represented as $P_{\text{sig}}$ and $P_{\text{noise}}$ respectively with $A_{\text{sig}}$ and $A_{\text{noise}}$ as amplitudes, then SNR can be expressed as [17]

$$\text{SNR} = \frac{P_{\text{sig}}}{P_{\text{noise}}} = \left(\frac{A_{\text{sig}}}{A_{\text{noise}}}\right)^2$$  \[A.25\]

$$\text{SNR}(\text{dB}) = 10 \log_{10}\left(\frac{P_{\text{sig}}}{P_{\text{noise}}}\right) = 20 \log_{10}\left(\frac{A_{\text{sig}}}{A_{\text{noise}}}\right)$$  \[A.26\]

If the noise in an image is not known, the SNR can be expressed as the ratio of the mean and standard deviation or root mean square of all gray level values of all pixels as shown below

$$\text{SNR} = \frac{\bar{g}}{\sqrt{\frac{1}{MN} \sum_{x=1}^{M} \sum_{y=1}^{N} (g(x,y) - \bar{g})^2}}$$  \[A.27\]

where $\bar{g} = \frac{1}{MN} \sum_{x=1}^{M} \sum_{y=1}^{N} g(x,y)$ is the mean.

We also need to take into account, the spatial resolution, before calculating the signal-to-noise ratio (SNR). Assuming the spatial resolution is represented by the three dimensions of a voxel, say, $\delta_x, \delta_y, \delta_z$, the SNR relation to spatial resolution can be expressed as [1]

$$\text{SNR} \propto (\delta_x)(\delta_y)(\delta_z)$$  \[A.28\]

### A.3.4 Contrast-to-noise ratio

In general, the Contrast-to-Noise Ratio (CNR) is a measure of image quality to distinguish the region of interest from its surroundings. In medical imaging, CNR is a measure of the capability to visualize and distinguish between the different regions (physiological structures, lesions) in the image [17].

Contrast in the image can be described as the difference in the gray level value of an object and its background. Contrast is normally normalized in 0-1 range as defined below
\[ C(x,y) = \frac{|f_{\text{obj}} - f_{\text{back}}|}{f_{\text{obj}} + f_{\text{back}}} \] \[ \text{[A.29]} \]

or

\[ C(x,y) = \frac{|f_{\text{obj}} - f_{\text{back}}|}{\max\{f_{\text{obj}}, f_{\text{back}}\}} \] \[ \text{[A.30]} \]

where \( f_{\text{obj}} \) and \( f_{\text{back}} \) are the average signal intensities or gray level values respectively in the object and background [17].

As explained in the previous section, standard deviation or root mean square of all gray level values of pixels can be expressed as noise. So, the CNR can be defined as

\[ \text{CNR} = \frac{|f_{\text{obj}} - f_{\text{obj}}^2|}{\sigma(x,y)} \] \[ \text{[A.31]} \]

where \( \sigma(x,y) \) is the standard deviation of the noise in the image [17].

### A.3.5 Pulse sequence

A pulse sequence is a series of RF pulses, gradient waveforms and data acquisition designed to acquire data for image formation. Each sequence use different RF pulses and gradient waveforms at different points of time with an aim to increase the speed or favor the signal of a particular tissue (contrast). The main purpose of pulse sequence is to manipulate the RF and gradient pulses to produce a desired signal [28] [31].

The three basic pulse sequences are: gradient echo, spin echo and inversion recovery. These basic pulse sequences can be further divided into other sequences based on many variations. Spoiled gradient echo, one of the variations of gradient echo is shown in Figure A.13.
A.3.5.1 Spin echo imaging

Spin echo is the most fundamental pulse sequence in MRI. A linear frequency gradient along the z direction is used for slice selection. Along with the selection gradient, a 90-degree pulse is applied for nuclear excitation. Spatial encoding gradients are applied in x and y directions. The phase encoding gradient along x direction is applied after the 90-degree pulse. After the 90-degree pulse, the net longitudinal magnetization vector starts returning to the thermal equilibrium state and transverse magnetization vector to zero. To rephase the spinning nuclei, a 180-degree pulse is applied along z direction before the nuclei lose their coherence. An echo is formed after the application of 180-degree pulse and the NMR signal is read through the readout gradient applied along the y direction. The time between the application of 90-degree pulse and the echo formation is the echo time, $T_E$, where the 180-degree pulse lies exactly in between the excitation pulse and echo [17].

