MAGNETIC RESONANCE IMAGING SENSE RECONSTRUCTION SYSTEM USING FPGA

MUHAMMAD FAISAL SIDDIQUI

FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

MAGNETIC RESONANCE IMAGING SENSE RECONSTRUCTION SYSTEM USING FPGA

MUHAMMAD FAISAL SIDDIQUI

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **MUHAMMAD FAISAL SIDDIQUI**

Registration/Matric No: **KHA130057**

Name of Degree: **DOCTOR OF PHILOSOPHY**

Title of Thesis: **MAGNETIC RESONANCE IMAGING SENSE RECONSTRUCTION SYSTEM USING FPGA**

Field of Study: **EMBEDDED SYSTEMS**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

ABSTRACT

Parallel imaging is a robust method for accelerating the data acquisition in Magnetic Resonance Imaging (MRI). Under-sampled data is acquired in parallel imaging to expedite the MRI scan process, which leads to aliased images. Sensitivity Encoding (SENSE) is a widely used technique to reconstruct the artefact free images from the Parallel MRI (pMRI) aliased data. Reconfigurable hardware based architecture for SENSE has a great potential to provide good quality image reconstruction with significantly less computation time. This thesis aimed to investigate and develop a novel parameterized architecture design for SENSE algorithm. The proposed design is implemented on Field Programmable Gate Arrays (FPGAs) platform, which can provide real-time SENSE reconstruction right on the receiver coil data acquisition system with no need to transfer data to the MRI server. Complex multiplier, complex matrix multiplier and pseudo-inverse modules are designed according to the algorithmic needs to increase the efficiency of the system. Furthermore, variable databus widths are used in the data path of the proposed architecture, which leads to reducing the hardware cost and silicon area. The use of eigenvectors decomposition (E-maps) and pre-scan methods for estimating sensitivity maps are also investigated. The reconstruction results are compared with the multi-core CPU and Graphical Processing Unit (GPU) based reconstructions of SENSE. This research also proposed an intelligent and robust classification technique to classify the MRI scans as normal or abnormal and also for validation purpose. The proposed classifier has been developed by using fast Discrete Wavelet Transform (DWT), Principal Component Analysis (PCA) and Least Squares Support Vector Machine (LS-SVM). Firstly, fast DWT is employed to extract the salient features of MRI images, followed by PCA, which reduces the dimensions of the features. Finally, LS-SVM is applied to MR image classification using reduced features.

The achieved reconstruction results are 850 times faster than the conventional multi-core CPU and 85 times faster than the GPU based reconstructions of SENSE, while maintaining the quality of the reconstructed images with significantly less artefact power ($<2.45\times10^{-4}$) and good mean SNR (35+ dB) values. The proposed system also provides better reconstruction results when using E-maps and achieves $<9\times10^{-4}$ and 29+ dB for artefact power and mean SNR, respectively. Center line profiles comparison also demonstrates the quality of the reconstructed images. The proposed system offers a reconstruction right on the multi-channel data acquisition module which minimizes the transmission cost and memory usage of the MRI system. Furthermore, its low power consumption features can be remarkable especially for portable MRI scanners. Moreover, the proposed classifier technique is significantly faster than the recent well-known methods, and it improves the efficiency by 71%, 3%, and 4% on feature extraction stage, feature reduction stage, and classification stage, respectively. The results indicate that the overall system is capable of reconstructing the high quality images from the pMRI aliased data in real-time and then classify it as normal or abnormal, therefore, it can be used as a significant tool in clinical practice.

ABSTRAK

Parallel imaging adalah satu kaedah yang mantap untuk mempercepatkan perolehan data dalam Magnetic Resonance Imaging (MRI). Data under-sampel diperoleh dalam pengimejan selari dengan mempercepatkan proses imbasan MRI, yang membawa kepada menjana imej alias. Sensitivity Encoding (SENSE) adalah teknik yang digunakan secara meluas untuk membina semula artifak imej percuma dari Parallel MRI (pMRI) data alias. Perkakasan pembentukan semula seni bina berasaskan untuk SENSE mempunyai potensi yang besar untuk menyediakan kualiti yang baik pembinaan semula imej dengan ketara kurang masa pengiraan. Tesis ini bertujuan untuk menyiasat dan membangunkan reka bentuk seni bina parameterized novel untuk algoritma SENSE. Reka bentuk yang dicadangkan dilaksanakan pada Field Programmable Gate Arrays platform (FPGAs), yang boleh memberikan masa nyata SENSE pembinaan semula hak ke atas sistem perolehan data penerima gegelung tanpa keperluan untuk memindahkan data kepada pelayan MRI. Pengganda Complex, matriks pengganda kompleks dan pseudo-songsang modul direka mengikut keperluan algoritma untuk meningkatkan kecekapan sistem. Tambahan pula, pembolehubah lebar bas data digunakan dalam laluan data seni bina yang dicadangkan itu, yang membawa kepada mengurangkan kos perkakasan dan kawasan silikon. Penggunaan vektor eigen penguraian (E-peta) dan kaedah pra-scan untuk menganggarkan peta sensitiviti juga disiasat. Hasil pembinaan semula dibandingkan dengan CPU berbilang teras dan pembinaan semula Unit Pemprosesan Grafik (GPU) berasaskan SENSE. Kajian ini juga mencadangkan teknik pengelasan pintar dan mantap untuk mengklasifikasikan imbasan MRI seperti biasa atau tidak normal untuk tujuan pengesahan. Pengelas yang dicadangkan itu telah dibangunkan dengan menggunakan fast Discrete Wavelet Transform (DWT), Principal Component Analysis (PCA) dan Squares Kurang Sokongan Mesin Vector (LS-SVM).

Pertama, DWT segera diambil kerja untuk mendapatkan ciri-ciri utama imej MRI, diikuti oleh PCA, yang mengurangkan dimensi ciri-ciri. Akhirnya, LS-SVM digunakan untuk pengelasan imej MRI menggunakan ciri-ciri dikurangkan. Hasil pembinaan semula yang dicapai adalah 850 kali lebih cepat daripada CPU berbilang teras konvensional dan 85 kali lebih cepat daripada pembentukan semula GPU berasaskan SENSE, di samping mengekalkan kualiti imej yang dibina semula dengan kuasa yang kurang artifak ($<2.45\times10^{-4}$) dan min SNR baik (35+ dB) nilai-nilai. Sistem yang dicadangkan juga memberikan hasil pembinaan semula lebih baik apabila menggunakan E-peta dan mencapai $<9\times10^{-4}$ dan 29+ dB untuk kuasa artifak dan min SNR, masing-masing. garis tengah profil perbandingan juga menunjukkan kualiti imej yang dibina semula. Sistem yang dicadangkan menawarkan hak pembinaan semula pada modul pemerolehan data pelbagai saluran yang mengurangkan penggunaan kos penghantaran dan memori sistem MRI. Tambahan pula, ciri-ciri penggunaan kuasa yang rendah boleh menjadi luar biasa terutamanya untuk pengimbas MRI mudah alih. Selain itu, teknik pengelas yang dicadangkan itu adalah ketara lebih cepat daripada kaedah baru-baru ini yang terkenal, dan ia meningkatkan kecekapan sebanyak 71%, 3%, dan 4% di atas pentas ciri pengekstrakan, peringkat pengurangan ciri, dan peringkat klasifikasi, masing-masing. Keputusan menunjukkan bahawa keseluruhan sistem mampu membina semula imej yang berkualiti tinggi daripada data PMRI dialiaskan dalam masa nyata dan kemudian mengklasifikasikan ia sebagai normal atau tidak normal, oleh itu, ia boleh digunakan sebagai alat yang penting dalam amalan klinikal.

ACKNOWLEDGEMENTS

I am thankful to ALMIGHTY ALLAH for helping me and giving me the courage to complete my PhD.

I would like to thank all the following people who have been pivotal in my PhD:

Dr. Ahmed Wasif Reza, my PhD supervisor, for his expert guidance and support. He is the kindest and most gentle person I know. He had been a constant source of motivation and encouragement throughout my PhD. Dr. Wasif has given me the tools, the means, the guidance, and the freedom to pursue my research directions. I feel deeply indebted for his all-out help. Dr. Wasif, I really appreciate your excellent supervision and mentoring during my PhD!

Dr. Hammad Omer, that's where my PhD starts, Dr. Hammad gave me this opportunity to work with his research group. His excellent ideas aroused my deep interest in MRI and I decided to take MRI as my topic of research in my PhD. Dr. Hammad has put days and nights providing me with answers and advice, and passionately editing my papers. For that, I am forever in debt. Dr. Hammad had been a source of great help in my whole PhD. I very much appreciate his support.

Dr. Jeevan A/L Kanesan, my PhD co-supervisor, for his expert guidance and support.

Mr. Chen Wei (GE Applications Specialist), I owe a great deal of gratitude for his great support when I needed the help. His knowledge of the internals of the MR system, MR imaging and GE scanner protocols is second to none. I must also acknowledge his guidance in setting up experiments on the MRI scanners at the University of Malaya Hospital.

Prof. Dr. Norlisah Ramli, for allowing me to use MRI resources at University of Malaya Medical Centre (UMMC).

Ms. Ai Leng Wui, for his help in setting up experiments on the MRI scanners at University of Malaya Medical Centre (UMMC). I could not have done my experiments at UMMC without her help, which I really appreciate.

Finally, and most importantly, I feel a profound sense of gratitude towards my parents for their never-ending support, love, prayers and continuous encouragement in order to achieve the highest academic qualification. I am also grateful to my wonderful wife *Ayesha,* thank you for being so patient, and thank you for giving me such loving support at every stage of my PhD. Finally, it will be unfair not to thank my son *Arham Faisal* for enlightening our lives with his love!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AF	:	Acceleration Factor
AP	:	Artefact Power
ASIC	:	Application Specific Integrated Circuit
CORDIC	:	COordinate Rotation DIgital Computer
CPU	:	Central Processing Unit
CS	:	Compressed Sensing
DWT	:	Discrete Wavelet Transform
FOV	:	Field Of View
FPGAs	:	Field Programmable Gate Arrays
GPU	:	Graphical Processing Unit
GRAPPA	:	Generalized Autocalibrating Partially Parallel Acquisitions
HDL	:	Hardware Descriptive Language
IP	:	Intellectual Property
LS-SVM	:	Least Square Support Vector Machine
MRI	:	Magnetic Resonance Imaging
NMR	:	Nuclear Magnetic Resonance
PCA	:	Principal Component Analysis

PET	:	Positron Emission Tomography
PI	:	Parallel Imaging
pMRI	:	Parallel Magnetic Resonance Imaging
SENSE	:	SENSitivity Encoding
SVM	:	Support Vector Machine
TE	:	Echo Time
TR	:	Repetition Time

CHAPTER 1: INTRODUCTION

1.1 Background

Magnetic Resonance Imaging (MRI) has seen a wide use by the medical practitioners to identify different pathological conditions of the patients. MRI is an advanced imaging modality (like Positron Emission Tomography (PET) and X-ray Computed Tomography (X-ray CT)) but superior to other technologies because it is a non-invasive and non-ionising technique. In addition, MRI is sensitized by the presence and properties of water in the body tissues. MRI has proven itself as a low risk, dominant and flexible assessment technique for medical examination over the years because of its features, like better soft tissue differentiation, high contrast and spatial resolution. MRI can detect certain diseases much earlier than other medical imaging techniques (Bauer, Wiest, Nolte, & Reyes, 2013).

An MRI system consists of different sub-systems. These sub-systems include high field magnets, gradient coils, receiver coils, transmit coils, RF amplifiers, controllers and workstation. All the acquired raw data are transferred to the workstation, where post-processing (Image reconstruction) is performed to produce the MRIs. Since MRI invention in the 1970s, one major limitation of MRI has been its long data acquisition time, which challenges the use of MRI for some applications and also increases the hospital resource usage. A significant effort to increase the imaging speed in MRI has been done in the recent past by improving the magnetic field strengths, gradient hardware and fast pulse sequence development. These advancements are fundamentally limited by the physical (Gradient amplitude and slew rate) and physiological (Nerve stimulation) constraints. The researchers have already met these limits; therefore, scan

time reduction by reduced data acquisition (encode the data more quickly) has received significant research attention more recently.

In reduced data imaging, advance image reconstruction algorithms are used to reconstruct fully sampled images without degrading the quality of the image. These methods can be categorised into two groups: (a) Techniques which rely on coherent under-sampling (e.g., parallel imaging); (b) Techniques which rely on in-coherent under-sampling (e.g., sparse acquisition) (Griswold et al., 2002; Holland et al., 2013; Hugger et al., 2011; M. Lustig, Donoho, & Pauly, 2007; Nam et al., 2014; Pruessmann, Weiger, Scheidegger, & Boesiger, 1999; Tayler, Holland, Sederman, & Gladden, 2011). The concept of Parallel Imaging (PI) in MRI has been a standout advancement which enabled to minimize the MRI scan time by acquiring the data in parallel. PI uses multiple receiver coils and skips some phase encode lines in *k*-space (raw data space in MRI) to reduce the data acquisition time. Parallel MRI (pMRI) produces aliased images due to under-sampling in the acquired *k*-space. Some suitable reconstruction algorithms (Bydder, Larkman, & Hajnal, 2002b; Griswold et al., 2002; Griswold, Jakob, Nittka, Goldfarb, & Haase, 2000; Heidemann, Griswold, Haase, & Jakob, 2001; Jakob, Grisowld, Edelman, & Sodickson, 1998; Kyriakos et al., 2000; Larkman & Nunes, 2007; Pruessmann et al., 1999; Sodickson, 2000; Sodickson & Manning, 1997; J. Wang et al., 2001) are required to remove this aliasing and to get the full resolution image.

The recent developments in MRI reconstruction algorithms also demand the best possible solution for their implementation in hardware. These platforms may consist of computation cores, general purpose Central Processing Unit (CPU), general purpose Graphics Processing Unit (GPU), Field Programmable Gate Arrays (FPGAs), and combination of these (Chiuchişan & Cerlincă, 2013; Cong, Sarkar, Reinman, & Bui,

2011; Dalal & Fontaine, 2006; Stone et al., 2008; B. Wang et al., 2009; Y. Wang et al., 2010; Xu, Cai, Gao, Zhang, & Hsu, 2007; Zhuo & Prasanna, 2005).

1.2 Problem Statement

In recent literature, different real-time parallel imaging reconstruction algorithms for pMRI have been investigated (Hansen, Atkinson, & Sorensen, 2008; Hansen & Sørensen, 2013; Saybasili, Herzka, Seiberlich, & Griswold, 2014; Saybasili, Kellman, Griswold, Derbyshire, & Guttman, 2009; Shahzad, Sadaqat, Hassan, Abbasi, & Omer, 2016). All these techniques are able to reconstruct the images once the raw data is available on the workstation. However, this study proposes a different approach to accelerating MRI i.e. a compact design which can be integrated in the receiver coil system, thus no need to transmit all the raw data (Hundreds of Megabyte) to the control room, which will improve the signal strength and SNR because the image will be reconstructed right on the receiver coil data acquisition system. Moreover, GPU or CPU or GPU+CPU platforms based solutions consume high power. This research is also focusing on providing a low power system implementation which can be used in modern portable MRI scanners. Thereby implementing a hardware based design for real-time SENSE reconstruction (most commonly used parallel MRI algorithm) architecture with less computational time, efficient memory usage, significantly less artefact power, good SNR, reduces the data transmission cost, and consumes less power while keeping the temporal quality of the images.

1.3 Aim

The main aim of this work was to develop a high throughput system for real-time MRI SENSE reconstruction, which has a potential to easily equip with the modern portable/compact MRI scanners.

1.4 Objectives

The objectives of the proposed research are as follows:

(i) To investigate a real-time optimized solution for reconstruction of parallel imaging aliased data.

(ii) To minimize the reconstruction time in parallel MRI.

(iii) To minimize the data transmission cost and memory usage in MRI systems.

(iv) To develop an efficient low power reconstruction system, especially for modern portable MRI scanners.

1.5 Thesis Contribution

The contributions of this thesis are the following:

(i) A novel FPGA implementation for real-time SENSE reconstruction in parallel MRI. The proposed system reduces the computational time for widely clinically used pMRI reconstruction technique i.e. SENSE. This system helps to reduce the transmission cost and memory usage of the conventional MRI system.

(ii) The proposed novel SENSE reconstruction system is also capable to reconstruct the under-sampled pMRI data with different sensitivity maps estimation methods.

(iii) A new method using LS-SVM for brain MRI classification. The results show that the proposed approach achieves significantly higher accuracy rate.

1.6 Thesis Outline

Chapter 2 is a brief account of MRI and parallel MRI thus provides a background for the forthcoming chapters which cover more advanced topics. Here I describe the basic MR imaging, *k*-space encoding, data acquisition in MRI, image resolution, basic description of parallel MRI and different domain approaches in parallel MRI.

Chapter 3 presents the proposed FPGA implementation for SENSE reconstruction. SENSE reconstruction is one of the most widely clinically used algorithm in MRI scanners these days. The methodology used to implement SENSE reconstruction is provided in this chapter. The proposed FPGA implementation of the real-time SENSE reconstruction provides a comprehensive and efficient tool to perform SENSE reconstruction in MRI scanners (especially in portable MRI scanners). The results show a successful implementation of SENSE on the data acquired by the actual scanners. The same system has been used to perform SENSE reconstruction in Chapter 4 as well.

In **Chapter 4**, SENSE reconstruction is performed using different sensitivity maps. The proposed FPGA implementation of the real-time SENSE reconstruction architecture is used to evaluate the results. In this chapter, two different sensitivity estimation methods (pre-scan and E-maps) are used for SENSE reconstruction. The proposed system provides comparable reconstruction results for both the sensitivity maps estimation methods.

Chapter 5 describes the proposed classifier to validate the MRI brain images. This medical decision support system successfully classifies the reconstructed images mentioned in the previous chapter to validate the results. The proposed new method achieves a high accuracy rate.

Finally, **Chapter 6** summarizes the contributions of my work to real-time SENSE reconstruction for parallel MRI and provides some insights into potential future research directions.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The literature review concludes the details of principles of MRI, parallel MRI and implementation of parallel MRI methods on different hardware platforms. A brief review of the physics and operational mechanism of the MRI has been discussed. Different parallel MRI reconstruction techniques have been exploited. The important aspects of a real-time implementation of parallel MRI reconstruction platforms have also been investigated in this chapter.

2.2 Nuclear Magnetic Resonance Physics

The actual elucidation of the Nuclear Magnetic Resonance (NMR) incorporates a quantum mechanical approach in its natural description, but on a macroscopic scale classical mechanics could be utilized to describe the NMR with quite precision. In this thesis, only the classical mechanical description is taken in to account for the NMR (McRobbie, Moore, Graves, & Prince, 2006).

2.2.1 Polarization

Protons are responsible for the generation of MRI signals in the body, particularly from the water molecules. A strong static field B_0 polarizes the protons, yielding a net magnetic moment oriented in the direction of the static field. It is the magnetization which actually produces the magnetic moment and hence the MRI signal. The direction of the concerned field and its orthogonal plane are often termed as the longitudinal direction and the transverse plane.

2.2.2 Bloch Equation

The magnetization *M* interaction with an external magnetic field *B* is governed by the Bloch equation:

$$\frac{dM}{dt} = M \times \gamma B + \frac{M_0 - M_z}{T_1} + \frac{M_{xy}}{T_2} \tag{2.1}$$

where M_0, M_z and M_{xy} are the equilibrium, longitudinal and transverse magnetization, respectively. Whereas, T_1 and T_2 are constants and are material specific and their values also depend on the types of tissues.

2.2.3 Resonance

A Radio Frequency (RF) excitation field B_1 is applied to the net total magnetization tips it and produces a magnetization component M_{xy} (or simply *m*), oblique to the static field. The magnetization B_0 starts precessing at a characteristic frequency and described as:

$$f_0 = \frac{\gamma}{2\pi} B_0 \tag{2.2}$$

here f_0 is the precession frequency, B_0 the static field strength, and $\frac{\gamma}{2\pi}$ is a constant (42.57 *MHz*/T). Normally a general purpose 1.5T clinical MR system has a frequency of about 64 *MHz*. The signal generated by the transverse component of the magnetization is detected by the receiver coil. The magnetization at a position *r* and time *t* is described by the complex expression (as shown below):

$$m(r,t) = \left| m(r,t) \right| . e^{-i\phi(r,t)} \tag{2.3}$$

where $|m(r,t)|$ is the magnitude of the transverse magnetization and $\phi(r,t)$ is its phase. The phase shows the inclination and direction of the magnetization on the transverse plane. The transverse magnetization $m(r)$ can represent numerous physical attributes of tissue, e.g. proton density, T1 and T2 relaxation. The desired image in MRI is $m(r)$, the image of the spatial distribution of the transverse magnetization.

2.2.4 T_1 and T_2 Relaxation

When the RF pulse is turned off, the transverse magnetization M_{xy} undergoes relaxation. The longitudinal magnetization component M_z recovers exponentially with a time constant T_1.

$$M_z(t) = M_0\left(1 - e^{\left(\frac{-t}{T_1}\right)}\right) \tag{2.4}$$

In the same way the transverse magnetization component M_{xy} also undergoes an exponential decay with a time constant T_2.

$$M_{xy}(t) = M_0\, \mathrm{e}^{i\omega_0 t} \cdot e^{\left(\frac{-t}{T_2}\right)} \tag{2.5}$$

These relaxation parameters (T_1 and T_2) have variation in their values because of the tissues' nature and are the intrinsic parameters which define the contrast in an MR image. There are T_1 weighted or T_2 weighted MR images relying on the selection of these relaxation parameters predefined in the pulse sequence.

2.3 MR Imaging

2.3.1 Localized Slice Excitation

Gradient fields help to excite only a slice of interest in the imaging volume because the gradient coil exhibits a locally varying magnetic field. Thus, in order to excite only a specific area of the patient's body, an RF pulse approaching to the frequency closest to 'Larmor frequency at that particular slice' is transmitted. As a result, other slices having different precession frequencies (due to gradient fields) would not be able to absorb the applied RF energy due to the resonance. The pulse centre frequency is determined by the slice position and the strength of the sliced gradients at any particular location. The thickness of the excited slice is controlled by the pulse bandwidth (range of frequencies within the pulse).

2.3.2 Spatial Encoding and *k*-space

A method (called Spatial Encoding) as explained in Section 2.3.1, is used to obtain a localized signal from a specific location in a human body. Spatial encoding in MRI is said to be comprised of the phase and frequency encoding gradients. An incremental phase encoding gradient is applied where each phase encode gradient value provides spatial encoding in terms of the phase. To complete the spatial information, a frequency encoding gradient (also known as readout gradient) is also applied simultaneously with the phase encoding gradient.

The MR signal is obtained during the frequency gradient. Sampling is done on the obtained signal and stored at different locations in a raw data matrix (known as *k*-space). The RF pulse is excited again and again until the whole *k*-space is filled (line by line) incorporating the different phase encoding gradients followed by the readout gradient. Importantly, this process of filling the *k*-space requires more number of phase

encode steps and each phase encode gradient requires a new RF excitation. This process leads to increase the time for the data acquisition. However, the readout gradient does not waste any extra time because it is applied in parallel with the phase encode gradient. From this discussion, it can be construed that the time to acquire MR images primarily depends on the amount of phase encode gradient steps. As *k*-space has the information of the MR image in the frequency domain, therefore, 2-D Inverse Fourier Transform of *k*-space produces MR image in the image domain.

The centre of the *k*-space depicts the information of low frequency components of the image e.g. the contrast information of the image. Whereas, the outer regions of the *k*-space contains the high frequency information of the image e.g. edges, contours etc.

2.3.3 Pulse Sequence and MR data acquisition

RF excitations are required at many levels while constructing the MR image along with the application of the gradient fields in a well oriented manner. It is not a recommended approach to extract MR image by fewer excitation because the MR signal decays exponentially, which decreases the image quality. Moreover, the performance of gradient system and physiological constraints put a limit on the quick transversal of the *k*-space. The main aim of the MR data acquisition is to fill the *k*-space. Once the *k*-space has been filled, its inverse Fourier transform can be employed to transform the MR data into image domain.

The filling of the *k*-space is defined by the pulse sequence shape e.g. Cartesian, Radial, Spiral etc. The MR signal is defined by the Bloch equation:

$$M_z(t) = M_0\left(1 - e^{\left(\frac{-t}{T_1}\right)}\right)e^{\frac{-TE}{T_2}} \tag{2.6}$$

here M_0 is the maximum detectable signal (dependent upon the magnetic field strength B_0 and the proton density); T_1 and T_2 are the relaxation time. Whereas, *TR* and *TE* are the RF pulse repetition time and the echo time, respectively (Figure 2.1). This equation signifies the strength of the MR signal "*M*" detected by the system and which significantly depends upon the pulse repetition time "*TR*" and the echo time "*TE"*, therefore, the values of *TR* and *TE* must be chosen with extreme care to have a maximum contrast. The desired MR image is produced by the selection of an appropriate pulse sequence with suitable protocol to show the required properties of the T_1 or T_2. In MRI scanners, variety of pulse sequences are available e.g. Spin Echo and its derivatives, Gradient Echo and its derivatives, EPI etc, each suitable for different applications (McRobbie et al., 2006).