Figure A.13: Spoiled GRE pulse sequence [31]
A.3.5.2 Inversion recovery imaging

The pulse sequences with an inversion pulse (180-degree) pulse before the RF excitation pulse are known as inversion recovery pulse sequences. The inversion recovery pulse sequence is composed of two parts. The first part consists of the inversion pulse and the second part is a self contained pulse sequence like EPI, RARE, spin-echo, or gradient echo. Normally, the first part is known as IR module and the second part as a host sequence [31].

The IR module can be combined with the host sequence in three ways. In some IR sequences, only one IR module is executed before a host sequence. In some cases, multiple IR modules are executed before a host sequence. In other cases, one IR module is followed with multiple host sequences such as in magnetic-prepared rapid gradient echo (MP-RAGE) [31].

Consider an inversion recovery pulse sequence with spin echo as the host pulse sequence. In spin echo imaging sequence, a frequency encoding gradient is applied for slice selection along with a 90-degree RF pulse for excitation. Prior to the host pulse sequence, a 180-degree pulse is applied along with frequency encoding gradient for slice selection. The inversion pulse allows the
relaxation of some or all of $T_1$ before the 90-degree pulse. The 90-degree pulse for excitation is applied after the $T_1$ time is recovered. The time to recover is known as inversion time [17].

Figure A.15: IR pulse sequence with spin echo as the host pulse sequence [17]

### A.3.5.3 Echo planar imaging

Echo Planar Imaging (EPI) is one of the fastest pulse sequences. In EPI, the frequency encoding gradient is applied for slice selection along with the 90-degree RF pulse for excitation. Multiple echoes are obtained using an oscillating gradient along the $x$-direction rather than application of single pulse gradient as is done in spin-echo pulse sequence. The phase inversion of the spinning nuclei is accomplished through the oscillating gradient resulting in periodic echoes. A readout gradient is applied along the $y$-direction to read the echoes. The entire image is obtained in a single shot [17].
Figure A.16: EPI pulse sequence [17]

A.3.5.4 Gradient echo imaging

Gradient Echo (GRE) is a pulse sequence that is mainly used in imaging that requires fast acquisition speed. GRE pulse sequences do not have the 180-degree refocusing pulse to form an echo. Instead, it uses an inverted readout gradient to rephrase the nuclei to form the gradient echo during the data acquisition period. The negative lobe first dephases the nuclei and then they are rephrased in the positive lobe of frequency gradient. Also, GRE pulse sequences are fast as low flip angle is used to affect the longitudinal magnetization component [17] [31].
B. Matlab code for simulation of motion effects and calculation of ghosting percentage

clc;
clear all;
close all;

% Establish filename convention used in image series
fileFolder = uigetdir(pwd,'Series 8')
files = dir(fullfile(fileFolder, '*.dcm'));
fileNames = {files.name}

%%% Obtain the scan information from metadata
info = dicominfo(fullfile(fileFolder, fileNames{20}))

%%% Obtain voxel size from the info
voxel_size = [info.PixelSpacing; info.SliceThickness]'

%%% Read one file to get size
I = dicomread(fullfile(fileFolder, fileNames{1}));
classI = class(I);
sizel = size(I);
umImages = length(fileNames);
%% Read slice images : populate 3-D matrix
hWaitBar = waitbar(0, 'Reading DICOM files');

% Create array
mri = zeros(sizeI(1), sizeI(2), numImages, classI);

for i=length(fileNames):-1:1
    frames = fullfile(fileFolder, fileNames{i});
    mri(:,:,i) = uint16(dicomread(frames));
    waitbar((length(fileNames)-i+1)/length(fileNames))
end

delete(hWaitBar)

%% Method
mri = flipdim(mri,3); % Make data face forwards
im = mri(:,:,30); % Pick middle slice for viewing (Original image)

max_level = double(max(im(:))); figure(1), imt = imshow(im, [0,max_level]); colorbar;

bcgd = im(:,:,1:30);
bcgdmean = mean2(bcgd); % Mean of background

%% Input parameters and perform simulation
prompt1 = 'Input the Repetition Time ';
TR = input(prompt1);
prompt2 = 'Input the Time Of Motion ';
TM = input(prompt2);
prompt3 = 'Input the Amount Of Motion ';
AM = input(prompt3);
ix = 256; % image size in x direction
iy = 256; % image size in y direction
x=0;
y=0;
PES = floor(TM/TR); % Phase encoding steps
TOS = menu('Choose the type of motion','Head-foot','Right-left','YAW','PITCH');

switch(TOS)
    case 1
    % Head-foot movement
    im2 = [im(AM+1:iy,:),zeros(AM,iy)]; % Shift and put AM rows equal to zero
end
case 2
% Right-Left movement
im2=[im(:,AM+1:ix) zeros(ix,AM)]; % Shift and put AM columns equal to zero
case 3
% Yaw (rotation around the z-axis)
im2=imrotate(double(im),AM,'nearest','crop'); % Rotate the image by AM degrees
case 4
% Pitch (rotation around the x-axis)
addpath('C:\Users\asw192\Desktop\Coronal SPGR- FOV220 & PES192');

% Images of first dataset rotated in ImageJ software
corimage(:,:,1)=imread('rotdegc1.jpg');
corimage(:,:,2)=imread('rotdegc2.jpg');
corimage(:,:,3)=imread('rotdegc3.jpg');
corimage(:,:,4)=imread('rotdegc4.jpg');
corimage(:,:,5)=imread('rotdegc5.jpg');
addpath('C:\Users\asw192\Desktop\FSE PD AXIAL OBL- FOV220 & PES256');

% Images of second dataset rotated in ImageJ software
fseimage(:,:,1)=imread('rotdegd1.jpg');
fseimage(:,:,2)=imread('rotdegd2.jpg');
fseimage(:,:,3)=imread('rotdegd3.jpg');
fseimage(:,:,4)=imread('rotdegd4.jpg');
fseimage(:,:,5)=imread('rotdegd5.jpg');
im2 = corimage(:,:,1); % Pick the image of dataset 1 with AM degrees of rotation
% im2 = fseimage(:,:,AM); % Pick the image of dataset 2 with AM degrees of rotation
otherwise
    warning('No motion');
end

shiftimifft = fftshift(fft2(double(im)));
shiftimifft2 = fftshift(fft2(double(im2)));

% Combined k matrix with k matrix 1 and k matrix 2
%combinedi2 = [shiftimifft2(:,1:(iy-PES)) shiftimifft(:,iy-PES+1:iy) ];
%column from bottom to top
%combinedi2 = [ shiftimifft(:,1:PES) shiftimifft2(:,(PES+1):iy)];
%column from top to bottom
%combinedi2 = [ shiftimifft2(1:(iy-PES),:); shiftimifft((iy-PES+1):iy,:)];
%row from bottom to top

combinedi2 = [shiftimifft(1:PES,:); shiftimifft2(PES+1:iy,:)];
%row from top to bottom

combinedimage2 = ifft2(ifftshift(combinedi2));
%MRI image of the combined k-space matrix

acombedimage2 = abs(combinedimage2);
max_level2 = double(max(acombinedimage2(:)));
figure(2), imt2 = imshow(acombinedimage2, [0,max_level2]); colorbar;

%Line plot
originalline = im(:,132);
motionline = acombinedimage2(:,132);

figure(40);
plot(0:255,originalline,0:255,motionline,0:255,mean(originalline),'r--',0:255,mean(motionline),'b--'); %Line plot
legend('Line plot-original image','Line plot-image with motion','Mean of line plot-original image','Mean of line plot-image with motion');

% Calculate ghosting percentage
% Step1 Creation of pure brain mask

mask2 = [im(:,(ix/2)+1:ix) zeros(ix,(ix/2))]; %left shift
max_level3 = double(max(mask2(:)));
figure(3), imt3 = imshow(mask2, [0,max_level3]); colorbar;

mask3 = [ zeros(ix,(ix/2)) im(:,1:(ix/2))]; %right shift
max_level4 = double(max(mask3(:)));
figure(4), imt4 = imshow(mask3, [0,max_level4]); colorbar;

amask2=abs(mask2);
amask3=abs(mask3);

for x=1:1:ix
    for y=1:1:iy
        if (amask2(x,y)==0)
            output1(x,y)=im(x,y);
        end
    end
end

for x=1:1:ix
    for y=1:1:iy
        if (amask3(x,y)==0)
            output2(x,y)=im(x,y);
        end
    end
end
for x=1:1:ix
    for y=1:1:iy
        if(output1(x,y)==output2(x,y))
            output3(x,y)=output1(x,y);
        end
    end
end

max_level5 = double(max(output1(:)));  
figure(5), imt5 = imshow(output1, [0,max_level5]);  colorbar;

max_level6 = double(max(output2(:)));  
figure(6), imt6 = imshow(output2, [0,max_level6]);  colorbar;

max_level7 = double(max(output3(:)));  
figure(7), imt7 = imshow(output3, [0,max_level7]);  colorbar;