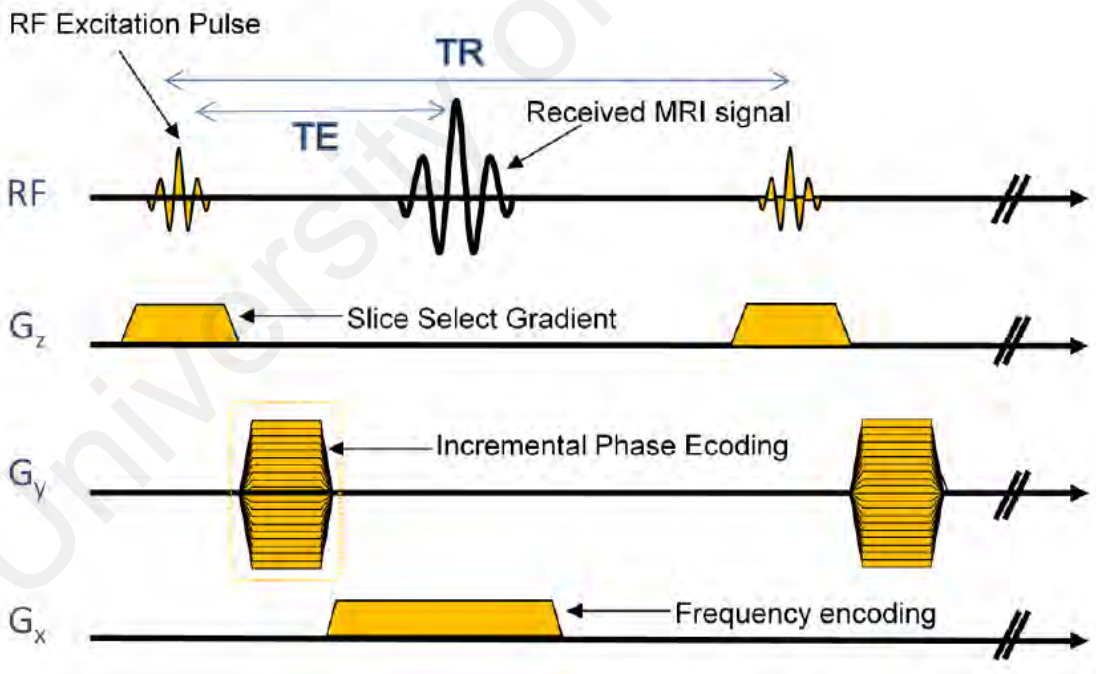


Figure 2.1: A general representation of a pulse sequence diagram, 'TR' is the repetition time between two RF pulses and 'TE' is the Echo time (Omer, 2012).

2.3.4 Image Resolution and Field of View

The number of samples acquired in the direction of frequency encode defines the image resolution and Field of View (FOV) of the image is defined by the number of phase encoding steps. The number of frequency encoding samples may be conventionally taken as 256 or 512 as it does not add extra time in the process of the image acquisition. However, the phase encoding steps affect the MR data acquisition time to a greater extent. The FOV is defined by the distance between the adjacent *k*-space lines (as shown in Figure 2.2). The mathematical relation between *k*-space line and FOV is defined as:

$$\Delta k_y = \frac{2\pi}{FOV_y} \tag{2.7}$$

where Δk_y is represented the gap between the two adjacent *k*-space lines, the subscript *y* shows the phase encode direction (Larkman & Nunes, 2007).

2.4 Parallel Imaging

In conventional MRI system, the phase encoding procedure is performed sequentially and consumes majority of the scan time. This process is done by switching the phase encode gradient (the magnetic field gradient) for every acquired *k*-space line and this is time consuming. Unfortunately, MRI system has some limits to switch magnetic field gradients rapidly. These limits are associated with the physical (hardware technical issues) and physiological constraints; therefore, the only solution to increase the imaging speed is encode the data more quickly.

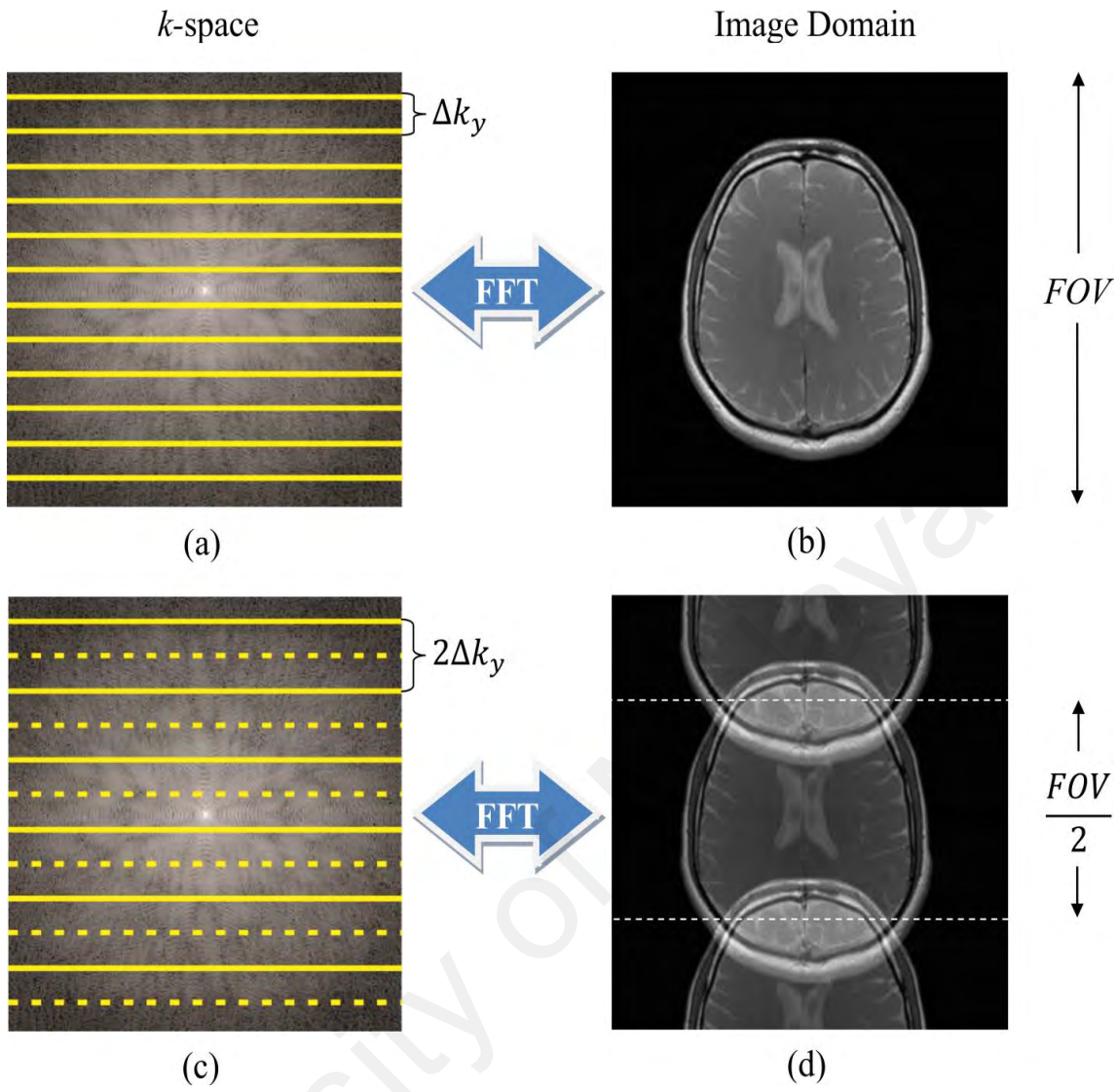


Figure 2.2: Relationship between *k*-space lines and FOV (a) Fully sampled *k*-space (b) Image domain representation (c) Doubled the gap between two adjacent lines in *k*-space (d) FOV reduces to half which may cause aliasing.

Parallel MRI (pMRI) is one of the most advance developments in magnetic resonance imaging in the last decade. pMRI decreases the acquisition time without the need of further increase the gradient performance. pMRI is a technique which uses multiple receiver coils for acquiring the data in parallel. Multiple independent channel receivers generate multiple coil images and each coil image is weighted differently by the spatial sensitivity of its coil. pMRI uses the spatial information (which is inherent in the local coil arrays) in conjunction with gradient encoding to reduce the scan time. The additional knowledge of the spatial sensitivity information allows to reduce the number

of phase encode steps which substantially reduces the acquisition time in MRI (Blaimer et al., 2004; Larkman & Nunes, 2007).

2.4.1 Parallel MRI reconstruction techniques

Over the years, great development progress in parallel MRI field has been done; thereby many solutions are investigated for pMRI reconstruction techniques (Blaimer et al., 2004; Larkman & Nunes, 2007). These techniques can be categorized in 'image-domain' methods (e.g. SENSE) and '*k*-space' methods (e.g. GRAPPA). The most well known and clinically used techniques are SENSE (Pruessmann et al., 1999) and GRAPPA (Griswold et al., 2002). However, various other methods such as SMASH (Sodickson & Manning, 1997), AUTO-SMASH (Jakob et al., 1998), g-SMASH (Bydder et al., 2002b), mSENSE (Wang et al., 2001), PILS (Griswold et al., 2000), SPACE RIP (Kyriakos et al., 2000) and SPIR-iT (Lustig & Pauly, 2010) have also been developed.

The 'image-domain' and '*k*-space' methods differ in the stage at which the reconstruction process has to be done (Figure 2.3). In 'image-domain' algorithms, firstly the acquired partial *k*-space is transformed by inverse Fast Fourier Transform (iFFT) to image domain which generates corresponding aliased images. Then image domain processing is applied to reconstruct the desired image from the pMRI under-sampled data. However, the method of *k*-space operates at partial *k*-space and estimates the missing data of the *k*-space. Once all the missing *k*-space lines have been estimated then iFFT of the full *k*-space is obtained to get a full FOV MR image.

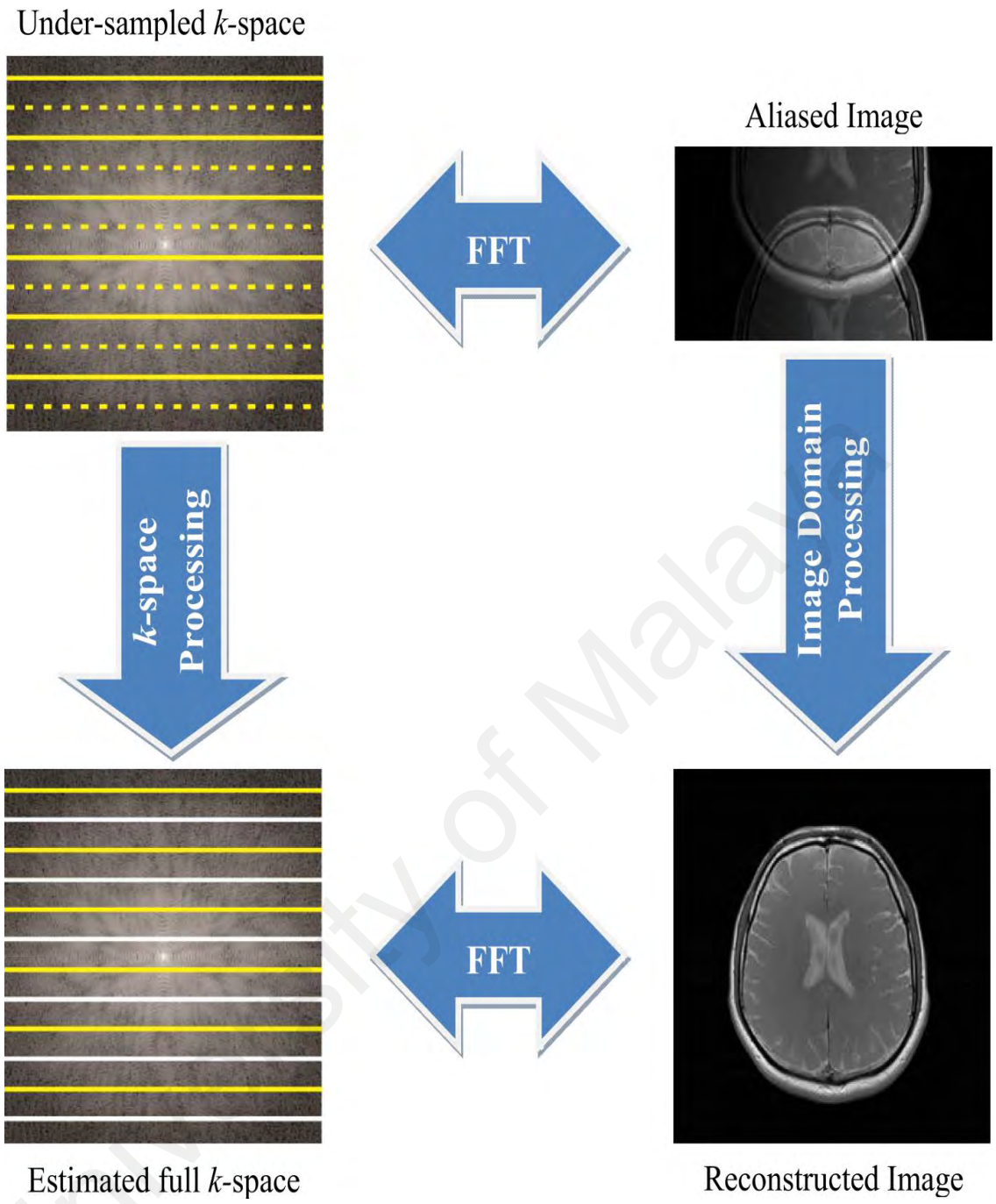


Figure 2.3: A general description of 'image-domain' and '*k*-space' based pMRI techniques.

Table 2.1 shows the acronyms of pMRI methods and terminology used by the MRI scanner manufacturers. Most commonly employed pMRI algorithms in commercial MRI systems are SENSE, GRAPPA and their variant methods.

Table 2.1: Parallel MRI methods used in commercial MRI scanners.

Name	Acronym	Method	Manufacturer
SENSitivity Encoding	SENSE	Image-based reference	Philips
Array Spatial Sensitivity Encoding Technique	ASSET	Image-based reference	General Electric
Auto-calibrating Reconstruction for Cartesian Imaging	ARC	Image-based reference scan hybrid (image- and *k*-space based)	General Electric
GeneRalized Auto-calibrating Partially Parallel Acquisition	GRAPPA	k-space based, auto-calibrated with reference scan option	Siemens
modified SENSitivity Encoding	mSENSE	Image-based, auto-calibrated with reference scan option	Siemens
SPEEDER	---	Image-based, reference scan	Toshiba

SENSE is an 'image-domain' pMRI reconstruction technique. In this method, the following operations are applied on the acquired under-sampled *k*-space (from the MRI scanner) to reconstruct the full FOV image (Figure 2.4):

1. iFFT of the reduced *k*-space data of each receiver coil to produce aliased images.
2. Find out the solution of a system of linear equations by using the knowledge of the coil sensitivity profiles, which produces the final un-aliased image.

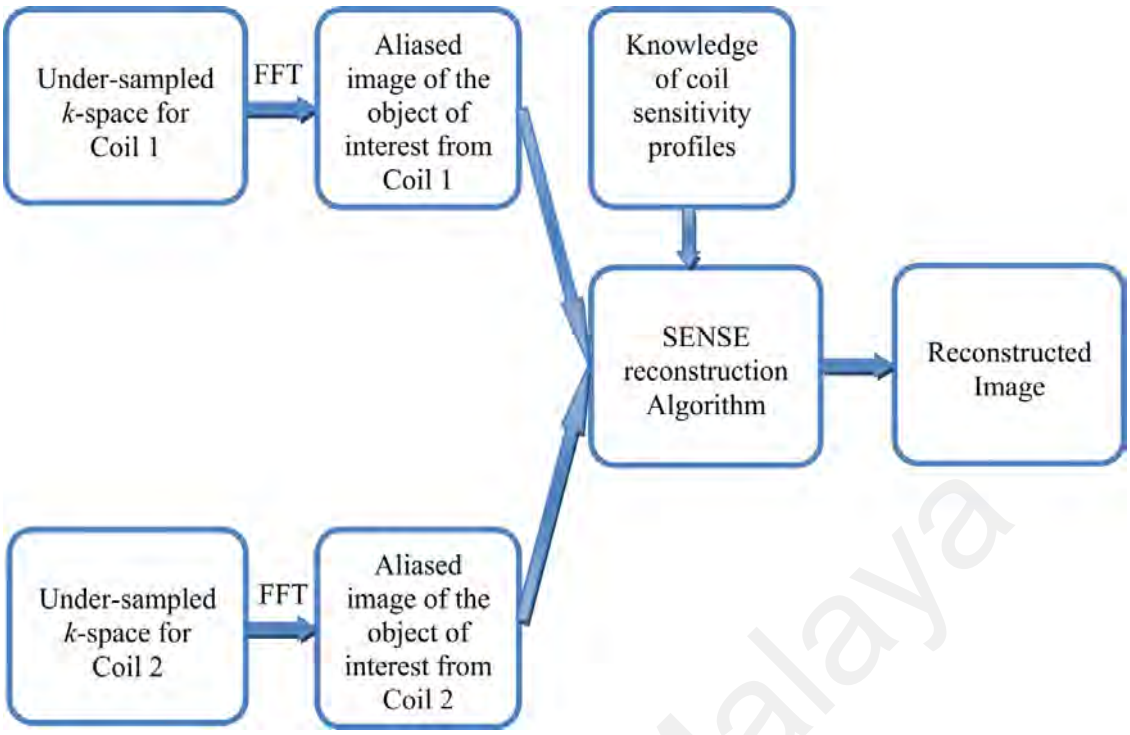


Figure 2.4: An overview of 'image-domain' Parallel MRI (SENSE).

GRAPPA reconstruction is a '*k*-space' based method and it directly operates on the acquired reduced *k*-space (with some additional lines for calibration) with the following steps (Figure 2.5):

1. The additional data (calibration lines) in the reduced *k*-space is used to estimate the weights for the missing *k*-space lines.
2. The weights are then applied to estimate missing *k*-space lines for each coil data which leads to generating fully sampled estimated *k*-space.
3. iFFT of this fully sampled *k*-space produces images for each receiver coils data.
4. Combine all the receiver coils images by applying "sum-of-squares" technique which produces the final reconstructed image.

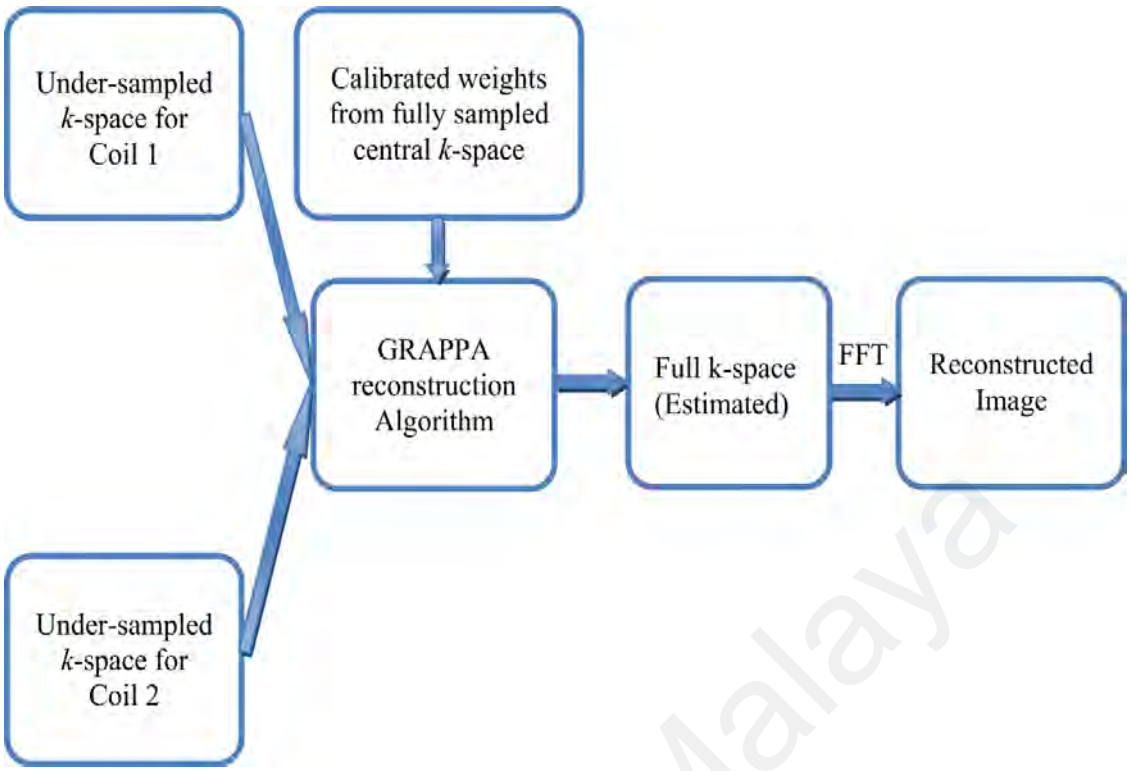


Figure 2.5: An overview of k-space space Parallel MRI (GRAPPA).

Apart from the aforementioned devices, non-Cartesian trajectories based pMRI techniques have also been investigated. The non-Cartesian *k*-space trajectories (such as spiral and radial) combined with pMRI is another emerging field of research at present era. Non-Cartesian sampling offers some distinct advantages (e.g. reduce the MRI data acquisition time, reduce the motion artefacts and use of gradient system efficiently) as compared to conventional Cartesian based pMRI. However, the main disadvantage of this method is that it requires more complex computational, such as density compensation function and gridding, steps for reconstruction. These additional expensive computation steps lead to increase the reconstruction time, which is undesirable in clinical MRI scanners. Conjugate Gradient SENSE (CG-SENSE) (Pruessmann, Weiger, Börnert, & Boesiger, 2001), Non-Cartesian GRAPPA (Griswold, Heidemann, & Jakob, 2003; Heidemann et al., 2006; Seiberlich, Ehses, Duerk, Gilkeson, & Griswold, 2011; Seiberlich, Lee, et al., 2011), PARS (Yeh, McKenzie,

Ohliger, & Sodickson, 2005), PILS (Griswold et al., 2000) and SPIRiT (Lustig & Pauly, 2010) are some of the examples of non-Cartesian parallel imaging algorithms.

2.4.2 Parallel MRI reconstruction hardware platforms

Recent advancements in pMRI significantly reduce the data acquisition time in MRI but increase the reconstruction time and also the complexity of the reconstruction algorithms. Therefore, pMRI requires more sophisticated hardware platforms for image reconstruction. General purpose single core processor computer systems are not adequate to handle the computational load involved in pMRI reconstruction. Consequently, the modern MRI scanners have multi-core CPU systems (such as Blade CPU) for reconstruction purposes. Moreover, computer clusters are also used for MRI reconstruction in past (Kressler, Spincemaille, Prince, & Wang, 2006). However, general purpose multi-core CPU computer systems are not application specific and not offer massively parallel computations. Computer clusters platform for MRI reconstruction is expensive, not easy to maintain, and normally installed far from the MRI scanner, therefore, it is not a feasible practical solution. In literature, different feasible and practical alternative hardware platforms (such as GPU or FPGA) are exploited instead of large clusters systems for MRI reconstruction. GPU and FPGA offer highly parallel computing which is ideal for pMRI reconstruction.

Table 2.2 shows a brief overview of the real-time implementation of pMRI on different hardware platforms. In previously published work, FPGAs are used to accelerate the MRI reconstruction and offer real-time reconstruction as well. In (Dalal & Fontaine, 2006), the authors proposed a reconfigurable FPGA based design for multi-coil MRI data reconstruction. They claimed that this FPGA implementation is more liable than CPU and cluster computing. The authors have also introduced dynamic partial configuration which allows the programmers to re-program only the partial part of the

Table 2.2: Summary of parallel imaging reconstructions implemented on different hardware platforms.

Title	Reconstruction Algorithm	Sampling Schemes	Hardware Platform	Portability	Reconstruction before transmission
A Reconfigurable FPGA-based 16-Channel Front-End for MRI (Dalal & Fontaine, 2006)	2DFFT	Cartesian	FPGA	Easy	Yes
Design of an MR image processing module on an FPGA chip (Li & Wyrwicz, 2015)	2DFFT	Cartesian	FPGA	Easy	Yes
Cartesian SENSE and k-t SENSE Reconstruction Using Commodity Graphics Hardware (Hansen et al., 2008)	SENSE and k-t SENSE	Cartesian	GPU	Difficult	No
Parallel MRI reconstruction Algorithm Implementation on GPU (Shahzad et al., 2016)	SENSE	Cartesian	GPU	Difficult	No
Gadgetron: An Open Source Framework for Medical Image Reconstruction (Hansen & Sørensen, 2013)	GRAPPA	Cartesian	GPU	Difficult	No
Real-Time Flow With Fast GPU Reconstruction for Continuous Assessment of Cardiac Output (Kowalik et al., 2012)	Iterative SENSE	Arbitrary	GPU	Difficult	No
Accelerating advanced MRI reconstructions on GPUs (Stone et al., 2008)	CG-SENSE	Arbitrary	GPU	Difficult	No

Table 2.2: continued.