% Step2 Creation of pure brain and ghost region masks for left and right regions using the brain mask created in step 1

mask4 = [output3(:,(ix/2)+1:ix) zeros(ix,(ix/2))];  %left shift
max_level8 = double(max(mask4(:)));  
figure(8), imt8 = imshow(mask4, [0,max_level8]);  colorbar;

mask5 = [ zeros(ix,(ix/2)) output3(:,1:(ix/2))];  %right shift
max_level9 = double(max(mask5(:)));  
figure(9), imt9 = imshow(mask5, [0,max_level9]);  colorbar;

for x=1:1:ix
    for y=1:1:iy
        if(mask4(x,y)~=0)
            mask6(x,y)=acombinedimage2(x,y);
        end
    end
end

for x=1:1:ix
    for y=1:1:iy
        if(mask5(x,y)~=0)
            mask7(x,y)=acombinedimage2(x,y);
        end
    end
end

max_level10 = double(max(mask6(:)));  
figure(10), imt10 = imshow(mask6, [0,max_level10]);  colorbar;

max_level11 = double(max(mask7(:)));  
figure(11), imt11 = imshow(mask7, [0,max_level11]);  colorbar;

% Step3 Applying the formula to calculate ghosting percentage
ablt4=abs(mask4); % left pure brain region
avg4=mean2(ablt4);

ablt5=abs(mask5); % right pure brain region
avg5=mean2(ablt5);

ablt6=abs(mask6); % left pure ghost region
avg6=mean2(ablt6);

ablt7=abs(mask7); % right pure ghost region
avg7=mean2(ablt7);

leftmeanp1= (avg6 - bcgdmean)/(avg4 - bcgdmean); % ghosting percentage in left region
leftmeanp2= (avg7 - bcgdmean)/(avg5 - bcgdmean); % ghosting percentage in right region
leftmeanp3= (leftmeanp1+leftmeanp2)/2 % ghosting percentage

C. Matlab code for creating the reference images in correlation methods I and II
clc;
clear all;
close all;

vid = videoinput('winvideo', 1,'YUY2_640x480');
set(vid, 'ReturnedColorSpace', 'RGB');
set(vid, 'ROIPosition', [60 0 500 460]);
preview(vid);

chq=menu('Capturing','Start Capturing','Exit');
if chq==1

img1 = getsnapshot(vid);
imwrite(img1, 'reference.jpg');
end

closepreview(vid);

D. Matlab code for calibration of correlation method I
clc;
clear all;
close all;
vid=videoinput('winvideo',1,'YUY2_640x480');
set(vid, 'ROIPosition', [60 0 500 460]);
set(vid,'TriggerRepeat',Inf);
set(vid,'ReturnedColorSpace','rgb');
preview(vid);

chc=menu('Correlation Data Analysis','Start Analysis','Exit');

%Set to 1 if the control enters first time in the while loop%
flag=1;

%Global Variable Declaration
 global CurrentImage StopProgram;
StopProgram=0;
closepreview(vid);
if chc==1
    if(StopProgram==0)
        %Take New Frames from the Video which is captured currently%
        RGBImage=getsnapshot(vid);
        RGBImage=getsnapshot(vid);

        %Show the captured Image%
        figure(1)
        subplot(3, 2, 1);
        imshow(RGBImage);
        title('Live Feed Image');

        CurrentImage=RGBImage;
        %Create time delay%
        delay = timer('TimerFcn',@(x,y)MotionAnalysis(CurrentImage),'StartDelay',0.2);
        start(delay);
        wait(delay);
        delete(delay);
    end
end

function MotionAnalysis(CurrentImage)

    %Comparision Images%
    CompareImage(:,:,1)=imread('reference.jpg');

    %Initialisation%
    Match(3)=0;
    global StopProgram;
    StopProgram = 0;
    %RGB Images%
    figure(1)
    subplot(3, 2, 2);
    imshow(CurrentImage(:,:,1));
    title('R Image');
figure(1)
subplot(3, 2, 3);
imshow(CurrentImage(:,:,2));
title('G Image');

figure(1)
subplot(3, 2, 4);
imshow(CurrentImage(:,:,3));
title('B Image');

if(StopProgram==0)
    for j=1:1:3
        Match(j) = corr2(CompareImage(:,:,j,1),CurrentImage(:,:,j));
    end
end
fprintf('
The Correlation value in r space =%d',Match(1));
fprintf('
The Correlation value in g space =%d',Match(2));
fprintf('
The Correlation value in b space =%d',Match(3));
end

E. Matlab code for calibration of correlation method II

clc;
clear all;
close all;

%Current Image
vid=videoinput('winvideo',1,'YUY2_640x480');
set(vid,'TriggerRepeat',Inf);
set(vid,'ReturnedColorSpace','rgb');
set(vid, 'ROIPosition', [60 0 500 460]);
preview(vid);

chc=menu('Motion Detection Analysis','Start Motion Detection','Exit');

%Set to 1 if the control enters first time in the while loop%
flag=1;

%Global Variable Declaration
global CurrentImage StopProgram;
StopProgram=0;
closepreview(vid);
if chc==1
    while(StopProgram==0)
        %Take New Frames from the Video which is captured currently%
        RGBImage=getsnapshot(vid);
        RGBImage=getsnapshot(vid);

        %Show the captured Image%
        figure(1)
        subplot(3, 2, 1);
imshow(RGBImage);
title('Live Feed Image');

CurrentImage=RGBImage;
%create time delay%
delay = timer('TimerFcn', @(x,y)MotionAnalysis(CurrentImage), 'StartDelay', 0.2);
start(delay);
wait(delay);
delete(delay);
end
end

function MotionAnalysis(CurrentImage)

%Comparision Images%
CompareImage(:, :, :, 1) = imread('north1.jpg');
CompareImage(:, :, :, 2) = imread('north2.jpg');
CompareImage(:, :, :, 3) = imread('south1.jpg');
CompareImage(:, :, :, 4) = imread('south2.jpg');

%Initialisation%
Match(3) = 0;
global StopProgram;
StopProgram = 0;

%RGB Images%
figure(1)
subplot(3, 2, 2);
imshow(CurrentImage(:,:,1));
title('R Image');

figure(1)
subplot(3, 2, 3);
imshow(CurrentImage(:,:,2));
title('G Image');

figure(1)
subplot(3, 2, 4);
imshow(CurrentImage(:,:,3));
title('B Image');

for i=1:1:4
    for j=1:1:3
        Match(j) = corr2(CompareImage(:, :, j, i), CurrentImage(:, :, j));
    end
    fprintf('\nThe Correlation value in r space =\d', round(Match(1)*100)/100);
    fprintf('\nThe Correlation value in g space =\d', round(Match(2)*100)/100);
    fprintf('\nThe Correlation value in b space =\d', round(Match(3)*100)/100);
end
F. Matlab code for calibration of relative luminance method

```matlab
clc;
clear all;
close all;

vid = videoinput('winvideo', 1,'RGB24_640x480');
set(vid, 'ReturnedColorSpace', 'RGB');
set(vid, 'ROIPosition', [60 0 500 460]);
%set(vid, 'ROIPosition', [160 60 320 320]);
preview(vid);

chq=menu('Capturing','Start Capturing','Exit');
if chq==1
    img1 = getsnapshot(vid);
imwrite(img1,'0 mm up-down.jpg');
end
closepreview(vid);

im=imread('0 mm up-down.jpg');

y1=im(:,:,1);
y2=mean2(y1)
y3=im(:,:,2);
y4=mean2(y3)
y5=im(:,:,3);
y6=mean2(y5)
```

G. Matlab code for testing of relative luminance method

```matlab
clc;
clear all;
close all;

%Current Image
vid=videoinput('winvideo',1,'RGB24_640x480');
set(vid, 'ROIPosition', [60 0 500 460]);
set(vid, 'TriggerRepeat',Inf);
```
set(vid,'ReturnedColorSpace','rgb');
preview(vid);

chc=menu('Motion Detection Analysis','Start Motion Detection','Exit');

%Set to 1 if the control enters first time in the while loop
flag=1;