Title	Reconstruction Algorithm	Sampling Schemes	Hardware Platform	Portability	Reconstruction before transmission
Real-time imaging with radial GRAPPA Implementation on a heterogeneous architecture for low-latency reconstructions (Saybasili et al., 2014)	Radial GRAPPA	Radial	GPU	Difficult	No
Algebraic Reconstruction Technique for Parallel Imaging Reconstruction of Undersampled Radial Data: Application to Cardiac Cine (Li et al., 2015)	Algebraic	Radial	GPU	Difficult	No

FPGA. This FPGA implementation has consumed 5 *ms* to reconstruct 256×256 matrix size image, i.e., 200 images per second and 25 frames/s for 8-channel receiver coils data. Another FPGA based implementation for multi-coil MRI data real-time reconstruction has been proposed in (Li & Wyrwicz, 2015) with the performance of 400 frames/s (has consumed 2.5 *ms* for 128×128 matrix size image). There are two advantages of this implementation over previous works: (1) no off-chips hardware are used which increases portability of the design (2) address generation technique is used instead of direct matrix transposition for computing 2D FFT. These both techniques can reconstruct MR image from multi-coil MRI data in real-time and can equip with the data acquisition system of MRI. However, these techniques are limited for multi-coil MRI data and currently not capable to reconstruct the images from under-sampled parallel MRI data. Therefore, there is a gap to introduce such modules which can work for pMRI reconstruction as well (Dalal & Fontaine, 2006; Li & Wyrwicz, 2015).

GPU based implementations have also recently attracted research interests for MRI reconstruction. Hansen et al. (Hansen et al., 2008) proposed a commodity graphics hardware (also known as GPU) implementation for Cartesian SENSE and k-t SENESE reconstruction. Cholesky method has used in this implementation to solve the linear equations (SENSE). The authors have achieved approximately 2 *ms* computation time for 8-coils data with an acceleration factor of 2. Hansen et al. have also produced an open source framework for medical image reconstruction (named: Gadgetron) for the researchers to contribute and organize their implementations on this platform. Several different reconstruction modules are available in the form of Gadgets for the researchers to re-use them in their implementations (Hansen & Sørensen, 2013). In recently published work (Shahzad et al., 2016), the authors have proposed a GPU implementation for SENSE with left pseudo-inverse method. This implementation has achieved 4.7 *ms* for the reconstruction of 256×256 pixel-resolution image (Shahzad et

al., 2016). Moreover, GPU based implementation for non-Cartesian pMRI real-time reconstruction has also been investigated. Several solutions are suggested in the literature to decrease the reconstruction time for non-Cartesian pMRI using the GPU platform. Conjugate gradient method is implemented on GPU in (Stone et al., 2008), to accelerate the reconstruction speed of non-Cartesian pMRI. This implementation allows to reconstruct the non-Cartesian spiral trajectory data 21 times faster than the quad core CPU based reconstruction. Kowalik et al. proposed a novel GPU implementation of non-Cartesian pMRI reconstruction (iterative SENSE) for real-time assessment of cardiac MRI data. Spiral trajectory is used in this algorithm and achieved 7.7 times faster reconstruction than the CPU reconstruction (Kowalik et al., 2012). Moreover, GPU implementation of real-time radial trajectory data reconstruction has also been explored. Radial GRAPPA reconstruction is not an iterative and robust algorithm, but computationally expensive. A heterogeneous system using multi-core CPUs and GPUs has proposed to implement real-time radial GRAPPA reconstruction especially for cardiac and dynamic musculoskeletal MRI data with the performance of significantly less reconstruction time (i.e. 67 *ms*) (Saybasili et al., 2014). In (Li et al., 2015), the authors have proposed algebraic reconstruction technique (ART) for non-Cartesian PI reconstruction of under-sampled radial data and implemented it on the GPU. The reconstruction time of GPU-accelerated ART is 15 times faster as compared to CPU implementation.

To overcome the shortcomings of previously reported state-of-the-art work, FPGA implementation of real-time SENSE reconstruction is proposed in this thesis. The proposed implementation is capable to handle under-sampled pMRI data, having no data transfer overhead, uses memory efficiently, reduces the data transmission cost, consumes less power, and offers portability.

2.5 Summary

In this chapter, MRI fundamentals, parallel MRI and real-time implementation of pMRI is discussed. By critically analysing various works, there have been raised several issues of concern regarding the implementation of real-time pMRI reconstruction algorithms. The main points are summarized as follows:

- Real-time parallel imaging reconstruction algorithms are investigated on different platforms (Multi-core CPU, GPU, CPU+GPU, FPGA).
- FPGA implementations of MRI reconstruction algorithm in previously published work are only capable to reconstruct the images from fully-sampled data.
- All real-time MRI techniques implemented on multi-core CPU, GPU or CPU+GPU platforms are able to reconstruct the images once the raw data is available on the workstation. That means all the pMRI raw data should be first transfer to the workstation before reconstruction, which eventually increases the memory usage and transmission cost.
- FPGA platform offers portability as compared to other platforms. However, GPU implementation required CPU or multi-core CPU as a host system; therefore, it is an expensive solution especially for portable MRI scanners.
- MRI raw data is normally stored in host system (CPU or multi-core CPU) in the GPU implementation of MRI reconstruction; therefore, data transfer overhead is introduced in GPU based implementations. This data transfer from host system memory to GPU memory normally takes time more than the reconstruction operational time, which leads to increase the overall reconstruction time.

- These platforms (CPU, multi-core CPU or GPU) consume higher power than FPGA platform. The FPGA platform provides a low power solution for pMRI reconstruction, which is a suitable feature for portable MRI scanners.

CHAPTER 3: FPGA IMPLEMENTATION FOR REAL-TIME SENSE RECONSTRUCTION

3.1 Introduction

In recent years, Magnetic Resonance Imaging (MRI) has seen a wide use in hospitals for imaging various parts of the patient's body. However, one major limitation of MRI currently is its long data acquisition time, which challenges its utilization for some applications and also increases the hospitals' resource usage and power consumption.

Parallel Imaging (PI) has been a standout amongst the most eminent advancements in the MRI field which empowers to increase the rate of the MRI data acquisition by acquiring the data in parallel. In PI, the use of multiple receiver coils and skipping some phase encode lines in *k*-space (raw data space in MRI) reduces the data acquisition time significantly. This under-sampling in *k*-space produces aliasing in the MRI image and some suitable reconstruction algorithm is required to remove this aliasing. Parallel MRI (pMRI) reconstruction techniques have been the central focus of research in recent years to remove this aliasing. Different solutions for pMRI reconstruction have been proposed by the researchers, which can be broadly categorized into 'image-domain' approaches (e.g., SENSE) and '*k*-space' approaches (e.g., GRAPPA) as discussed in Section 2.4.1. Parallel imaging (e.g., SENSE algorithm) combined with Compressed Sensing (CS) (Lustig et al., 2007) based hybrid techniques have also been exploited to further increase the acceleration factor in MRI scans. These hybrid techniques have provided better reconstruction image quality and/or acceleration factor by overcoming the individual algorithm deficiencies (Liang, Liu, Wang, & Ying, 2009; Lustig & Pauly, 2010; Otazo, Kim, Axel, & Sodickson, 2010; Pawar, Egan, & Zhang, 2015). In recent literature, different real-time parallel imaging reconstruction algorithms have also been investigated (Hansen et al., 2008; Hansen & Sørensen, 2013; Saybasili et al., 2014;

Saybasili et al., 2009; Shahzad et al., 2016). All these techniques are able to reconstruct the images once the raw data is available on the workstation. However, this research proposed a novel architecture design for real-time SENSE reconstruction right on the receiver coil data acquisition system with no need to transfer all the raw data to the server (workstation).

SENSE (Pruessmann et al., 1999) is computationally intensive by nature, which may consume longer time and power if not optimally implemented. To satisfy such computation-hungry applications effectively, different platforms are used, e.g., computation cores, general purpose Central Processing Unit (CPU), general purpose Graphics Processing Unit (GPU), Field Programmable Gate Arrays (FPGAs), or a combination of these (Birk, Zapf, Balzer, Ruiter, & Becker, 2014; Chiuchişan & Cerlincă, 2013; Cong et al., 2011; Dalal & Fontaine, 2006; Eklund, Dufort, Forsberg, & LaConte, 2013; Kressler et al., 2006; Li & Wyrwicz, 2015; Omer & Dickinson, 2010; Pratx & Xing, 2011; Saybasili et al., 2014; Shahzad et al., 2016; Stone et al., 2008; Wang et al., 2009; Wang et al., 2010; Xu et al., 2007; Zhuo & Prasanna, 2005). FPGA platforms have a versatile mapping of application specific parallelism and high computational density per Watt compared to GPUs and multi-core CPUs. Modern GPUs provide multithreading and high memory bandwidth to increase the computational efficiency. However, FPGA based designs allow to explore parallelism in the algorithm in more depth by identifying the dependencies between the variables of the design at the hardware level. FPGA technology has promising power efficiency among other platforms, such as GPU or CPU. In addition, GPU needs a dedicated host system to operate which consumes larger space, high power and increases the cost of the design. However, some modern GPUs kits are available which can work stand alone, but still consume higher power. Furthermore, FPGA can work stand alone on peak performance, which yields to reduce the cost and overall power consumption of

the system. Furthermore, FPGA architecture provides a platform for researchers to validate the design, enables rapid prototyping of the complex algorithms, and a chance to avail debugging procedures.

A real-time reconstruction of SENSE has a great potential to decrease the computational time for pMRI image reconstruction, provides flexibility to store the reconstructed absolute data only instead of the huge amount of raw complex data, and also decreases the quantity of the data to be transmitted from the receiver coils to the workstation in an MRI system. Application specific hardware designs for FPGA provide greater speed than a software implementation on the general purpose platforms and also dissipate less power. Recently, researchers are more keen towards portable MRI scanner because it can be deployed in remote areas as well as in military field hospitals (Cooley et al., 2015; Kose & Haishi, 2011; Sarty, 2015; Zotev et al., 2008). The low power requirements of portable MRI scanners and lightweight construction, generate the requirement of a low power FPGA based application specific hardware design which would provide customized solution and greater speed for image reconstruction than a software solution on the general purpose platforms (GPU or/and CPU), and would dissipate less power at the cost of design efforts (Wang et al., 2010).

The main motivation of this work is to design a high throughput system for SENSE algorithm. The system needs to be more generalized so that it can provide a scalable hardware platform for researchers. In this chapter, a novel parameterized architectural design implementation of SENSE for different number of receiver coils is presented. A parameterized architectural design allows the designers to reconfigure the hardware databus widths and also provides feasibility to reuse the modules for scaling the architecture. Parallelism in SENSE algorithm was exploited in this research to increase the throughput of the design. The proposed architecture was developed using Hardware

Descriptive Language (HDL), i.e., Verilog. The novelty of the proposed scheme is an application specific hardware based design for real-time SENSE reconstruction in parallel MRI. Previously published FPGA based systems (Dalal & Fontaine, 2006; Li & Wyrwicz, 2015) are only capable to reconstruct the multi-channel fully-sampled MR images using the conventional method, but the proposed FPGA implementation is capable to reconstruct the images from pMRI under-sampled data using SENSE algorithm.

3.2 Material and Methods

3.2.1 SENSE Algorithm

SENSE is an image-domain algorithm in parallel MRI where scan time is reduced by acquiring fewer lines in *k*-space, which reduces the Field of View (FOV) producing an aliased image. SENSE is a widely used algorithm in clinical MRI scanners to reconstruct the unfolded image from the under-sampled data. In SENSE, the missing lines of the *k*-space are estimated by the knowledge of the receiver coil sensitivities. Coil sensitivity maps of the receiver coil elements are used to calculate the aliased signal component at each pixel location in the aliased image and these signals are allocated at the actual pixel positions in the unwrapped (reconstructed) image. Acceleration factor (denoted by "R") describes the extent of under-sampling in *k*-space. The FOV is defined by the distance between the adjacent *k*-space lines (Larkman & Nunes, 2007; Pruessmann et al., 1999).

Figure 3.1 illustrates a pictorial representation of SENSE algorithm for two receiver coils with an acceleration factor $R = 2$. The aliased pixel "S_1" in the aliased image from receiver coil 1, marked by a red square, contains the information of the two pixels "M_1" and "M_2" of the unfolded image. The same pixel positions on sensitivity profiles of

coil 1, marked by "C_{11}" and "C_{12}", will decide the weights of the "S_1" for locations "M_1" and "M_2", respectively. Similarly, for the aliased pixel "S_2" in the aliased image from receiver coil 2, marked by a yellow square, contains the information of the two pixels "M_1" and "M_2" of the unfolded image.

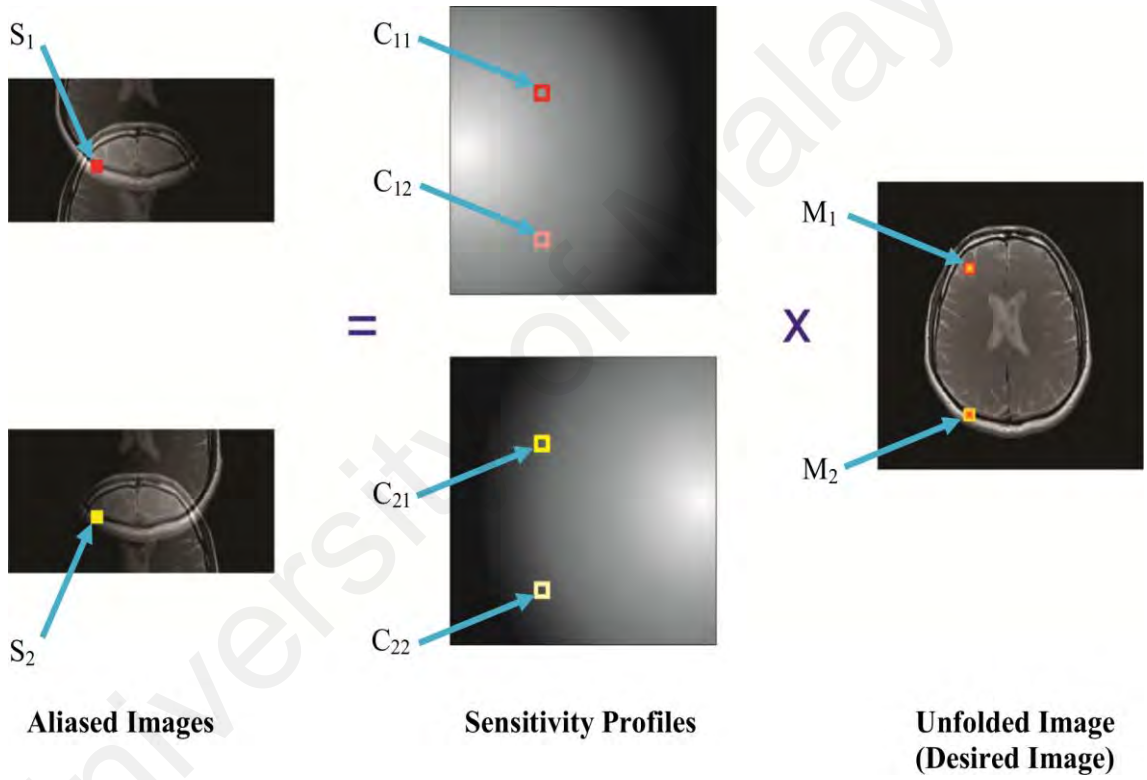


Figure 3.1: A pictorial representation of SENSE algorithm for two receiver coils with an acceleration factor $R = 2$.

The signal measured by each of the two receiver coils, for a pixel at location (x, y) in the aliased image is given by:

$$S_1(x, y) = C_1(x, y)\, M(x, y) + C_1(x, y + FOV/2)\, M(x, y + FOV/2) \qquad (3.1)$$

$$S_2(x, y) = C_2(x, y)\, M(x, y) + C_2(x, y + FOV/2)\, M(x, y + FOV/2) \qquad (3.2)$$

where "S" represents the signal value in the aliased image at a particular location. "C" is the encoding matrix and "M" represents the signal components to be estimated. In matrix notation, SENSE problem can be defined as:

$$\begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \cdot \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \qquad (3.3)$$

For more number of receiver coils (#coils = 4) and acceleration factor (R = 4) it will become:

$$\begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \\ C_{41} & C_{42} & C_{43} & C_{44} \end{bmatrix} \cdot \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \end{bmatrix} \qquad (3.4)$$

Theoretically, the acceleration factor should be less than the number of receiver coils. However, the maximum achievable AF for 8 receiver coils is 2 or 3 practically because of imperfections in the receiver coil sensitivities (Larkman & Nunes, 2007). So normally the number of receiver coils is always greater than the acceleration factor, therefore, SENSE problem will typically become an over-determined inverse problem e.g. (Equation 3.5) for 4 receiver coils with R = 2:

$$\begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \\ C_{31} & C_{32} \\ C_{41} & C_{42} \end{bmatrix} \cdot \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \qquad (3.5)$$

In matrix notation:

$$S = CM \qquad (3.6)$$

The number of rows of the encoding matrix "C" corresponds to the number of receiver coils (N_C), and the number of columns depends upon the number of overlapped signals at one location in the aliased image (N_A) (Larkman & Nunes, 2007; Omer & Dickinson, 2010). So the dimension of the "S", "C" and "M" matrices are $N_C \times 1$, $N_C \times N_A$ and $N_A \times 1$. "S" represents the aliased image components acquired from the scanner, "C" is derived from the receiver coil sensitivity maps[1]. So the solution image "M" is given by:

$$M = C^{-1}S \qquad (3.7)$$

But, usually the encoding matrix "C" is a rectangular matrix, so pseudo-inversion is required and Equation 3.7 becomes:

$$M = [(C^{*}C)^{-1}\, C^{*}]\, S \qquad (3.8)$$

The core process to implement SENSE reconstruction is to compute the pseudo-inverse of the matrix "C". Matrix inversion is a computationally intensive mathematical operation in hardware.

[1] Sensitivity maps details are provided in CHAPTER 4.

3.2.2 FPGA Implementation of SENSE

The importance of speed and power in modern portable MRI scanners generates the requirement to develop application specific hardware for real-time SENSE reconstruction. Application specific architecture designs achieve better performance; reducing the computation time for the scan, and decreasing the power consumption of the resources. Different hardware platforms or combination of different platforms may be used to implement such reconstruction algorithms. Table 3.1 provides a comparison of some of these platforms including CPUs, GPUs and FPGAs (Pereira, Athanas, Lin, & Feng, 2011; Sundararajan, 2010; Wang et al., 2010).

Table 3.1: Comparison of CPU, GPU and FPGAs.

	CPU (Core i7)	**GPU (GTX 780)**	**FPGAs (Virtex-6)**
Peak GFlops	70	3977	628
Design Effort	Easy	Middle	Hard
Design Size	Large	Large	Small
ASIC Implementation	No	No	Yes
Computational Density per Watt	Less	Medium	High
Power (W)	130	250	7

Reconfigurable hardware (such as FPGA) based designs for image processing (e.g., MR image reconstruction) improves the performance space ratio at the cost of the design efforts (Wang et al., 2010). FPGAs based designs can exploit different levels of parallelism according to the application with high computational density per Watt. The power efficiency of FPGAs is much better compared to other platforms such as CPU or GPU. Smaller design size also improves the silicon area efficiency of the FPGA based system and provides portability as an added advantage. Furthermore, instead of

spending heavy costs to provide a hardware solution for any application, the parameterized architecture design for FPGA provides a platform for the researchers to validate the design, enables rapid prototyping of complex algorithms, and a chance to avail debugging procedures. Moreover, FPGA prototype can be easily implemented in ASIC (application-specific integrated circuit) design.

This research proposed a parameterized architectural design of SENSE algorithm for different number of receiver coils (4, 6 and 8) for the acceleration factor of 2 and 3 as implemented in HDL (Verilog) on ML605 evaluation Virtex 6 kit. The input data matrices (Sensitivity maps and the acquired data) are stored in the memory of the FPGA. The SENSE architecture module performs the reconstruction operation on the given input (fetching data from memory) and then the reconstructed image (output) is transferred to MATLAB (R2013a) via UART transmission for analysis and comparison. Fixed point arithmetic is used to represent the decimal numbers in binary notation. Furthermore, complex numbers are saved in 16-bits real part and 16-bits imaginary part. However, the data widths can be changed as per requirement because the architecture is generalized by the parameterization method.

The design flow of the proposed FPGA implementation of SENSE is depicted in Figure ß3.2 and Figure ß3.3. In Figure ß3.2, the block diagram presents the operations of the left pseudo-inverse ($C^{+} = (C^{*}C)^{-1}C^{*}$) (Strang, 2009) to compute the matrix inversion in hardware which includes matrix transpose, complex matrix multiplier and square matrix inversion modules. However, encoding matrix data block represents the encoding matrix data fetches from the memory of the FPGA. Matrix transpose block generates the conjugate transpose of a matrix i.e. "C^{*}". It is implemented by only rearranging the inputs (rows to columns) and the sign of the complex part of the inputs, without performing any additional computations.

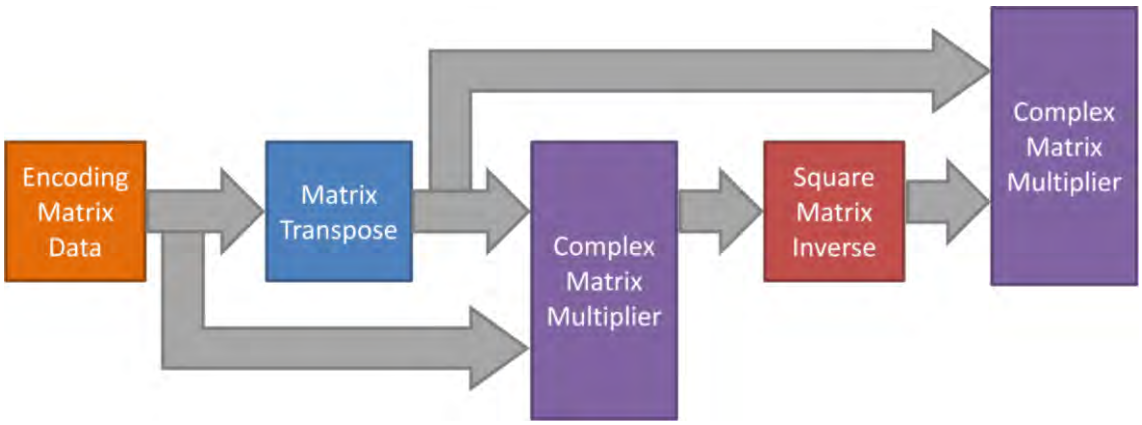


Figure 3.2: Block diagram of the pseudo-inverse block of the proposed SENSE architecture.

The complex matrix multiplier module is designed to multiply two complex matrices. It consists of add/sub and multiplier (complex) modules. For computing complex multiplication (such as $C_n^{-1}(x,y)\times S_n(x,y)$) in hardware, 3 multipliers and 4 add/sub modules are used instead of conventional complex number multiplication method which consumes 4 multipliers and 2 add/sub units. This approach has increased the efficiency of the architecture design because multiplier is more costly than add/sub operation in hardware implementation. Square matrix inverse module calculates the inverse of a given matrix. Matrix inversion is calculated by first finding the adjoint of a matrix ($adj(C^*C)$) then divided by its determinant. Divider IP-core (Intellectual Property-core) is used to perform division in the determinant method. Divider module consumes more resources than other modules; therefore, an effort has been made to use less number of divisions to implement the results in this architecture. The multiplicative inverse of a number module is designed to reduce the number of divisions. So multiplicative inverse value is multiplied by each element of the matrix when division is required. To compute the pseudo-inverse of the encoding matrix "C", the conjugate transpose "C^*" is multiplied by the encoding matrix "C" in the first step, which produces "C^*C". In the second step, the square matrix inverse block generates the inversion of the "C^*C"

matrix. Finally, square matrix inverse block output "$(C^{*}C)^{-1}$" is multiplied by conjugate transpose of an encoding matrix "C^{*}" to complete the pseudo-inverse operation. Once the matrix inversion is calculated, then two complex matrices (C^{+} and S) need to be multiplied for completing the operations of equation (3.8) which generates the conventional SENSE reconstructed image (Complex domain), as shown in Figure ꝑ3.3 conventional SENSE part.

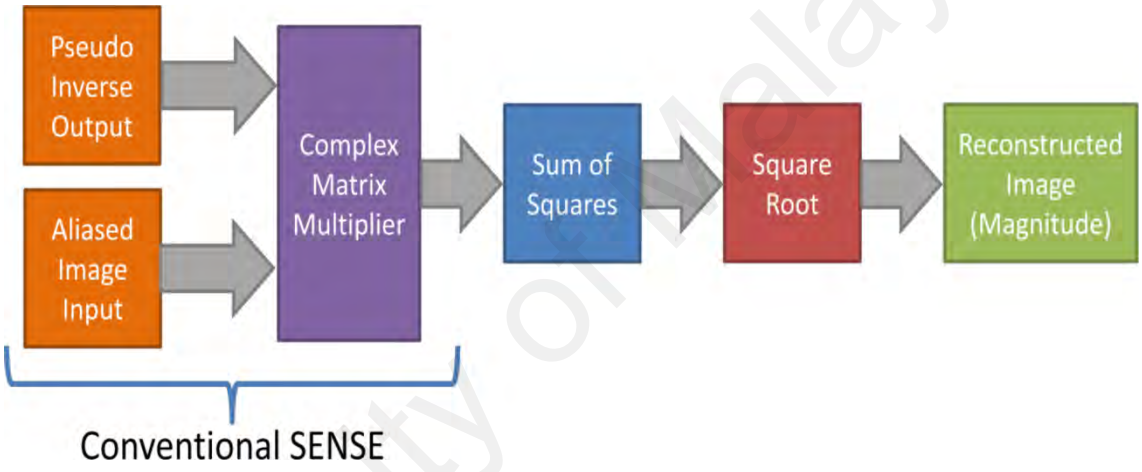


Figure ꝑ3.3: Block diagram of the proposed SENSE system implementation.

The main advantage of the architecture presented in this research is the real-time SENSE reconstruction right on the receiver coil data acquisition system so only the magnitude of the reconstructed image needs to be transferred to the MRI workstation (no need to transmit all the complex raw data). Therefore, few more steps (as shown in Figure ꝑ3.3) are needed to compute the magnitude of the complex reconstructed images (Equation 3.9):

$$|M| = \sqrt{M_r^2 + M_i^2} \qquad (ꝑ3.9)$$

The sum of squares of the real and imaginary parts of the reconstructed image (complex domain) is calculated using complex multiplier and adder modules. Then CORDIC (Walther, 1971) IP-core is used to compute the square root of the sum-of-squares to compute the magnitude of the reconstructed image. Finally, only the magnitude image of the reconstructed data is transferred from the receiver coil system to the MRI workstation to visualize the image.

The register transfer logic (RTL) schematic diagrams of the proposed real-time SENSE reconstruction system are shown in Appendices (A-D). The RTL schematic diagram of the top level of the proposed system is shown in Appendix A. It consists of memories, SENSE reconstruction, absolute comparator, controller unit, and data transmission (UART) modules. Memories are used to store folded images, sensitivity maps and reconstructed image. Folded images, sensitivity maps and output image memories are suffixed by imfold, cmap and output, respectively. Absolute comparator module (named as abs_comp) took real and imaginary values as input to generate the magnitude of the complex number, which is needed to compute the magnitude of the reconstructed image. The core process of the SENSE algorithm is implemented by the SENSE reconstruction module. It took the inputs (i.e., folded images and sensitivity maps) from the top module of the system and produces the output according to the SENSE reconstruction principles. The RTL schematic diagram of the proposed SENSE reconstruction is illustrated in Appendix B, which shows the internal circuitry of the SESNE reconstruction module. The top level of SENSE reconstruction module is comprised of matrix inversion module and complex matrix multiplier modules. The matrix inversion module computes the left pseudo inverse of the encoding matrix (as discussed earlier in this section), the detailed internal schematic diagram of the pseudo inversion module is depicted in Appendix C. It comprised of multiple add/sub, integer multiplier and complex number multiplier modules. Moreover, the variable databus

widths were used in intermediate connections of the architecture. The usage of variable bit-width in intermediate data-paths provides an efficient use of the silicon area and reduces the power consumption. Finally, the reconstructed image was transferred through the UART module to the workstation.

The controller unit (as shown in Appendix D) controls the data to be written into specific registers and also deals with the control signals according to the status inputs. The controller unit also ensures to avoid the hazards in the proposed architecture. It also generates the desired memory addresses of the rows and columns of the input matrices to be fed into the system. The control signals enable the permission to write in the specific register by write enable (we) signal. In addition, the uart_en control signal manages the UART data transmission on/off. Furthermore, the status signals such as done and img_tx are activated once the SENSE reconstruction is completed and once the reconstructed image is transferred to the system, respectively.

The proposed design has a latency of 1-clock cycle to compute one sample (set of aliased pixels) of SENSE reconstruction. The total number of clock cycles required for SENSE reconstruction of the image of size ($X \times Y$), with acceleration factor *R*, are:

$$N_{C.C} = \frac{X \times Y}{R} \qquad (3.10)$$

where " $N_{C.C}$ " is the total number of clock cycles required by the proposed design to compute the SENSE reconstruction.

3.2.3 Datasets

The MRI datasets of phantom and human head (axial plane) were acquired using GE MR450, 1.5Tesla (T) MRI scanner at St. Mary's Hospital, London, with Fast Spin Echo sequence and the following parameters: Slice Thickness $3\,mm$, Matrix Size 256×256,

Flip Angle 90°, TR $520\,ms$, TE $15\,ms$, FOV $55\,mm$. Moreover, 3T datasets of axial plane of human head anatomy are acquired using GE Signa HDxt, 3.0T MRI scanner at the University of Malaya Medical Centre (UMMC), Kuala Lumpur. For 3T scans, Fast Spin Echo sequence was used with the following parameters: Slice Thickness $5\,mm$, Matrix Size 256×256, Flip Angle 90°, TR $4660\,ms$, TE $105\,ms$, FOV $24\,cm$.

For both 1.5T datasets, a fully sampled *k*-space was acquired by each of the receiver coils. A reference image was obtained by applying sum-of-squares reconstruction on the acquired multi-coil fully sampled *k*-space data. Under-sampling was performed retrospectively in the phase-encoding direction to simulate the folded images. Aliased images were produced by discarding a specific number of *k*-space lines, depending upon the acceleration factor (*R*), from the fully sampled *k*-space. For *R* = 2, one from every two consecutive *k*-space lines were removed, for *R* = 3, two out of every three consecutive *k*-space lines were skipped. Then the inverse Fast Fourier transform (iFFT) was applied on the sub-sampled *k*-space to generate aliased images. Whereas, for 3T datasets, aliased images were acquired from the scanner by applying a research mode protocol and the reference image was also acquired from the scanner directly. The sensitivity maps were estimated by using the pre-scan method (Pruessmann et al., 1999). The pre-scan method for estimating coil sensitivity maps is discussed in Section 4.2.1.1. The dimensions of the sensitivity map matrix, aliased image (under-sampled) and reference image (fully-sampled) are shown in Table 3.2.

Table 3.2: Datasets dimensions.

		4 Coils	6 Coils	8 Coils
Aliased Image	$R = 2$	$128 \times 256 \times 4$	$128 \times 256 \times 6$	$128 \times 256 \times 8$
	$R = 3$	-----	-----	$85 \times 256 \times 8$
Sensitivity Map		$256 \times 256 \times 4$	$256 \times 256 \times 6$	$256 \times 256 \times 8$
Fully Sampled Image		256×256	256×256	256×256

3.2.4 Quantification Parameters

To quantify the reconstruction performance of the proposed design, Artefact Power (AP) and Signal-to-Noise Ratio (SNR) maps were calculated.

3.2.4.1 Artefact Power (AP)

Artefact power (AP) is based on "Square Difference Error" concept. Mathematically, AP can be defined as (Ji, Son, & Rane, 2007):

$$AP = \frac{\sum \left\| I^{reference}(x,y)\right| - \left| I^{reconstructed}(x,y) \right\|^2}{\sum \left| I^{reference}(x,y)\right|^2} \qquad (3.11)$$

where $I^{reference}$ is the fully sampled (Full FOV) image and $I^{reconstructed}$ represents the reconstructed image. The above formula reveals that AP will be zero, if $I^{reference} = I^{reconstructed}$, which means that the reconstructed image is identical to the reference image. Similarly, if the reconstructed image is significantly different from the reference image, then AP will be a bigger value (i.e., closer to 1).

3.2.4.2 Signal-to-Noise Ratio (SNR) Maps using the pseudo multiple replica method

SNR map was calculated using "Pseudo Multiple Replica" approach. In clinical parallel imaging applications, this technique is universally applicable for a quantitative validation of the reconstructed images. In this method, correctly scaled and correlated synthetic random noise is added to the acquired *k*-space before “black-box” image reconstruction. The same process is repeated 100 (or more) times to produce a stack of independent image replicas with different synthetic noise each time. The noise can be calculated using these replica images by emulating the gold-standard actual multiple replica method (Robson et al., 2008). Mean SNR can be evaluated using the following mathematical expression:

$$SNR_{mean} = \frac{\sum SNR_{map}(x, y)}{X \times Y} \qquad (3.12)$$

where X and Y represent the total number of rows and columns of the image, respectively. The higher values of mean SNR ensure the quality of the reconstructed images.

3.3 Results and Discussion

The proposed architecture design was implemented on FPGA (ML605- Virtex-6, XC6VLX240T). For comprehensive comparison, SENSE code was also implemented in standard C language on multi-core CPU (Core i7, with 2.9 *GHz* clock and 4 GB RAM). Moreover, GPU results were gathered by validating the same datasets on GPU based implementation (Shahzad et al., 2016). NVIDIA GeForce GTX 780 GPU (with 2304 cores and memory of 3 MB) was used for these experiments.

3.3.1 Resource utilization

The proposed architecture code has been synthesized by Xilinx ISE 13.2 software to find the maximum operating frequency and resource utilization of the proposed architecture. The information related to the number of resources, look up tables (LUTs), and DSP slices used by the proposed architecture, was gathered after performing the post place and route procedure on XILINX tool. Table 3.3 shows the details of slice logic utilization of the proposed architecture. The proposed architecture for an acceleration factor "2" utilized less than 1% of the registers, 49% of LUTs and 94% of the available DSP48E1s modules. Whereas, for an acceleration factor "3" the proposed architecture consumed less than 1% of the registers, 99.88% of LUTs and 97% of the available DSP48E1s modules.

Moreover, the proposed parameterized SENSE architecture provides a scalable configuration; therefore, the proposed generic modules can also be reused to implement the SENSE reconstruction system for higher acceleration factors and for a larger number of receiver coils as well. The parameterization feature allows to change the data widths of any module (input, output and/or register) according to the algorithmic needs.

Table 3.3: Device utilization of the proposed architecture.

Device Utilization	Number of utilization	
	***R* = 2**	***R* = 3**
Registers	15	330
LUTs	74769	150538
DSP48E1s	729	746

Furthermore, low power consumption is also one of the main features of the proposed design, because FPGA designs consume significantly lesser power than multi-core CPU or GPU. This characteristic of the proposed design is valuable for the modern upcoming portable MRI scanners where power cost function is crucial (Cooley et al., 2015; Kose & Haishi, 2011; Sarty, 2015; Zotev et al., 2008).

The proposed FPGA based system generates SENSE reconstructed images right on the receiver coil data acquisition system, therefore, the possibility of noise amplification may occur due to the digital system located so closely to the MR receiver coils. However, researches already developed the FPGA based data acquisition systems for multi-channel receiver coils in MRI (Gebhardt et al., 2016; Tang, Sun, & Wang, 2012). These digital receiver systems are capable to work near the MRI systems and acquired the multi-channel receiver coil data without noise interference. Also in MRI systems, electromagnetic interference (EMI) shielding techniques are also used to suppress the unwanted signals generated by other sources (Weissler et al., 2014; Yamamoto et al., 2011). Thus, it proves that FPGA based systems are MRI-compatible. Hence, the proposed FPGA system for real-time SENSE reconstruction can be easily equipped with the digital receiver systems, which are especially designed for direct data acquisition of MRI data to operate without noise interference. In addition, different digital filters are also used to remove the unwanted signal or noise.

3.3.2 Computational time analysis

The proposed architecture has been tested using 1.5T and 3.0T MRI datasets. The computation time required for SENSE reconstruction by the proposed design is shown in Table 3.4. The computation time was calculated according to the maximum operating frequency. The proposed architecture, when operating at the maximum frequency,

consumes $0.054\,ms$ and $0.049\,ms$ for SENSE reconstruction of acceleration factor 2 and 3 datasets, respectively.

Table 3.4: Computation time of the proposed architecture.

Acceleration Factor	Maximum Operating Frequency (*MHz*)	Computation Time (*ms*)
$R = 2$	604.778	0.054
$R = 3$	445.61	0.049

The proposed system reconstruction time is lesser than the MRI data acquisition time, which provides advantages in many ways. In MRI, the benefits of faster-than-acquisition reconstructions are: (1) MRI radiographer can get immediate feedback which allows him/her to detect any imaging problem on run-time. That way, if MRI radiographer finds any problem during the scan, then the same subject could be re-scanned with the correct protocol at the spot. This strategy will improve the efficiency of the scanner. (2) Run-time error detection in MRI scan would also improve the diagnostic efficiency and will decrease the cost of the scan. (3) Computational resources would be less utilized due to faster reconstruction, which eventually reduces the power consumption of the hospital resources. It allows to accommodate more patients in a given time.

The performance comparison of SENSE reconstruction in terms of computation time for FPGA, GPU and multi-core CPU is shown in Table 3.5. The results show that the computation time of the proposed FPGA implementation for SENSE is remarkably smaller (approximately eight hundred fifty times better) than the software based reconstruction on multi-core CPU and almost 52 times faster than the GPU based implementation, when operated at 600 *MHz* frequency. Moreover, FPGA based

proposed design also provides promising results when operating at a lower frequency (i.e. 200 *MHz*), and achieves lesser computational time than the implementation on the other platforms (i.e. GPU or Multi-core CPU). This is because the proposed design is hardware based, whereas CPU implementation is purely a software based method. In GPU implementations of SENSE (Hansen et al., 2008; Shahzad et al., 2016), MRI raw data is stored in the CPU memory because GPU does not have enough memory space to store all the MRI raw data in its memories. Therefore, the data transferring overhead originates in GPU case and this takes a majority of the computation time. So the total SENSE reconstruction time is composed of: (1) the kernel execution time in which SENSE algorithm operations are done (2) data transfer overhead time in which data is transferred to the device (GPU). In Table 3.5, both times are shown for better comparison.

Multi-core CPU and GPU computation times increase as the number of receiver coils are increased. However, the results show that the proposed FPGA based design consumes the same time for any number of receiver coils (8 or less) because it has been optimised for 8 receiver coils. The graphical representation of the computation time of the GPU and FPGA implementations is illustrated in Figure 3.4.

Table 3.5: Computation time comparison between multi-core CPU, GPU and the proposed architecture reconstruction.

Number of receiver coils (with $R = 2$)	SENSE computation time (*ms*)								
	Multi-core CPU	GPU						FPGA	
		Kernel and data transfer			Kernel Only				
	(Shahzad et al., 2016)	(Hansen et al., 2008)	(Shahzad et al., 2016)	(Shahzad et al., 2016)[2]	(Hansen et al., 2008)	(Shahzad et al., 2016)	(Shahzad et al., 2016)[2]	@ 600 *MHz*	@ 200 M*Hz*
4	18	1.8	2.6	1.6	0.56	1.0	0.7	0.055	0.164
6	31	2.2	3.1	2.3	0.85	1.4	1.0	0.055	0.164
8	47	3.5	4.7	2.9	1.13	1.7	1.2	0.055	0.164

[2] The same method with modern GPU (GTX 780)

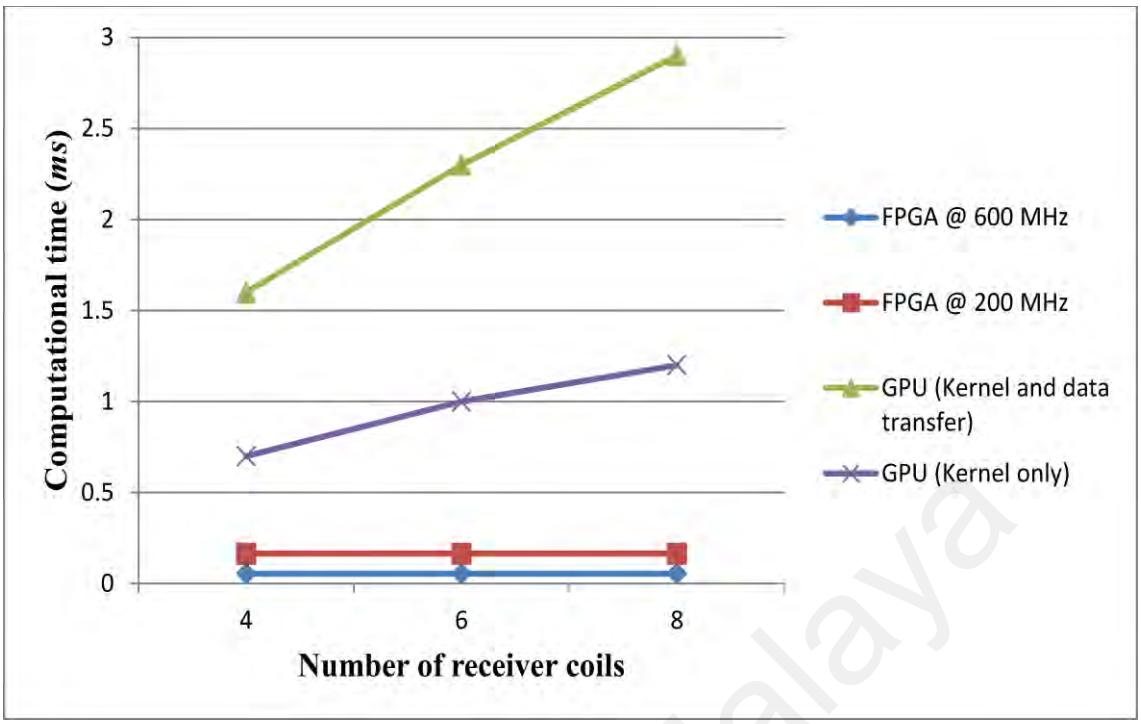


Figure 3.4: Computational time comparison between FPGA and GPU for different number of receiver coils.

In (Shahzad et al., 2016), the proposed GPU implementation consumed approximately $4.7\,ms$ to reconstruct the images using SENSE algorithm for a dataset of 256×256 matrix, 8 receiver coils and $R=2$, that is 85 times slower than the FPGA reconstruction performed in this thesis. Furthermore, the proposed FPGA implementation performance compare with GPU implementation (Hansen et al., 2008) is increased by 63 in terms of speed-up factor. The MR reconstruction time claimed in (Li & Wyrwicz, 2015) is $2.5\,ms$ for 128×128 fully sampled parallel MRI dataset. However, this time will increase almost to double for 256×256 matrix size image. Moreover, parallel MRI is a more advanced technique than the conventional (2DFFT) MR reconstruction. The acquisition time of the parallel MRI (with acceleration factor = 2) is lesser than the non-accelerated parallel MRI scan; therefore, the total scan time (acquisition + reconstruction) consumed by the proposed FPGA implementation is significantly smaller than the method proposed in (Li & Wyrwicz, 2015).

3.3.3 Image Reconstruction

The image reconstruction from the under-sampled data (1.5T and 3.0T datasets) was performed using the proposed FPGA system. Figure 3.5 and Figure 3.6 show the reconstructed images with the proposed hardware design. 1.5T datasets reconstructed images with their artefact power for acceleration factor 2 and 3 are illustrated in Figure 3.5. Whereas, Figure 3.6 (a) and Figure 3.6 (b) show the reconstructed images for 3.0T datasets with their artefact power for acceleration factor 2 and 3, respectively. Artefact power is used as a quantification parameter for evaluating the quality of the reconstructed images (Ji et al., 2007). Smaller values of artefact power indicate a reconstructed image with better quality. The results show that the proposed FPGA system produces reconstructed images with significantly less artefact power. The artefact power of the FPGA reconstructed images is $2.449\times10^{-4}/4\times10^{-3}$, $4.5465\times10^{-4}/4.5\times10^{-3}$, $1.6545\times10^{-4}/8.5\times10^{-3}$ and $3.0917\times10^{-4}/1.3\times10^{-2}$ (Acceleration factor of 2 / Acceleration factor of 3) for phantom, 1.5T human head, 3T human head_1 and 3T human head_2 datasets, respectively. The results show that the quality of the reconstructed images (when $R = 2$) is better than the reconstructed images (when $R = 3$).

The reconstructed images prove that the proposed FPGA based reconstruction, for an acceleration factor of 2, produces artefact free images. The visual assessment of $300\times$ magnified portion of the reconstructed images reveals that the proposed FPGA system generates high resolution images. However, the reconstruction results for the case of acceleration factor = 3; having slightly less resolution. This is because the eight channel receiver coil array data is used for these experiments and when a greater under-sampling is applied then it might be difficult to have good reconstruction results because of greater under-sampling.

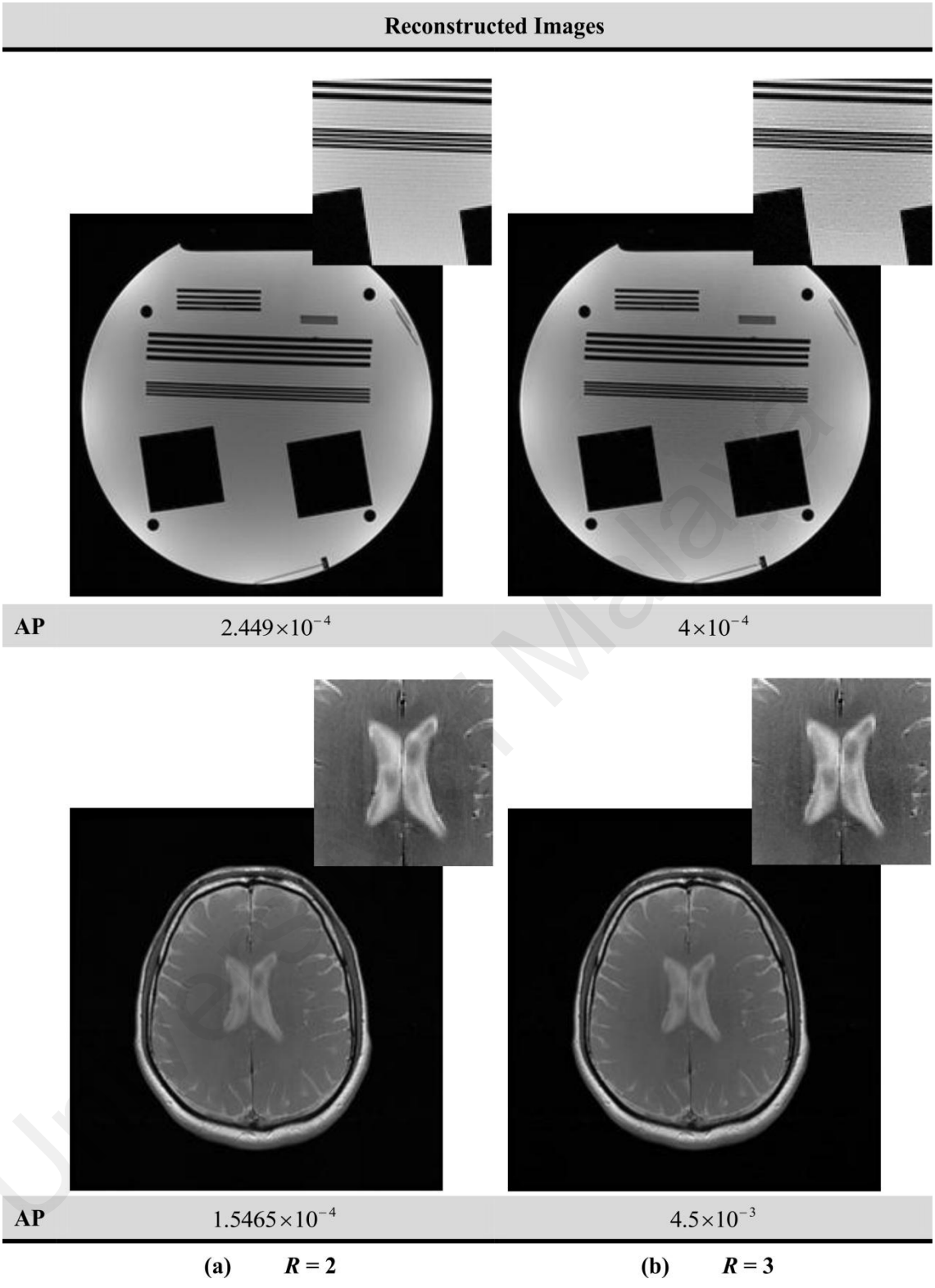


Figure 3.5: Reconstructed Images (1.5T) with Artefact Power (AP): (a) Acceleration Factor, $R = 2$ (b) Acceleration Factor, $R = 3$.