%Global Variable Declaration
global CurrentImage timeduration;
closepreview(vid);

duration = inputdlg('Enter the duration of the scan:',...
  'INPUT TIME', [1 40]);

timeduration = str2num(duration{1});  %#ok<ST2NM>

% Initialization of head-foot and head turning movements

time=0;
data1=0;
data2=0;

% Head-foot movement
figure(2)
subplot(2,1,1);
plot(time,data1,:r*');
axis([0 (timeduration+2) -2 14])
xlabel('Time (s)'); % Setting up the xlabel for data1
ylabel('Motion in mm'); % Setting up the ylabel for data1
title('Head-foot movement'); % Setting up the title for data1
hold on;

% Head turning movement
figure(2)
subplot(2,1,2);
plot(time,data2,:r*');
axis([0 (timeduration+2) -2 6])
xlabel('Time (s)'); % Setting up the xlabel for data2
ylabel('Motion in degrees'); % Setting up the ylabel for data2
title('Head Turning movement'); % Setting up the title for data2
hold on;

if chc==1
  tic
while toc<=timeduration
    %Take New Frames from the Video Which is captured currently%
    RGBImage=getsnapshot(vid);

    %Show the captured Image%
    figure(1)
    subplot(3, 2, 1);
    imshow(RGBImage);

    %Update the figure with the new image
    updatefig(1)

    %Calculate the mean and standard deviation of the captured image
    mean_val=mean2(RGBImage);
    std_val=std2(RGBImage);

    %If the mean value is greater than the threshold, set the flag to 1
    if mean_val > threshold
        flag=1;
    end

    %If the flag is 1, update the current image
    if flag==1
        CurrentImage=RGBImage;
        flag=0;
    end

  end
end
function MotionAnalysis_marker1(CurrentImage)

%Initialisation%

global timeduration;

%RGB Images%
figure(1)
subplot(3, 2, 2);
imshow(CurrentImage(:,:,1));
title('R Image');

figure(1)
subplot(3, 2, 3);
imshow(CurrentImage(:,:,2));
title('G Image');

figure(1)
subplot(3, 2, 4);
imshow(CurrentImage(:,:,3));
title('B Image');

y1=CurrentImage(:,:,1);
y2=mean2(y1);
y3=CurrentImage(:,:,2);
y4=mean2(y3);
y5=CurrentImage(:,:,3);
y6=mean2(y5);

fprintf('
The mean value of red channel = %d, green channel = %d, blue channel = %d',y2,y4,y6);

if(y2>y4)
    if((61.9<y2) && (y2<80.6))
        data1 = 2.5;
        data2 = 0;
        time = toc
        beep
    elseif((80.6<y2) && (y2<94.5))
        data1 = 5;
    elseif((94.5<y2))
        data1 = 6;
    elseif((y2<61.9))
        data1 = 1;
    endif
endif
data2 = 0;
time = toc
beep

elseif((94.5<y2) && (y2<106.8))
data1 = 7.5;
data2 = 0;
time = toc
beep

elseif(106.8<y2)
data1 = 10;
data2 = 0;
time = toc
beep
endif(y2<61.9)
data1 = 0;
data2 = 0;
time = toc
beep;
end

elseif(y4>y2-1.5)
if((12.1<y6) && (y6<13.6))
data2 = 1;
data1 = 0;
time = toc
beep

elseif((10.2<y6) && (y6<12.1))
data2 = 2;
data1 = 0;
time = toc
beep
endif((8<y6) && (y6<10.2))
data2 = 3;
data1 = 0;
time = toc
beep
endif(y6<8)
data2 = 4;
data1 = 0;
time = toc
beep
elseif(y6>12.1)
data2 = 0;
data1 = 0;
time = toc
beep;
end

end %elseif loop ends here
\begin{verbatim}
% Head-foot movement
figure(2)
subplot(2,1,1);
plot(time,data1,'r*');
axis([0 (timeduration+2) -2 14])
xlabel('Time (s)'); % Setting up the xlabel for data1
ylabel('Motion in mm'); % Setting up the ylabel for data1
title('Head-foot movement'); % Setting up the title for data1
hold on;

% Head turning movement
figure(2)
subplot(2,1,2);
plot(time,data2,'r*');
axis([0 (timeduration+2) -2 6])
xlabel('Time (s)'); % Setting up the xlabel for data2
ylabel('Motion in degrees'); % Setting up the ylabel for data2
title('Head Turning movement'); % Setting up the title for data2
hold on;
end

H. Markers used for testing
\end{verbatim}
References