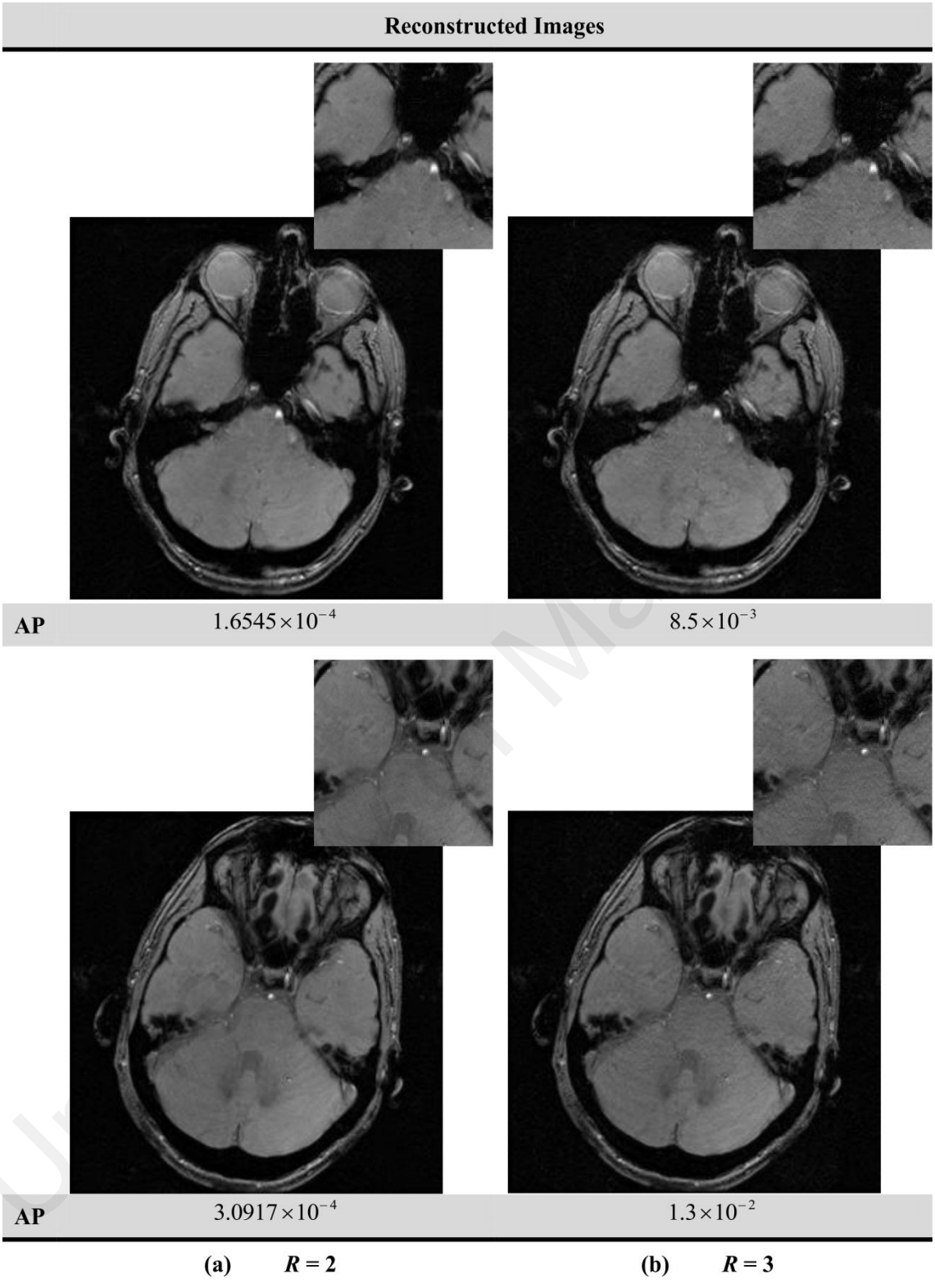


Figure 3.6: Reconstructed Images (3.0T) with Artefact Power (AP): (a) Acceleration Factor, $R = 2$ (b) Acceleration Factor, $R = 3$

3.3.3.1 Comparison of the Reconstructed Images

Figure ß3.7 illustrates the images reconstructed using the multi-core CPU, FPGA (proposed hardware design) and GPU for different number of receiver coils (4, 6 and 8). Figure ß3.7 (a) presents the reference image (fully-sampled image), Figure ß3.7 (b) shows the multi-core CPU reconstructed images, Figure ß3.7 (c) and Figure ß3.7 (d) illustrate the FPGA and GPU reconstructed images, respectively. The artefact power of each reconstructed image is also mentioned in Figure ß3.7.

The results show that the FPGA reconstructed images (from the under-sampled data) are almost identical to the original images with low artefact power values, i.e., 3.5×10^{-4}, 2.46×10^{-4} and 2.45×10^{-4} for 4, 6 and 8 receiver coils phantom datasets, respectively. Similarly, the artefact powers for human brain datasets are: 1.73×10^{-4}, 1.66×10^{-4} and 1.55×10^{-4} for 4, 6, and 8 receiver coils, respectively. The GPU reconstructed images are also almost identical to the original images. Artefact power of the GPU reconstructed images are in the acceptable range with values: 2.79×10^{-4}/ 1.75×10^{-4}, $2.44\times10^{-4}/1.64\times10^{-4}$ and $2.40\times10^{-4}/1.52\times10^{-4}$ (Phantom/Brain) for 4, 6 and 8 receiver coils datasets, respectively. Similarly, the artefact values for multi-core CPU reconstructed images are: $2.46\times10^{-4}/1.79\times10^{-4}$, $2.40\times10^{-4}/1.63\times10^{-4}$ and $2.40\times10^{-4}/1.52\times10^{-4}$ (Phantom/Brain) for 4, 6 and 8 receiver coils datasets, respectively. The artefact power of the proposed architectural design reconstruction is almost similar to other platforms (CPU or GPU) reconstruction; and gives satisfactory results for an acceptable artefact power, as shown by Figure ß3.7.

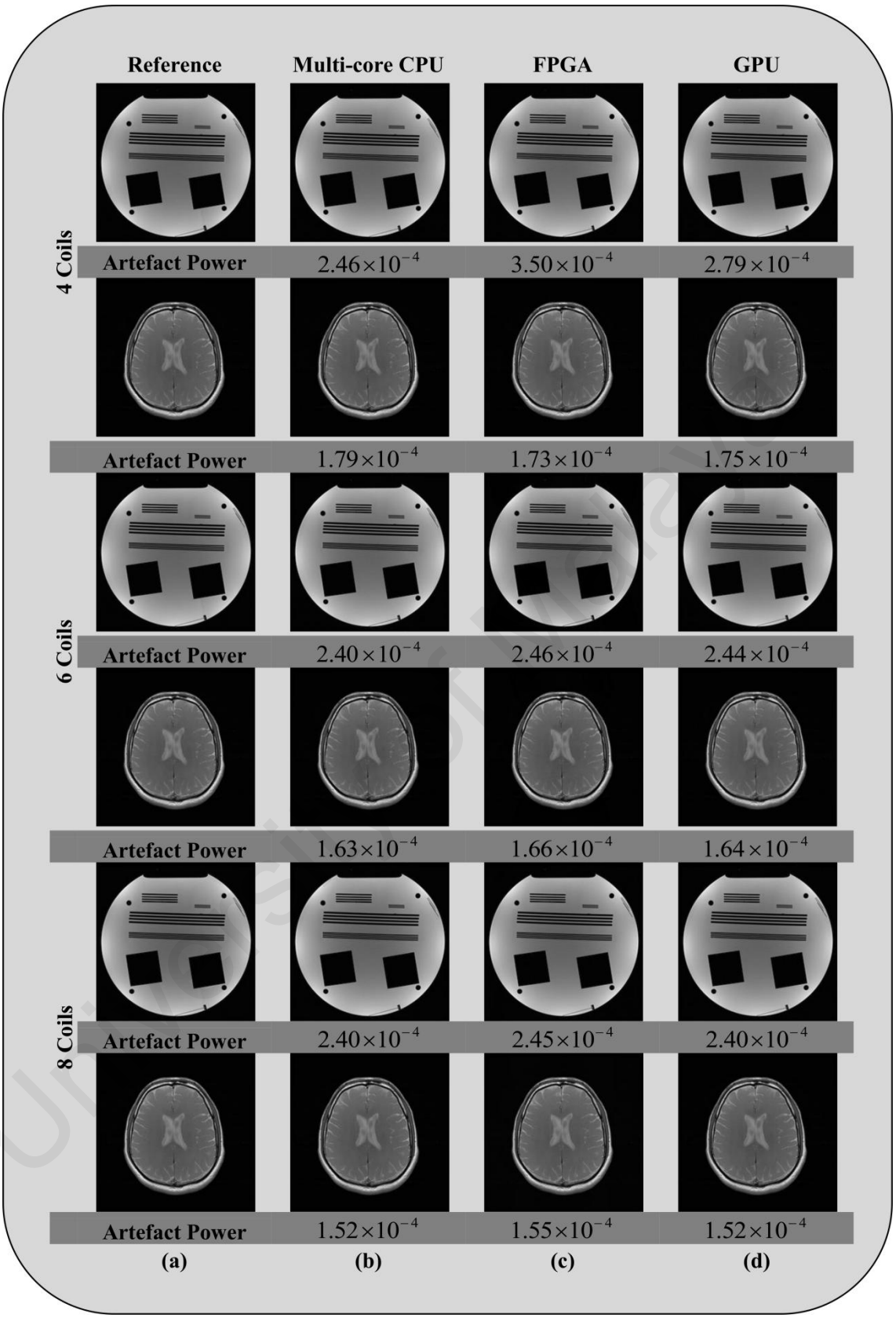


Figure 3.7: Reference images and reconstructed images with their artefact power values (a) Reference image (b) Multi-core CPU reconstructed image (c) Proposed design reconstructed image (d) GPU reconstructed image.

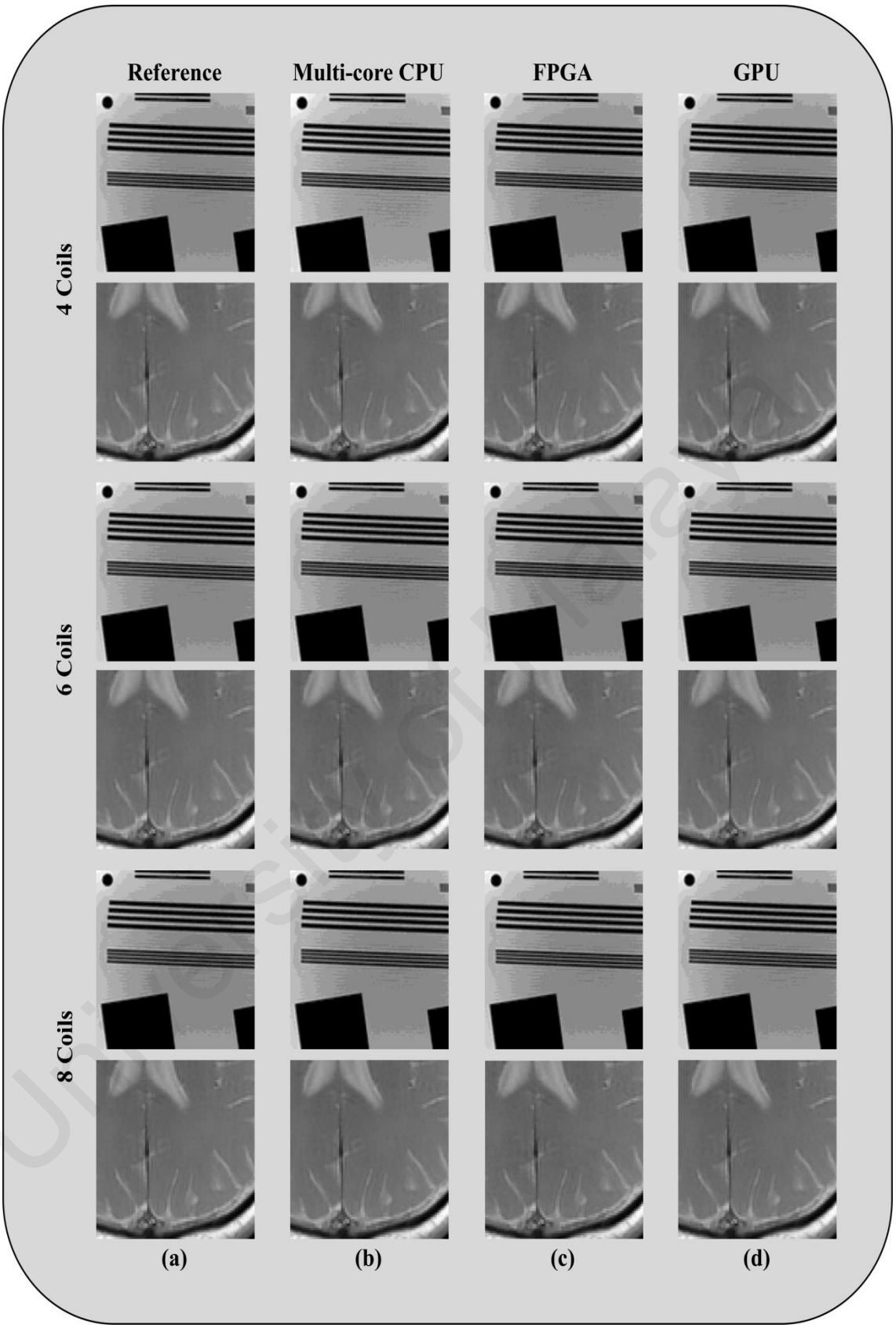


Figure 3.8: Magnified section of (a) Reference image (b) Multi-core CPU reconstructed image (c) Proposed design reconstructed image (d) GPU reconstructed image.

Figure 3.8 shows the magnified ($300\times$) views of the resultant images and the reference images for better comparison. The clarity of the magnified reconstructed images as compared to the reference magnified images shows a remarkable efficiency of the proposed hardware design.

3.3.4 Qualitative Evaluation

For qualitative evaluation, mean SNR of the solution images obtained from the proposed FPGA design is evaluated using pseudo multiple replica method (Robson et al., 2008). Figure 3.9 shows the SNR maps with mean SNR of the reconstructed images (obtained by the proposed FPGA implementation) for different acceleration factors.

The reconstructed images for acceleration factor 2 (Figure 3.9 (a)) exhibit no significant apparent noise in the reconstructed images with mean SNR of 36.8371 dB, 31.9605 dB, 36.8575 dB and 36.0844 dB for phantom, human head, 3.0T human head_1 and 3.0T human head_2 datasets, respectively. The reconstruction results (for an acceleration factor of 3) of the phantom and human head axial datasets in terms of mean SNR are 31.5365 dB, 26.684 dB, 33.5638 dB and 33.8102 dB, respectively. The proposed FPGA implementation achieves greater than 30 dB SNR on average for different acceleration factors which validate the accuracy of the proposed system. The measured SNR of the reconstructed images for $R = 2$, is slightly better than the SNR of $R = 3$ reconstructed images. This is because SNR is inversely proportional to the square root of the acceleration factor "R" (Larkman & Nunes, 2007), therefore, when acceleration factor increases it decreases the SNR of the reconstructed images in PI.

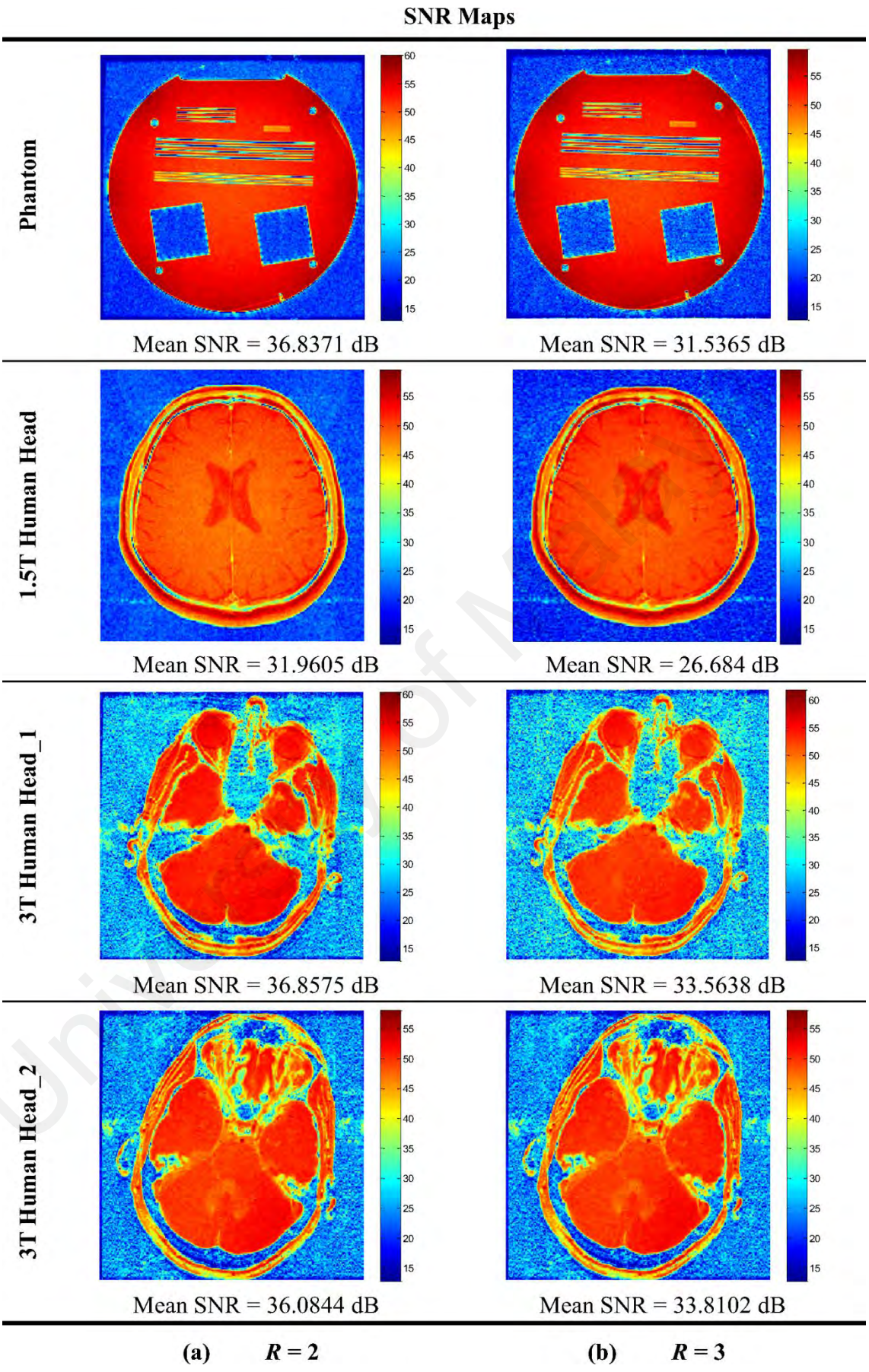


Figure 3.9: Pseudo multiple replica based SNR maps of the reconstructed images with mean SNR (a) Acceleration Factor, $R = 2$ (b) Acceleration Factor, $R = 3$.

3.3.4.1 Qualitative Comparison with Different Platform Reconstructions

Figure ß3.10 and Figure ß3.11 show the SNR maps with the mean SNR values of the reconstructed images for the phantom and human head data, respectively. The mean SNR values of the reconstructed phantom images are more than 34 dB in each case. Moreover, for human head reconstructed images it is greater than 28 dB for 4 receiver coils while it improves to 30+ dB for 6 and 8 receiver coils in the multi-core CPU and FPGA implementations. The results show that the proposed architecture, multi-core CPU and GPU implementation have achieved almost 30+ dB mean SNR values in the reconstructed images. This proves that the proposed architecture produces high quality reconstructed images with good SNR values similar to multi-core CPU or GPU implementations.

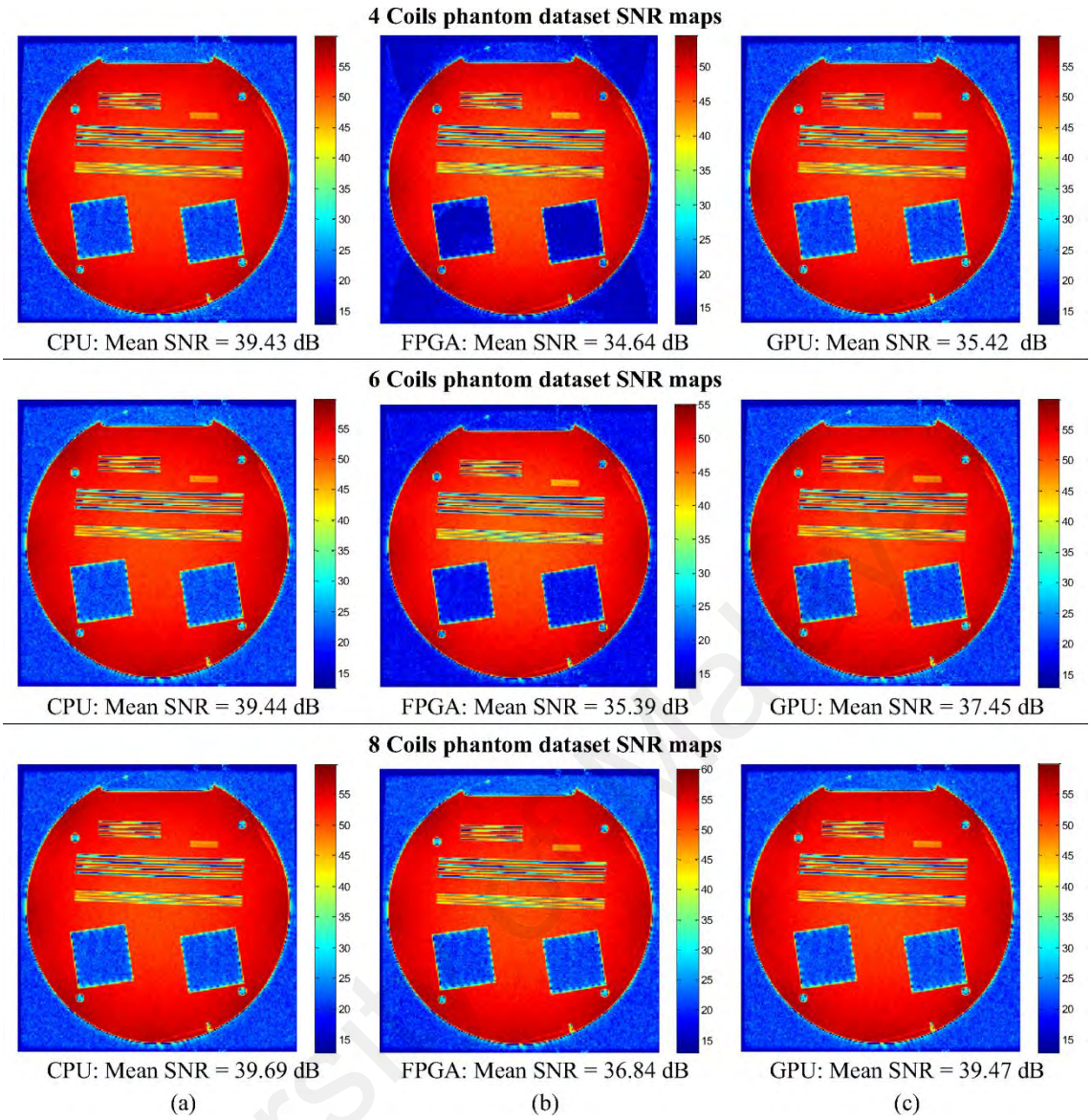


Figure 3.10: Pseudo multiple replica based SNR maps with mean SNR values of phantom reconstructed images (a) Multi-core CPU reconstructed image (b) Proposed design reconstructed image (c) GPU reconstructed image.

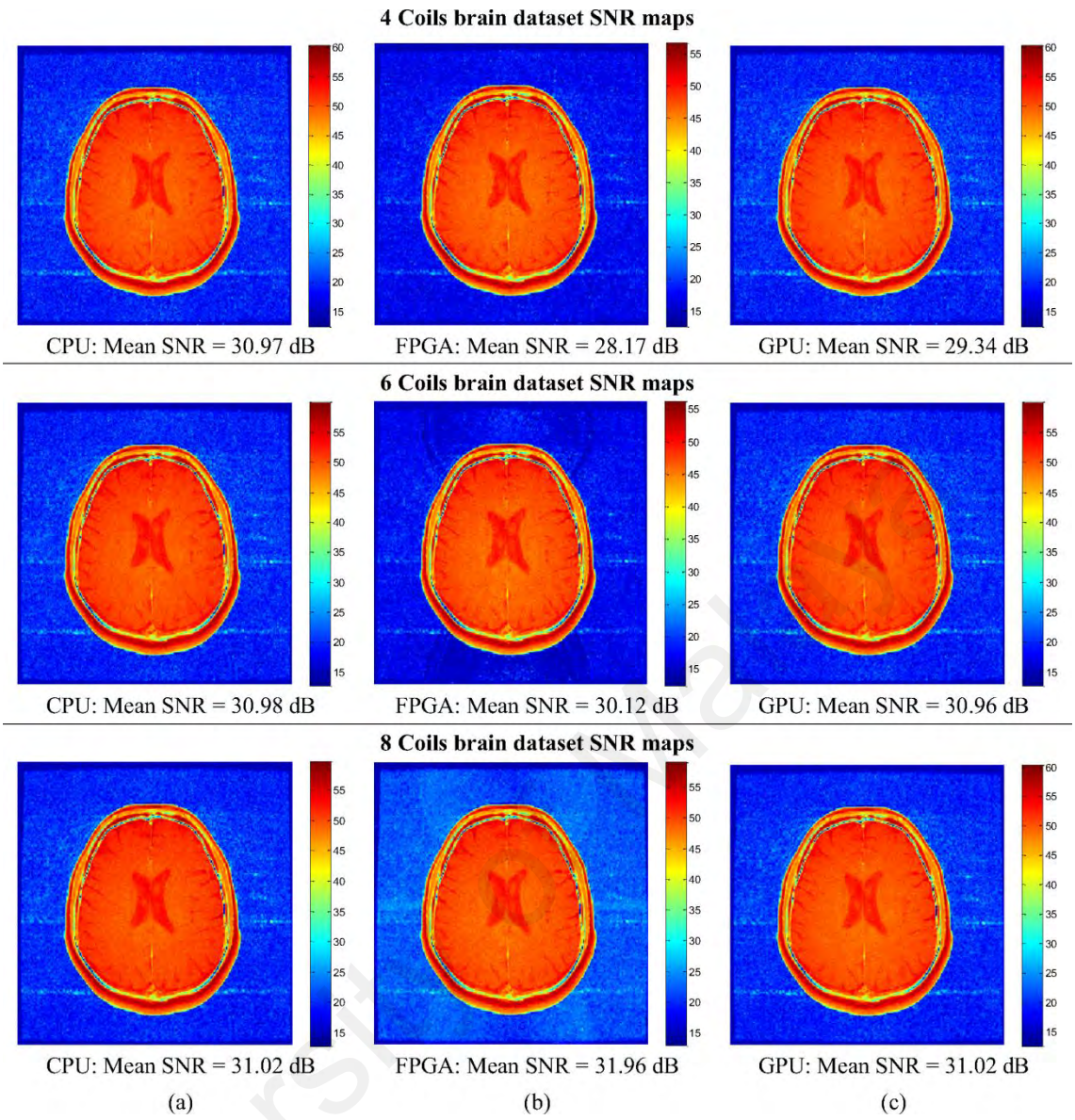


Figure 3.11: Pseudo multiple replica based SNR maps with mean SNR values of human brain reconstructed images (a) Multi-core CPU reconstructed image (b) Proposed design reconstructed image (c) GPU reconstructed image.

3.3.5 Efficient Memory Usage and Reducing Transmission Cost

In this research, real-time parameterized SENSE reconstruction architecture is proposed. Conventionally, in MRI scanners all the acquired data is transmitted to the workstation and then reconstruction algorithms are applied on the multi-core CPU, GPU or CPU+GPU platforms (Dalal & Fontaine, 2006; Eklund et al., 2013; Hansen et al., 2008; Hansen & Sørensen, 2013; Omer & Dickinson, 2010; Pratx & Xing, 2011; Saybasili et al., 2014; Stone et al., 2008; Wang et al., 2010). There are many recent techniques proposed for accelerating MRI. However, all these techniques deal with the data once it has been transmitted to the workstation. The raw data of multiple receiver coils in parallel MRI require hundreds of Megabytes of memory for one scan and all the data need to be transmitted to the workstation and stored in the memory cabinets. The proposed design offers a different approach to accelerating MRI, i.e., a compact design which can be equipped with the receiver coil data acquisition system, thus there is no need to transmit all the acquired raw data to the control room, which will improve the signal strength and SNR. A real-time reconstruction using the proposed design (without the need to transmit the acquired data to the workstation) provides significant advantages of not requiring to store the multiple receiver coils raw data in the memory cabinets located on the workstation of the MRI system. The proposed architecture can be integrated with the receiver coil data acquisition system itself, so only the reconstructed image needs to be stored in the memory cabinets of the workstation and it consumes very less memory (typically 128KB for a 256×256 matrix size image). Furthermore, the reconstruction in the proposed architecture will not experience interference of transmission losses because the image reconstruction from all the acquired data is performed right on the receiver coil data acquisition system, thereby minimizing the signal transmission losses and improving the quality of the reconstructed image.

Table 3.6: Transmission / Memory usage improvement of the proposed system.

	Conventional MRI Scanner Method		Proposed System Method		Difference [3] (KB)
	Memory (KB)	Transmission (KB)	Memory (KB)	Transmission (KB)	
Low Resolution Images (Calibration)	0	128	0	0	0 / -128
Sensitivity Maps	2048	0	0	0	-2048 / 0
Folded Images	1024	1024	0	0	-1024 / -1024
Reconstructed Image (absolute)	128	0	128	128	0 / 128
Total	3200	1152	128	128	-3072 / -1024

The proposed method achieves significant improvements over conventional image reconstruction on the MRI scanner. It leads to reduce the amount of data transfer (from the MRI scanner to the workstation) as compared to the current method used in MRI scanners. The results show that the proposed system manages memory efficiently than the conventional method (multi-core CPU or GPU reconstruction) used in MRI scanners and decreases the memory usage up to 2048 KB and 1024 KB in storing sensitivity maps and folded images respectively when number of receiver coils = 8, Image Size = 256×256, number of bits used to represent the intensity of each pixel = 16 and R = 2, and the calibration scan low resolution image (used for sensitivity maps estimation) was of size = $64 \times 64 \times 8$. Table 3.6 shows the transmission and memory usage

[3] The difference is calculated as: the difference between the conventional MRI scanner memory usage and the proposed system memory usage / the conventional MRI scanner transmission cost – the proposed system transmission cost

improvement of the proposed system over conventional methods used in MRI scanner for reconstruction.

3.4 Summary

The computation time and image resolution are major concerns in MR imaging. This chapter presents a scalable FPGA implementation of run-time SENSE reconstruction which is shown to reduce the computation time, memory usage, transmission cost while maintaining the necessary high temporal resolutions in the reconstructed images.

An eight channel receiver coil system was used to acquire the datasets on 1.5T and 3.0T MRI scanners. The results demonstrate good reconstruction quality as depicted by significantly smaller values of artefact power ($< 2.45 \times 10^{-4}$) and good mean SNR close to 32+ dB when acceleration factor is set to 2. The results show a slight degradation in the quality of the reconstructed images for acceleration factor = 3, however, measured mean SNR (31+ dB) and artefact power ($< 7.5 \times 10^{-3}$) are still good. A visual inspection of the reconstructed images ($300\times$ magnified) also exhibits the quality and better resolution of the reconstruction results. The results also suggest that the proposed system achieves improvement up to 95.8% on memory usage and reduces the transmission cost up to 87.5%. Real-time image reconstruction by the proposed design also benefices the radiographer to find out the discrepancies during the scan. Furthermore, it has also been demonstrated that the proposed system works as efficiently as other platforms used conventionally in MRI scanners for image reconstruction (e.g. CPU or GPU). The proposed architecture provides a prototype solution for the modern portable MRI systems.

The main objective of this work is to provide a hardware design for real-time SENSE reconstruction (which can be embedded right on the receiver coil data acquisition

system) in MRI. This research presented an application specific parameterized architectural design for SENSE algorithm for the acceleration factor of 2 and 3.

CHAPTER 4: USING DIFFERENT SENSITIVITY MAPS FOR REAL-TIME SENSE RECONSTRUCTION

4.1 Introduction

Magnetic resonance imaging (MRI) is an advanced imaging modality commonly used in medical clinical practice to provide the information of the internal conditions of the tissues. The MRI machine uses radio waves and strong magnetic fields to create detailed images of the body. However, MRI requires relatively long data acquisition time conventionally, which challenges its utilization for some applications such as: breath-hold scans and makes it susceptible to patient movement. Long scan time in MRI also increases the on-time of the MRI system, which eventually increases the hospitals' resource usage and power consumption. MRI scan time can be reduced by increasing the speed of data acquisition sequence and/or acquiring fewer samples.

SENSE (SENSitivity Encoding) (Pruessmann et al., 1999), is one of the most clinically used parallel imaging technique, which utilizes the knowledge of the receiver coil sensitivity profiles to calculate the aliased signal component at each point in the acquired image and then places those aliased components back to their original positions to get the full resolution image. In SENSE, sensitivity profiles play a major role to allow the optimal image reconstruction. However, it is not often easy to estimate the receiver coil sensitivities accurately and even small errors in the sensitivity profiles generate artefacts in the reconstructed images. Many solutions have been explored in the recent past to estimate the sensitivity maps which can be divided into two categories (1) using pre-scan method (Bydder, Larkman, & Hajnal, 2002a; Zwart, Gelderen, Kellman, & Duyn, 2002; Pruessmann et al., 1999; Walsh, Gmitro, & Marcellin, 2000) (2) auto-calibration method using *k*-space center lines (McKenzie, Yeh, Ohliger, Price,

& Sodickson, 2002; Uecker, Hohage, Block, & Frahm, 2008; Uecker et al., 2014; Ying & Sheng, 2007).

In pre-scan method, an additional quick scan (also called calibration scan) right before the actual scan is acquired to capture low resolution images from each receiver coil. These low resolution images are used to estimate the sensitivity maps in this method. The sensitivity maps estimated by pre-scan is totally depends upon the calibration scan, therefore, patient's movement in between the different experiments may generate motion artefacts. Whereas, auto-calibration method uses the additional calibration lines which are integrated in the actual scan instead of acquiring additional calibration scan. This technique provides direct sensitivity calibration for each individual scanned image which is benefices for those applications where coils are flexible or patients are not cooperative. Eigenvector maps based coil sensitivity maps estimation is an auto-calibration based technique. In this method, the sensitivity maps are estimated by exploiting the main eigenvectors of the null space of a calibration data. In this thesis, a novel FPGA implementation of real-time SENSE reconstruction is presented, which can work equally efficiently for pre-scan based sensitivity maps and eigenvector maps (referred as E-maps in this thesis).

4.2 Methods and Materials

FPGA (ML605- Virtex-6, XC6VLX240T) platform was used to evaluate the experiments. The input data matrices (Sensitivity maps and the aliased images) were stored in the memory of the FPGA. The FPGA implementation of real-time SENSE reconstruction described in CHAPTER 3 was used to perform the reconstruction using different sensitivity maps. The SENSE architecture module performs the reconstruction operation on the given input (fetching data from memory) and then the reconstructed

image (output) was transferred to MATLAB (R2013a) via UART transmission for analysis and comparison.

4.2.1 Sensitivity Maps Estimation

In SENSE, a successful reconstruction of the MR image is strongly associated with a precise knowledge of the receiver coil sensitivities. Referring to equation (3.8); an accurate encoding matrix is the fundamental component of SENSE algorithm. A particular coil contributes more signal strength from the areas of the object (being scanned) which are closer to that coil compare to the signal which are far from that specific coil. In the aliased image, each pixel contains signals from more than one actual pixel locations so the contribution of each pixel in the aliased location should be accurately known. The information of the encoding matrix (subset of sensitivity maps) provides weights to the aliased signals; therefore, precise estimation of receiver coil sensitivity maps is a crucial step in PI.

The coil sensitivity profile varies from patient to patient and different for different body parts. Therefore, the calibration of the receiver coils is needed for each subject examination. The preferable sensitivity profile estimation (coil calibration) technique should be intelligent (which can work without any assistance), reliable, fast and robust. Different coil sensitivity maps estimation methods are found which are clinically used in the MRI scanners. Each solution has advantages and dis-advantages which depends upon the nature of the specific application. Two well known methods to estimate the receiver coil sensitivities are used in this thesis: (1) pre-scan method (2) eigenvalue decomposition method.

4.2.1.1 Pre-Scan Method

The receiver coil sensitivity maps are generated for each receiver coil element from a short and low resolution calibration scan (pre-scan) which is usually performed at the beginning of the MRI examination. Many solutions have been given in the literature to estimate the sensitivity maps (Bydder et al., 2002a; Zwart et al., 2002; Pruessmann et al., 1999; Walsh et al., 2000) from this low resolution image. In this thesis, the method proposed in (Zwart et al., 2002; Pruessmann et al., 1999) is used with the following steps (as shown in Figure 4.1).

In separate calibration scan, the low resolution image "*K*" was acquired from each receiver coils with the dimension of $X_K \times Y_K \times N_C$: $64 \times 64 \times 8$. The acquired low resolution image was used to estimate the receiver coil sensitivity maps. Zero-padding was done on "*K*" to make full field-of-view low resolution image. Inverse Fast Fourier Transform (iFFT) was applied on the zero-padded *k*-space data to obtain low resolution image "*L*" of size $X \times Y \times N_C$. The values of "*X*" and "*Y*" depend upon the matrix size of the image during the scan. The low resolution image "*L*" contains spatially varying coil sensitivity knowledge with anatomical information. Therefore, it cannot be used as a sensitivity maps because of the anatomical details.

Different methods have been proposed to remove these anatomical details from the low resolution images. Sensitivity profiles can be estimated by any of these methods: (1) Dividing each receiver coil low resolution magnitude image "L_M (n)" by the body coil image (Pruessmann et al., 1999) (2) Dividing "L_M (n)" by sum-of-squares image (de Zwart et al., 2002) (3) By using relative sensitivity maps method: Dividing "L_M (n)" by anyone of the receiver coil image (Bydder et al., 2002a).

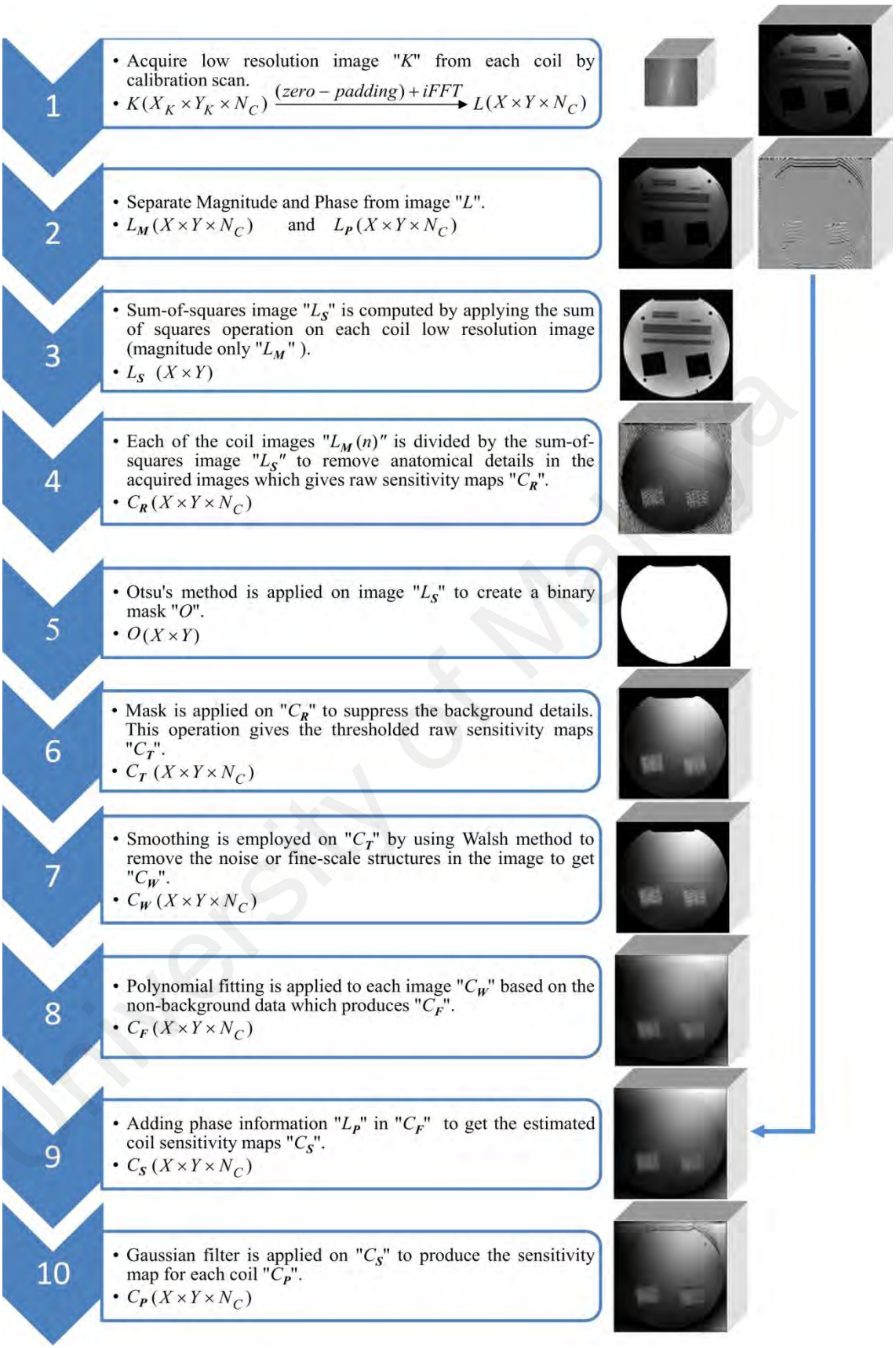


Figure 4.1: Flow diagram of estimating the receiver coil sensitivity maps using pre-scan method.

This research used sum-of-squares method. Sum-of-squares operation was computed on each of the receiver coil low resolution magnitude image "L_M (n)" to get the sum-of-squares image "L_S". Then each of the coil images "L_M (n)" was divided by the sum-of-squares image "L_S" to remove anatomical details in the acquired images which estimates the raw sensitivity maps "C_R". The dimensions of the "C_R" will be $X \times Y \times N_C$.

The raw sensitivity maps "C_R" were refined by some additional numerical processes. In the first step, Otsu's method (Fan & Zhao, 2007) was applied on the sum-of-squares image "L_S" for creating a binary 2-D mask "O". The computed 2-D mask "O" was multiplied by each of the receiver coil raw sensitivity maps "C_R (n)" to remove the background details which produces thresholded raw sensitivity maps "C_T". Furthermore, some more steps were involved to minimizing the propagation of additional noise from the calibration scan into the reconstructed image. Smoothing was applied on "C_T" by using Walsh method (Walsh et al., 2000) to remove the noise or fine-scale structures in the thresholded raw sensitivity maps to get "C_W". In the next step, polynomial fitting was applied to each image "C_W (n)" based on the non-background data which produces spatially varying sensitivity maps "C_F". Adding the phase information back into the "$C_{F''}$ in order to acquire estimated coil sensitivity maps "C_S". Finally, the Gaussian smoothing filter was applied on "C_S" to produce the pre-scan sensitivity maps "C_P".

4.2.1.2 Eigenvalue Decomposition Method

Eigenvalue decomposition method (Uecker et al., 2014) is an advanced and implicit technique of estimating receiver sensitivity maps. It is a process that determines the subspace of the multi-coil data from a calibration region. The following procedure (as

depicted in Figure 4.2) was used to estimate sensitivity maps (E-maps) by eigenvalue decomposition method.

The calibration data "B" was gathered from the centre of the k-space of each of the receiver coil image. The dimension of the cropped data "B" was $X_K \times Y_K \times N_C$, where the values of X_K and Y_K depend upon the number of calibration lines (rows and columns) were used to compute the sensitivity maps. The calibration matrix "A" was computed by moving a sliding window also known as the kernel, horizontally, pixel-by-pixel on the calibration data "*B*". In each iteration, one column of "*A*" was filled by the selective portion of the calibration data from each receiver coil according to the kernel. This process generated a 2-D matrix that leads to a matrix structure known as Block-Hankel (Uecker et al., 2014). Singular Value Decomposition (SVD) was applied on the calibration matrix "*A*" to get the singular values "*S*" and singular vectors. Singular vectors were reshaped according to the kernel size to produce *k*-space kernels matrix. The *k*-space kernel matrix "*k*" corresponds to the basis vectors of overlapping blocks in *k*-space. Thresholding was applied on "*S*" to explore the range of the principal components (which majorly contribute) and the maximum number of principal components selected for kernel matrix "*k*" was represented by "*idx*". Then the null vectors are ignored from kernel matrix "*k*" and only the singular vectors "*S*" were selected according to the value of "*idx*". This process will crop the kernel matrix "*k*" dimension to $(X_K \times Y_K \times N_C \times idx)$.

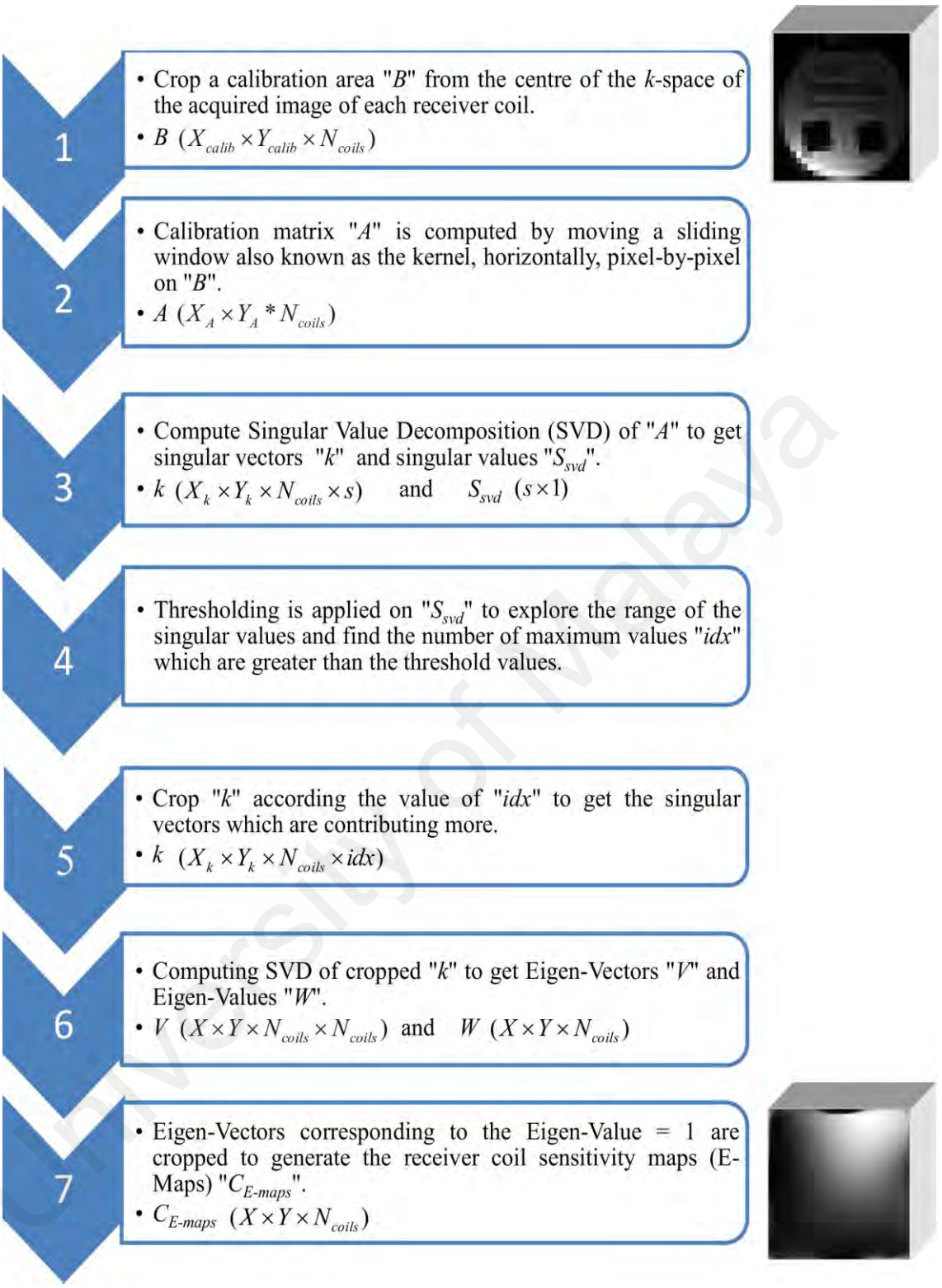


Figure 4.2: Flow diagram of estimating the receiver coil sensitivity maps using eigenvalue decomposition method.

Again SVD was applied on the cropped kernel matrix "*k*" to get the eigenvectors "*V*" and eigenvalues "*W*". Eigenvector corresponding to the eigenvalue = 1 were cropped to generate the receiver coil sensitivity maps "C_E" (E-maps). Furthermore, a kernel size of 6×6 has been used in this research with a calibration region of size 24×24 to evaluate the sensitivity maps by eigenvalue decomposition.

4.2.2 Datasets

The MRI datasets (same as CHAPTER 3) were used to evaluate the results of this chapter. Moreover, a separate quick calibration scan was acquired which is needed to estimate the sensitivity maps in the pre-scan method. Low-resolution full FOV images from each receiver coil were obtained. The calibration scan was acquired with 2D fast gradient echo with low spatial resolution. The scan parameters were fixed except the slice thickness. The quick scan (calibration scan) consumes approximately 6 to 12 seconds and it is fast enough to allow breath-holding for body scanning.

4.2.3 Quality Assessment Parameters

The reconstructed images using pre-scan and e-maps sensitivity maps are compared with the reference images (fully sampled images from the scanner) for visual inspection. Furthermore, Artefact Power (AP) (Ji et al., 2007), line profile analysis and signal-to-noise-ratio (SNR) maps (Robson et al., 2008) are also used for validation of the proposed design performance. AP is the square difference error between the reconstructed image and the reference image. Line profile is a quantifying parameter to analyze the quality of the reconstructed images. The central line of both the reconstructed and reference images is normalized and then plotted. Similar line profiles ensure better reconstruction. Moreover, "Pseudo Multiple Replica" method (Robson et al., 2008), has been used to calculate SNR maps. Pseudo multiple replica method is

universally applicable for quantitative validation of the reconstructed images in clinical parallel imaging applications.

4.3 Results and Discussion

The proposed FPGA implementation of SENSE using pre-scan and E-maps based sensitivity maps was performed on ML605 Virtex-6 evaluation kit. The final output (reconstructed image by the proposed design) was transferred from FPGA to MATLAB for quality assessment purposes. To validate the efficiency and accuracy of the proposed architecture, phantom and human head datasets were used. The dimensions of the encoding matrix, aliased image (folded image) and the reference image (fully-sampled), for the acceleration factor (R = 2) and 8 receiver coils were: $256 \times 256 \times 8$, $128 \times 256 \times 8$ and 256×256, respectively.

4.3.1 SENSE Reconstruction using Different Sensitivity Profiles

Figure ‎4.3 shows the receiver coil sensitivity maps (all eight channels) obtained by pre-scan and E-maps method. Each receiver coil sensitivity map shows a localized sensitivity. First two rows of the Figure ‎4.3 shows the sensitivity maps computed by pre-scan method, whereas, last two rows depicts the sensitivity maps estimated by the E-maps method.

In Figure ‎4.4, a comparison between the reconstructed images obtained using pre-scan sensitivity maps (conventional method) and E-maps estimated by the eigenvalue decomposition method are shown. Column 1 and column 2 of Figure ‎4.4 depict the sensitivity maps (channel#1) of both the sensitivity maps estimation methods. The results show that the proposed FPGA implementation produces artefact free reconstructed images (from the under-sampled data) with both the sensitivity profile estimation methods.

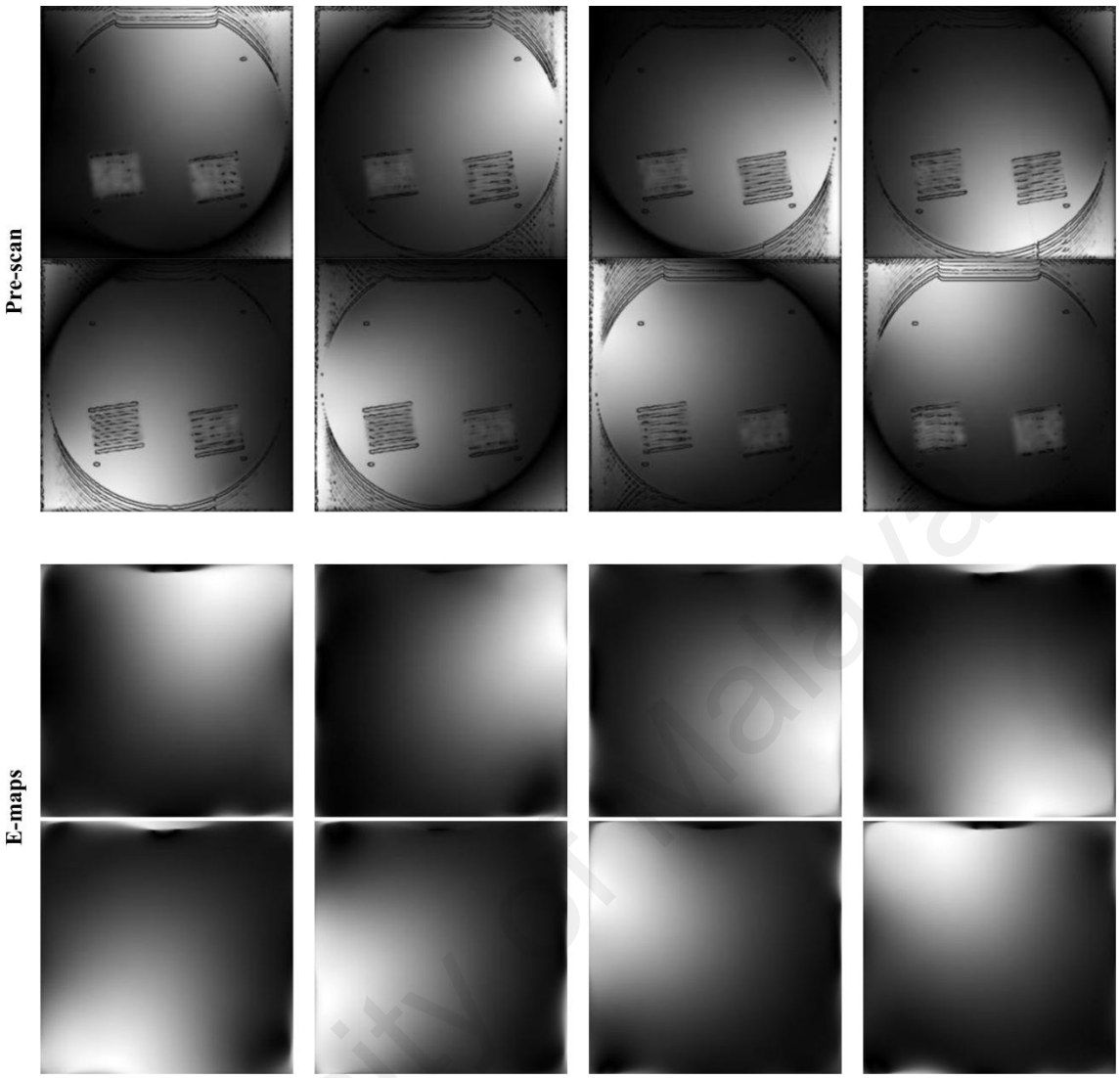


Figure 4.3: Receiver coil sensitivity maps (pre-scan and E-maps) for all the eight channels.

Figure 4.5 provides the artefact power of the proposed system reconstruction when using pre-scan and E-maps based sensitivity maps. The visual quality of the reconstructed images proves that the proposed system generates high temporal resolution images with significantly less artefact power, i.e., $2.449\times10^{-4}/5.9869\times10^{-4}$, $4.5465\times10^{-4}/8.9305\times10^{-4}$, $1.6545\times10^{-4}/4.6117\times10^{-4}$ and $3.0917\times10^{-4}/4.5569\times10^{-4}$ (Pre-scan/E-maps) for phantom, 1.5T human head, 3T human head_1 and 3T human head_2 datasets, respectively. The results show that artefact power of the

proposed system using E-maps is slightly higher than the pre-scan method reconstruction; however it is still in the acceptable range.

One significant advantage of the E-maps method to estimate sensitivity maps is that it does not require a pre-scan which makes it suitable for the cases where motion artefacts are considerable. SENSE with E-maps method to estimate sensitivity maps enable to reconstruct images from only the under-sampled data (having no central aliasing artefact) without any calibration scan. However, some extra calibration lines are acquired during the scan for E-maps method.

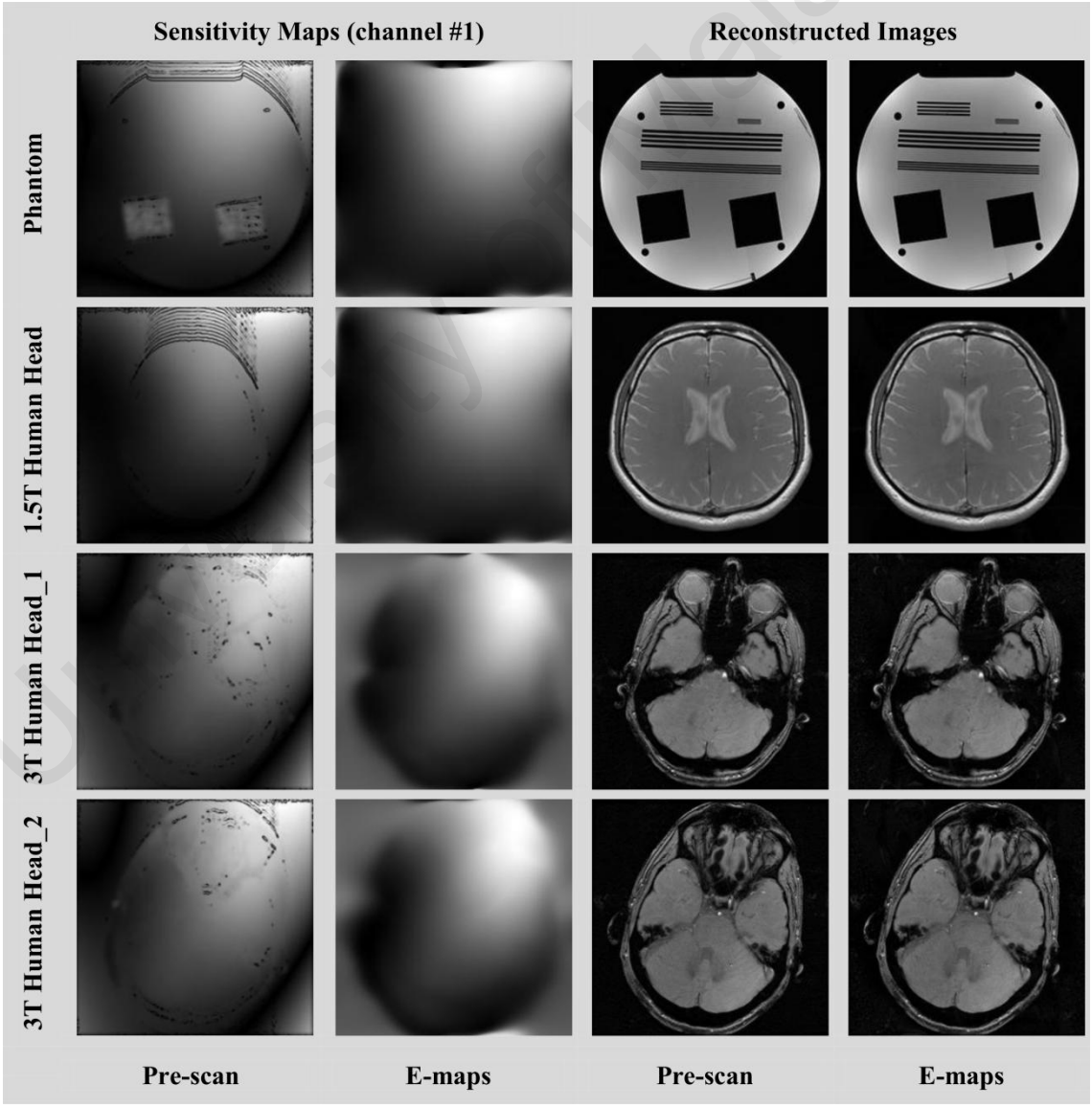


Figure 4.4: Sensitivity profiles and reconstructed images using sensitivity maps obtained from: (1) pre-scan (2) E-maps methods.

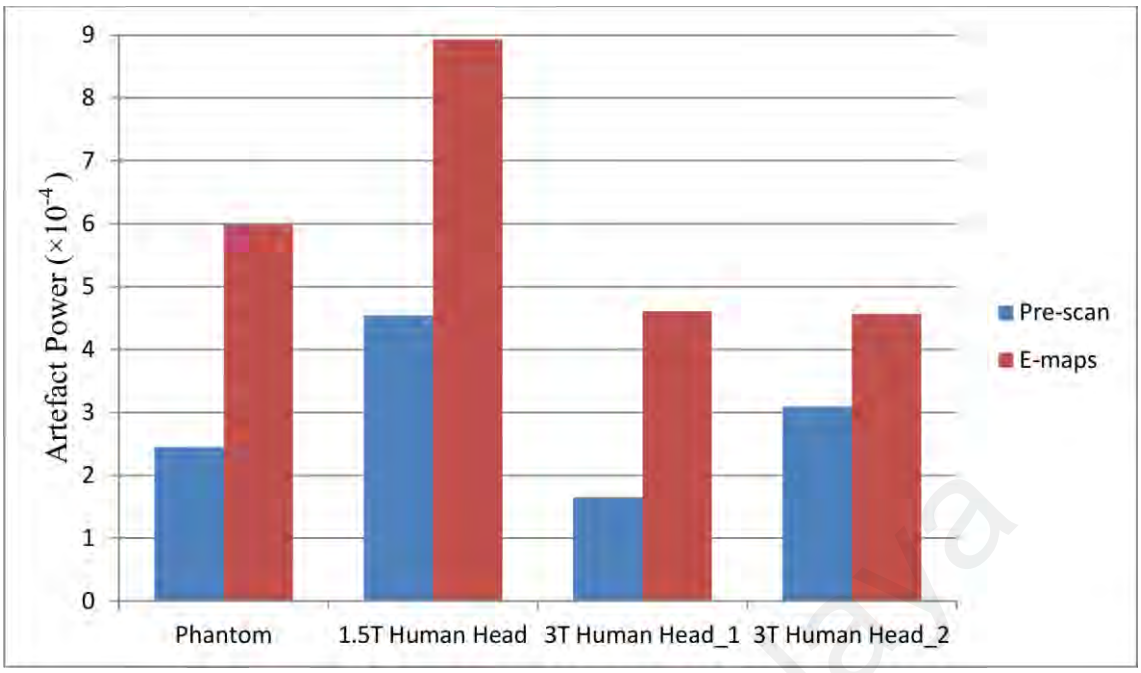


Figure 4.5: Artefact power comparison of the proposed SENSE architecture reconstruction using sensitivity maps obtained from: (1) pre-scan (2) E-maps methods.

4.3.2 Line Profile Analysis

Figure 4.6 and Figure 4.7 show the central line profile comparison of the reconstructed images produced by the proposed FPGA implementation and the reference images using 1.5T dataset and 3T dataset, respectively. Figure 4.6 (a-b) and Figure 4.7 (a-b) illustrate the line profiles of the reconstructed images generated by the proposed FPGA implementation using pre-scan sensitivity maps. Whereas, the line profiles of the proposed system reconstruction using E-maps are shown in Figure 4.6 (c-d) and Figure 4.7 (c-d).

The central line profiles of the reconstructed images prove that the proposed FPGA implementation of SENSE provides successful reconstruction results in both sensitivity estimation methods cases. The line profile comparison shows that there are no considerable variations in the line profiles of the reconstructed images as compared to the reference images.

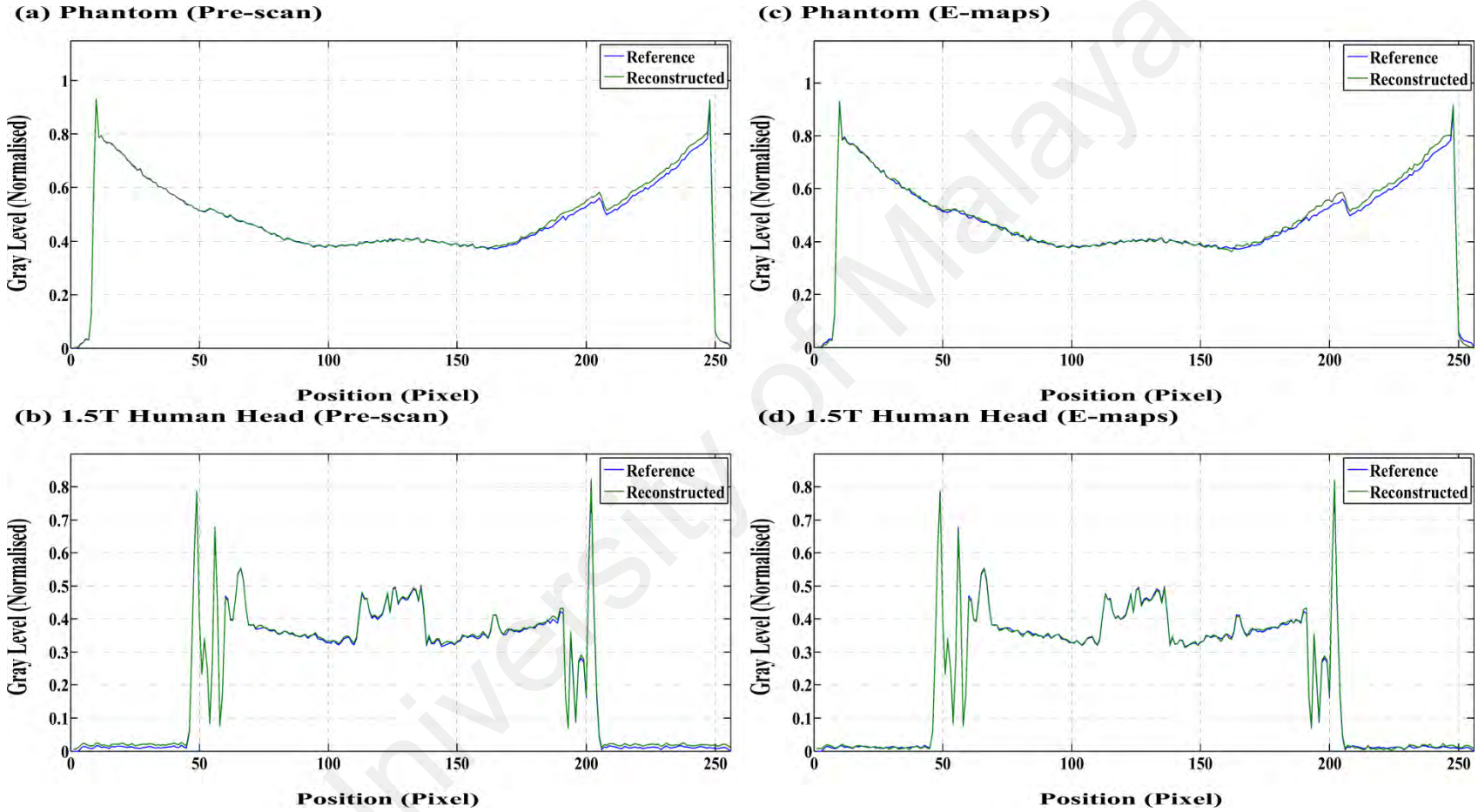


Figure 4.6: Central line profile comparison of the reconstructed images and the reference (fully-sampled) images (1.5T dataset). (a-b) show the line profiles of the reconstruction obtained from pre-scan method and (c-d) show the line profiles of the reconstructed images using E-maps.

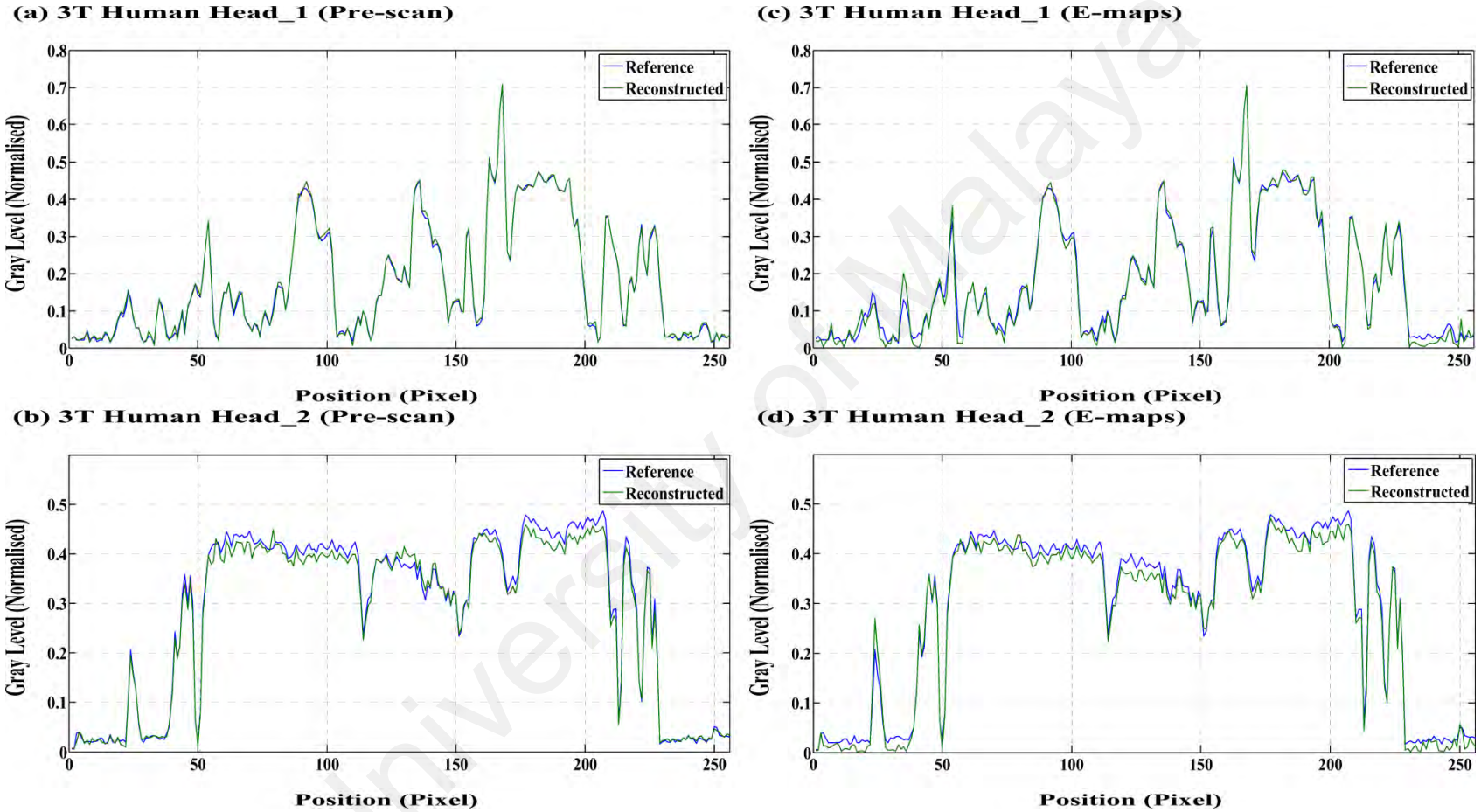


Figure 4.7: Central line profile comparison of the reconstructed images and the reference (fully-sampled) images (3T dataset). (a-b) show the line profiles of the reconstruction obtained from pre-scan method and (c-d) show the line profiles of the reconstructed images using E-maps.

4.3.3 Signal-to-Noise Ratio (SNR) Maps Evaluation

Signal-to-noise ratio (SNR) map was estimated using pseudo replica map method (Robson et al., 2008) for the reconstructed images obtained from the proposed FPGA design. SNR maps are widely used as an assessment tool for quantitative evaluation of the reconstructed images in clinical MRI. Figure ⁋4.8 shows the SNR maps of the reconstructed images with mean SNR values. Figure ⁋4.8 (a) illustrates the SNR maps of the pre-scan reconstruction, whereas Figure ⁋4.8 (b) shows the SNR maps of the E-maps based SENSE reconstruction. The proposed FPGA implementation shows that the mean SNR values of the reconstructed phantom images are more than 35 dB in each case. The mean SNR values for the E-maps based reconstruction (obtained from FPGA) of human head axial datasets are 29.8583 dB, 36.1761 dB and 35.6571 dB, respectively. Whereas the mean SNR values of the proposed system reconstruction with pre-scan sensitivity maps are slightly better, i.e., 31.9605 dB, 36.8575 dB and 36.0844 dB for human head axial datasets, respectively.

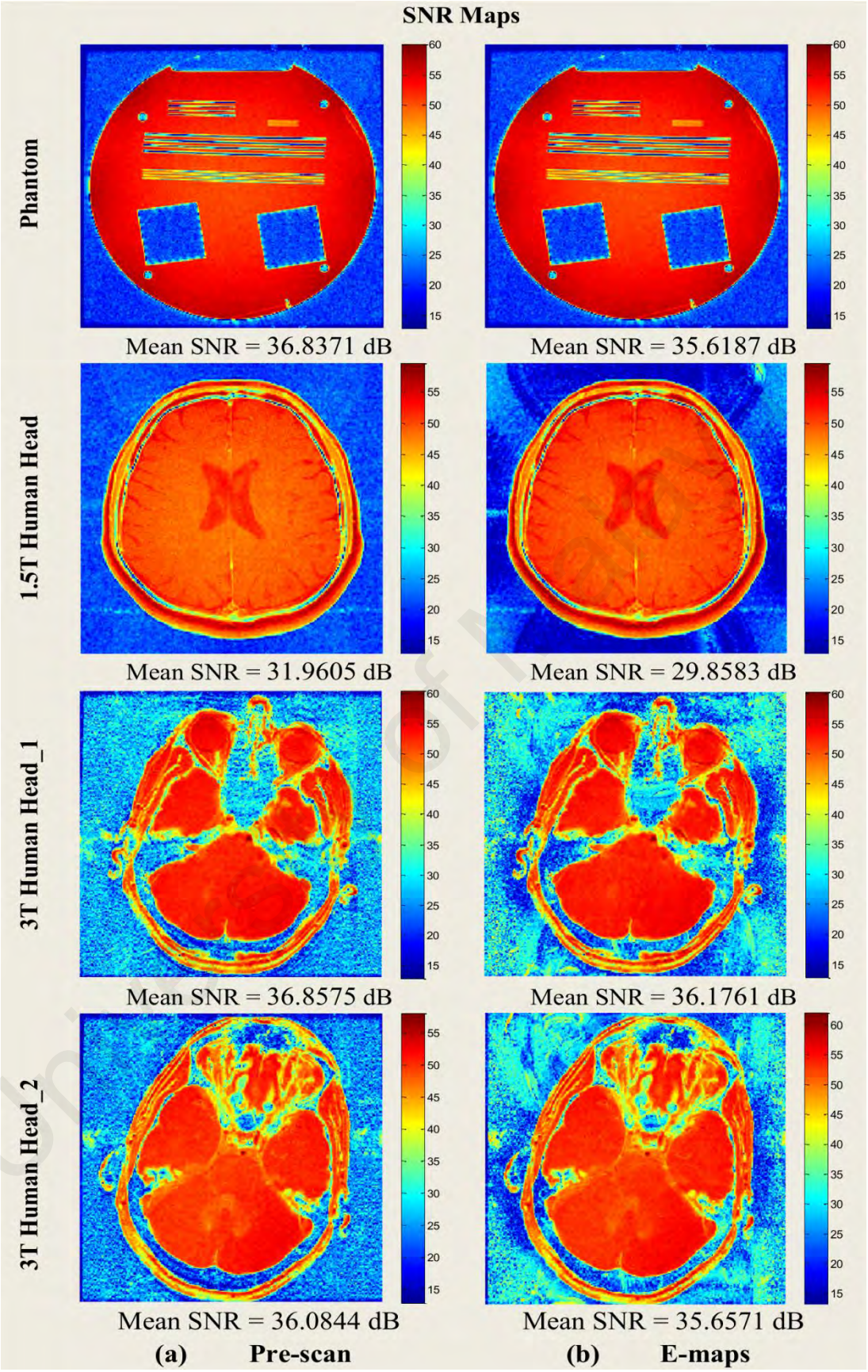


Figure 4.8: Pseudo multiple replica based SNR maps with mean SNR values of the reconstructed images using sensitivity maps obtained by (a) pre-scan and (b) E-maps methods.

4.3.4 Computation Time and Additional Advantages

Table 4.1 presents a comparison of the computation time with average artefact power and mean SNR between the proposed system implementation and different conventional implementations of SENSE (multi-core CPU and GPU) (Shahzad et al., 2016). The results show that the proposed FPGA implementation achieves significantly lesser reconstruction time than the multi-core CPU and GPU implementations while maintaining the artefact power and mean SNR values.

Table 4.1: Computation time analysis with avg. artefact power and mean SNR.

	Multi-core CPU (Shahzad et al., 2016)	**GPU (Shahzad et al., 2016)**	**FPGA (Proposed)**	
			Pre-scan	**E-map**
Reconstruction Time (*ms*)	47	4.7	0.164	0.164
Avg. Artefact Power	2.95×10^{-4}	2.98×10^{-4}	2.94×10^{-4}	6.02×10^{-4}
Mean SNR (dB)	35.43	34.07	35.44	34.33

The proposed system for run-time SENSE reconstruction merely consumes $0.164\,ms$ @ 200 *MHz* to produce reconstructed image, (Resolution size = 256×256 when $R = 2$), using any of the two (pre-scan or E-maps) sensitivity maps estimation methods. The proposed implementations in (Shahzad et al., 2016), consumed 4.7 *ms* and 47 *ms* when using multi-core CPU and GPU platforms, respectively. Table 4.1 shows the obtained results, which indicate that the proposed system (for both sensitivity methods) improves the computation time by 28.66 % and 286.59 % compared to the GPU and multi-core CPU implementations, respectively. The proposed FPGA implementation also achieves smaller significant mean SNR, similar to GPUs and multi-core CPUs implementations (Shahzad et al., 2016).

FPGA designs consume much lesser power than the conventional reconstruction platforms (CPU or GPU), as required for portable MRI scanners. Along with these technical improvements, the proposed design provides an accurate and efficient method of real-time SENSE reconstruction from multi-coil under-sampled data and works equally efficiently with different sensitivity profiles estimation methods. Finally, the proposed design implementation with E-maps enables SENSE reconstruction where coil sensitivities are difficult to estimate by conventional method.

4.4 Summary

In this chapter, a high-speed and low power FPGA implementation is proposed for real-time SENSE reconstruction using two different sensitivity maps (pre-scan and E-maps). The proposed FPGA based SENSE implementation has the potential to compute accurate images from under-sampled pMRI data. Efficient memory usage and less transmission cost with promising results in terms of artefact power, mean SNR and line profiles comparison are the significant features of the proposed work. Moreover, the overall results indicate that the proposed architecture is also useful for FPGA based pMRI reconstruction, when sensitivity maps are not easy to estimate.

CHAPTER 5: RECONSTRUCTED IMAGES VALIDATION USING CLASSIFIER

5.1 Introduction

In this modern era, the field of medical imaging proves its importance in hospitals, which increases the need of an automated and efficient diagnostic system for medical practitioners. The use of computer based automatic systems in medical image processing, medical analysis, verification, and classification, is now widespread and highly helpful. In recent years, significantly advanced imaging tools (e.g., Positron Emission Tomography (PET), X-ray Computed Tomography (X-ray CT) and Magnetic Resonance Imaging (MRI)) have been introduced in neurology and basic neuroscience fields, which enable in-vivo monitoring of the brain. MR image texture contains a rich source of information, which greatly increases the knowledge of the medical researchers to distinguish between the healthy and diseased anatomy, and this information is also a very helpful component for any classification system (McRobbie et al., 2006; Prasad, 2006; Scapaticci, Donato, Catapano, & Crocco, 2012; Westbrook, 2013). MRI has emerged as one of the popular choices to rule out alternative causes of dementia and to detect a variety of brain conditions, such as tumor, bleeding, swelling, infections, cysts, inflammatory conditions, or problem with the blood vessels (McKhann et al., 1984).

The radiologists' conventional process for brain MR images classification is visual inspection. However, because of the huge amount of imaging data, the existing manual measurements of analysis and interpretation of these structures are tedious, time consuming, costly, subject to fatigue of the human observer, and do not capture the full pattern of atrophy. Hence, it generates the requirement of developing automated diagnostic systems for analysis and classification of such medical images. These intelligent systems can be a great instrument for the medical personnel in diagnosis, pre-

surgical, and post-surgical procedures (Faisal, Parveen, Badsha, Sarwar, & Reza, 2013; Klöppel et al., 2008; Maji, Chanda, Kundu, & Dasgupta, 2007; Mwangi, Ebmeier, Matthews, & Steele, 2012; Zhang, Wu, & Wang, 2011).

In recently published work, various approaches of brain MRI classification have been proposed by different scholars. In general, most of the proposed systems consist of three sub-systems or phases. These phases are feature extraction, feature reduction, and classification. In (Chaplot, Patnaik, & Jagannathan, 2006), the authors have achieved 94% and 98% accuracy through classifiers based on Self-Organizing Map (SOM) and Support Vector Machine (SVM), respectively. They have used Discrete Wavelet Transform (DWT) for feature extraction but did not use any feature reduction technique. The authors in (El-Dahshan, Hosny, & Salem, 2010), have used Principal Component Analysis (PCA) with DWT for feature extraction and reduction. They have achieved 97% and 98% accuracy rates through feed-forward Back-Propagation Neural Networks (BPNN) and k-Nearest Neighbour (kNN) classifiers, respectively. Some other recent works by Zhang et al., (Zhang, Dong, Wu, & Wang, 2011; Zhang, Wang, & Wu, 2010; Zhang & Wu, 2012; Zhang, Wu, et al., 2011) have proposed different advanced methods for brain MRI classification to achieve high success rates. In all these schemes, they have used DWT and PCA for feature extraction and reduction, respectively. In (Das, Chowdhury, & Kundu, 2013), the authors introduced Ripplet Transform (RT) for feature extraction with Least Squares Support Vector Machine (LS-SVM) and achieved 100 % accuracy for small datasets (having less number of samples as well as less number of diseases) and 99.39 % accuracy for large datasets (having large number of samples including variety of diseases). However, the classification accuracies of most of the existing methods are greater than 90 % for small datasets but it decreases for large datasets. Therefore, the goal of this research is to achieve less complex and robust classifier system with highly accurate results for small as well as large datasets.

The main motivation of this work is to design an automated high accuracy classification system for brain MR image classification as normal or abnormal. The other incentive is to make the system generalized so that it can work equally efficiently for different brain MRI datasets (consists of a varying number of disease classes). In the proposed classifier system, fast DWT is used with PCA and LS-SVM. The fast DWT is used to compute only approximation features of the images and PCA reduces the dimension of the features, thus minimizing the computation time of the classification system. The categorization of images into normal or abnormal is done by the LS-SVM classifier, which automatically selects the appropriate parameter value of the kernel function by the proposed algorithm. Recent literature accounted that higher classification accuracy can be achieved by LS-SVM rather than other existing data classification techniques (Cawley & Talbot, 2004; Chu, Ong, & Keerthi, 2005). Some of the other latest classification methods are also examined in this chapter using the same datasets for comparative analysis.

5.2 Materials and Methods

The proposed classifier system for brain MRI is based on the following techniques: fast DWT, PCA, and LS-SVM. The main system is divided into two phases: training phase and testing phase. Figure 5.1 shows the block diagram and flow of the proposed design. The MRI benchmark dataset is required initially to perform the classification, so the standard and generalized database is established by obtaining brain MR images from 'Harvard Medical School' (http://med.harvard.edu/AANLIB/) and 'Open Access Series of Imaging Studies (OASIS)' (http://www.oasis-brains.org/). In the training and testing phases, first of all, fast DWT is computed which extracts the features of the images in the database. Then, these extracted features of the images are processed by the PCA block for feature reduction. Finally, in the training section, LS-SVM classifier is trained

by these reduced features. Meanwhile, the system is intelligent so that it gets the optimal values of the hyper-parameters of the radial basis function (kernel) for LS-SVM. Lastly, in the testing section, query image will be classified as normal or abnormal.

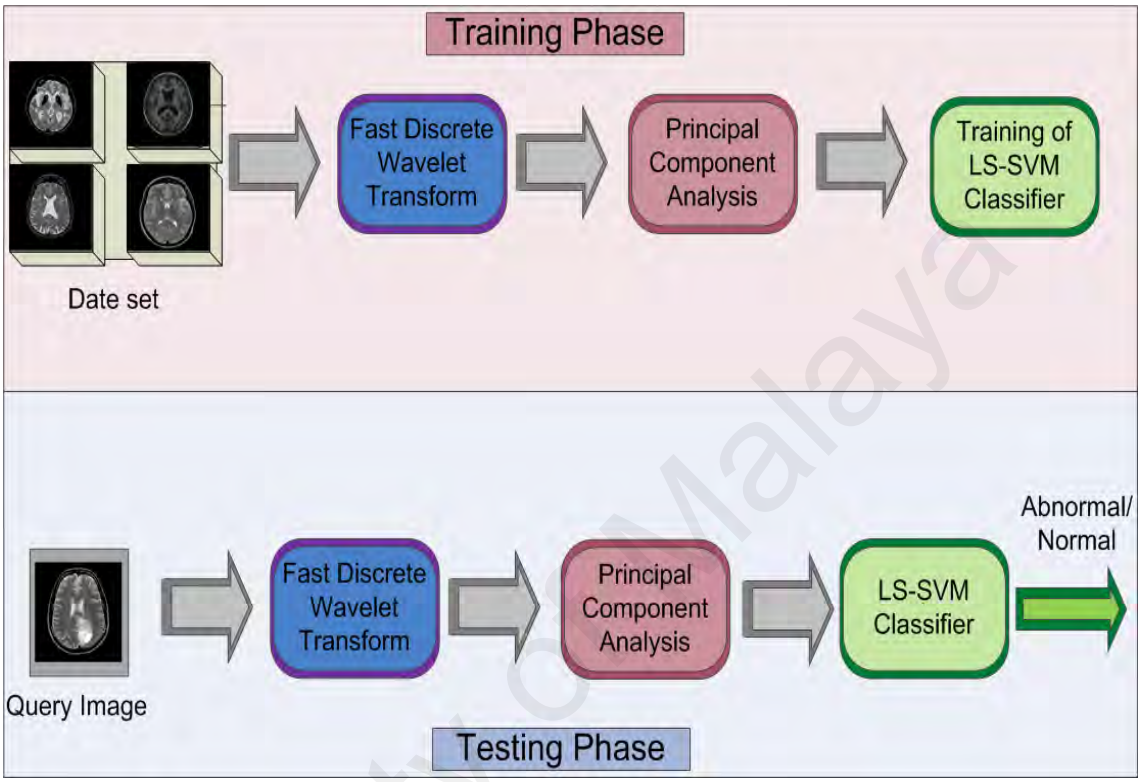


Figure 5.1: Proposed system methodology.

5.2.1 Feature Extraction Scheme

The prime objective of the feature extraction is to identify the relevant features in the image for faster, easier, and better understanding of the images. The extracted features provide the characteristics of the input image pattern, which includes major information about the images. The proposed classifier will get only important features of the images after feature extraction, which highly improves its efficiency and accuracy, and reduces the computation time.

There are many algorithms used in previous research, such as DWT, RT, and some other techniques. DWT and its various versions were used extensively by the scholars for feature extraction in brain MRI classification (Chaplot et al., 2006; El-Dahshan et al., 2010; Maitra & Chatterjee, 2006; Zhang, Dong, et al., 2011; Zhang et al., 2010; Zhang & Wu, 2012; Zhang, Wu, et al., 2011). In (Das et al., 2013), the authors used RT for feature extraction, which increases the complexity of the design and it is computationally expensive too. The proposed classifier system uses fast DWT because one of the characteristics of brain MRI is that it can be sparsified. Therefore, it can be represented in more sophisticated domains, such as wavelet domains (Lustig, Donoho, Santos, & Pauly, 2008).

5.2.1.1 2-D Fast Discrete Wavelet Transform

2-D fast DWT is a robust execution of the wavelet transform using the dyadic scales and positions (Zhang & Wu, 2011). The 2-D fast DWT is an iterative computational approach and can be expressed by Figure 5.2. The $F_i(m,n)$, $F_i^H(m,n)$, $F_i^V(m,n)$, and $F_i^D(m,n)$ are the approximation, horizontal, vertical, and diagonal coefficients in this figure, respectively. $F_i(m,n)$ provides the next approximation images computed by fast DWT and *i* is the value of the decomposition level of wavelet transformation. Whereas, *m* and *n* represent the row and column of the image, respectively. The blocks containing the $g(n)$ and $h(n)$ are low-pass and high-pass decomposition filters, respectively. Finally, blocks containing '2' with a down arrow illustrate down sampling by 2. The approximation component of the image can also be regarded as LL sub-band, while the horizontal (LH sub-band), vertical (HL sub-band) and diagonal (HH sub-band) can be regarded as the detailed components of the image. H represents a high pass filter applied on rows and columns of the image while L corresponds to low pass filter applied on rows and columns of the image.

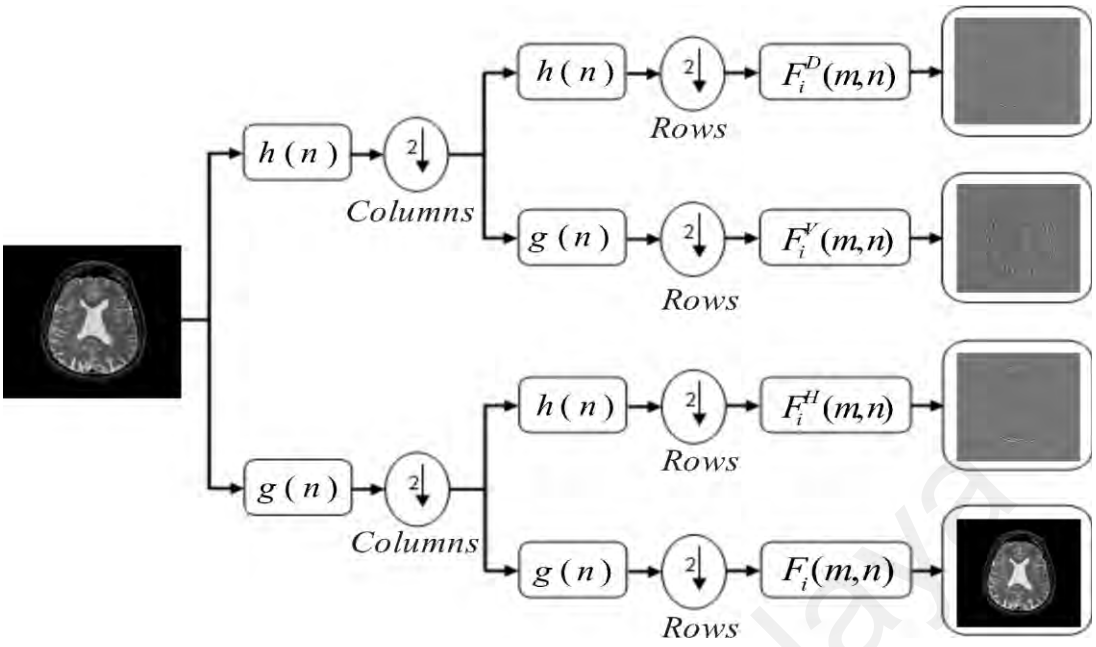


Figure 5.2: Schematic of 2D fast DWT

For example, LL sub-band image is produced: first columns are analyzed with low pass filter and then low pass filter is applied on rows. The approximation component will be treated as a mother component for the next iteration to find the next level of decomposition. Mathematically, the equation for approximation (LL sub-band) can be derived from the iterative method of filtering and down sampling operations as:

$$S1_i(m,n) = F_{i-1}(m,n) * g(n) \tag{5.1}$$

$$S2_i(m,n) = S1_i(m,n)\big|_{n=2k;} \qquad k \geq 0 \tag{5.2}$$

$$S3_i(m,n) = S2_i(m,n) * g(n) \tag{5.3}$$

$$F_i(m,n) = S3_i(m,n)\big|_{m=2k;} \qquad k \geq 0 \tag{5.4}$$

Finally,

$$F_i(m,n) = \left[g(n) * \left\{ F_{i-1}(m,n) * g(n) \right\} \Big|_{n=2k;\quad k \geq 0} \right] \Big|_{m=2k;\quad k \geq 0}, \qquad (5.5)$$

where $i = 1, 2, 3, \ldots, N$.

In equation (5.5), $F_0(m,n)$ (when $i = 0$) will be the initial image and $F_i(m,n)$ provides the next approximation images computed by the fast DWT and *i* is the value of the decomposition level of the wavelet transformation. In this work, 3-level "Haar" wavelet transform is used to extract the features. Only LL sub-band is computed instead of implementing the overall transformation to make the feature extraction faster in the proposed scheme as compared to the previous works. It significantly reduces the computation time of the proposed system without disturbing the accuracy of the classifier.

5.2.2 Feature Reduction using Principal Component Analysis

Feature reduction is one of the basic components of any robust classifier, which reduces the massive database by measuring certain properties or features. Large database also increases the excessive features to be classified and this increases the computation time and the storage memory. Furthermore, sometimes it increases the complexity of the classifiers, which is called curse of dimensionality. Therefore, it is necessary to reduce the dimensions of the large datasets by applying feature reduction schemes.

In the feature extraction phase, DWT also decreases the features, which shrinks the large computational data. A Brain MR image is represented by $F(m,n)$ with the dimension of *m*-rows and *n*-columns in this thesis. Normally, in brain MRI standard datasets, the images are square ($m = n$). After applying DWT for computing its

approximation component, the size of the image will become: $\left(\frac{m}{2^q}, \frac{m}{2^q}\right)$, here, q is the level of the wavelet transformation and $\frac{m}{2^q} << m$, where $q = 3$ in the proposed scheme. Therefore, the new image will be converted into $F_a\left(\frac{m}{2^q}, \frac{m}{2^q}\right)$ and the dimension decreases. However, still the dimension is large enough; as a result, there is a need of further reduction in dimension.

One of the most popular and widely used techniques to reduce the dimension is PCA. PCA extracts the linear lower-dimensional representation of the data such that the variance of the reconstructed data is preserved (Duda, Hart, & Stork, 2012; Sengur, 2008). This technique has three characteristics: it provides that the elements of the input vectors are uncorrelated with each other; it sorts out the largest variation resulting orthogonal components in ascending order, and it discards the least variance components in the dataset. By using PCA, the dimension of the feature extracted image is decreased from $\frac{m}{2^q}$ to the number of selected principal components. The number of the selected principal components depends upon the ratio between the total variance of the original feature set and the total variance of the reduced feature set. Therefore, the main thought behind using PCA in our research is to further reduce the dimensionality of the fast DWT approximation feature components. This leads to provide efficient information to the classifier for making rapid and accurate decision.

5.2.3 Support Vector Classification

SVM is an example of a supervised classification technique, which offers an efficient method of obtaining models for classifications (Patil, Shelokar, Jayaraman, & Kulkarni, 2005; Wang & Zhang, 2011). SVMs are supervised in the sense that they include a

training session to learn the differences between two groups, which are going to be classified. This algorithm is based on the theory of statistical learning, which helps in improving the general aptitude of machines to learn unseen data (Vapnik, 2000). Recently, SVMs are widely used in many real-life applications, such as object detection (Chen & Juang, 2013), face identification in images (Magalhães, Sousa, Araújo, & Correia, 2013), hand written alphabets recognition (Kumar, Sharma, & Rana, 2012), and brain images abnormalities classification (Das et al., 2013; Matoug, Abdel-Dayem, Passi, Gross, & Alqarni, 2012; Zhang & Wu, 2012). SVM classification is highly accurate having elegant mathematical tractability than other classification techniques, like artificial neural networks, Bayesian networks, and decision tree. In (Cawley & Talbot, 2004), the authors indicate that generally for the higher classification accuracy, an improved version of SVM, such as LS-SVM is remarkably better than other existing algorithms. In our proposed technique, LS-SVM classification is used due to its efficiency and robustness.

5.2.3.1 LS-SVM Classification

The main drawback of SVM has been overcome by the LS-SVM, which decreases the computational burden for huge-dimensional datasets. In LS-SVM approach, the cost function of the typical SVM is mathematically manipulated. This mathematical change simplifies the problem solution by introducing a least squares term in the cost function. Due to this improvement, the solution is acquired by solving a set of linear equations, instead of the conventional method of solving quadratic programming for SVM (Suykens & Vandewalle, 1999; Gestel et al., 2004). This approach significantly reduces the computation time and complexity of the classifier.

Suppose, a given training set $\{x_i, y_i\}_{i=1}^{N}$ where x_i is an *n*-dimensional vector from the *N*-size input space and has a corresponding label $y_i \in \{-1, +1\}$ for $i = 1, 2, \ldots, N$. The non-linear function estimation takes the following form:

$$y = sign\left[W'\varphi(x) + b\right] \tag{5.6}$$

where $\varphi(x)$ represents the high dimensional feature space, which is non-linearly mapped from the input space, W is the weight vector, and b denotes the bias term. Then, the solution of such optimization problem can be obtained as follows (Suykens & Vandewalle, 1999; Gestel, et al., 2004):

$$\min_{W,b,e} J(W,b,e) = \frac{1}{2}W'W + \frac{\gamma}{2}\sum_{i=1}^{N} e_i^2 \tag{5.7}$$

subject to equality constraints:

$$y_i\left[W'\varphi(x_i) + b\right] = 1 - e_i, \quad i = 1, 2, \ldots, N \tag{5.8}$$

where γ is the regularization parameter and e_i represents the random errors.

The Lagrangian expression corresponding to equation (5.7) can be defined as:

$$J\left(W,b,e_i,\alpha\right) = J(W,b,e) - \sum_{i=1}^{N}\alpha_i\left\{y_i\left[W'\varphi(x_i) + b\right] - 1 + e_i\right\} \tag{5.9}$$

where $\alpha_i (i = 1, 2, \ldots, N)$ are Lagrange multipliers. The conditions for optimality yields:

$$\begin{cases} \dfrac{\partial L}{\partial W} = 0 \Rightarrow W = \sum_{i=1}^{N} \alpha_i \, y_i \, \varphi(x_i) \\ \dfrac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i \, y_i = 0 \\ \dfrac{\partial L}{\partial e_i} = 0 \Rightarrow \alpha_i = \lambda \, e_i \\ \dfrac{\partial L}{\partial \alpha_i} = 0 \Rightarrow y_i \left[W' \varphi(x_i) + b \right] - 1 + e_i = 0 \end{cases} \quad (5.10)$$

$$i = 1, 2, \ldots, N$$

Referring to Suykens's and Gestel's works (Suykens & Vandewalle, 1999; Gestel, et al., 2004), the solution can be achieved by solving the following linear system:

$$\begin{bmatrix} 0 & y^T \\ y & \beta\beta^T + \gamma^{-1} I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \vec{1} \end{bmatrix} \quad (5.11)$$

where $\beta = \left[\varphi(x_1) \, y_1, \; \varphi(x_2) \, y_2, \; \ldots, \; \varphi(x_N) \, y_N \right]^T$, $y = \left[y_1, \; y_2, \; \ldots, \; y_N \right]^T$, $\vec{1} = \left[1, 1, \ldots, 1 \right]^T$, $\alpha = \left[\alpha_1, \; \alpha_2, \; \ldots, \alpha_N \right]^T$, and $Z = \beta\beta^T$ takes the form as $Z_{kl} = y_k \, y_l \, \varphi(x_k)^T \, \varphi(x_l) = K(x_k, x_l)$ $(k, \, l = 1, \, 2, \, \ldots, N)$ according to Mercer's condition.

For the choice of kernel functions $K(x_k, x_l)$ satisfying the Mercer conditions and a new test sample point *x*, the LS-SVM classifier is given by:

$$y(x) = sign\left[\sum_{i=1}^{N} \alpha_i \, y_i \, K(x, x_i) + b \right] \quad (5.12)$$

Table 5.1 illustrates some of the common kernel functions used in LS-SVM. In this thesis, LS-SVM is used with Radial Basis Function (RBF) as a kernel function for training because previous literature proved that RBF is a more supported kernel function. RBF also reduces the computational complexity and improves the performance of LS-SVM (Cawley & Talbot, 2004). While using RBF as a kernel in LS-

SVM, there are two tuneable hyper-parameters, which should be optimized accurately for achieving the best results. The hyper-parameters of RBF are σ and γ. The trade-off between margin maximization and error minimization is controlled by regularization parameter γ, while the kernel parameter σ determines the width of the kernel.

Table 5.1: Common kernel functions for LS-SVM.

Kernel Name	Function Expression
Linear	$K(x,y) = x^T y$
Polynomial	$K(x,y) = \left(1 + \frac{x^T y}{\sigma^2}\right)^d$
RBF	$K(x,y) = \exp\left\{-\frac{\|x-y\|^2}{\sigma^2}\right\}$
SRBF	$K(x,y) = \exp\left\{-\frac{\sum_{i=1}^{N}(x-y)^2}{\sigma^2}\right\}$

5.2.4 Hyper-parameters Optimization and Generalization of LS-SVM.

For applying LS-SVM with any kernel function, one of the most important issues is to choose the hyper-parameters, which plays a critical role in the performance of the classifier. With different parameters of the same kernel function, the LS-SVM prediction model provides different performance (the best optimization of hyper-parameters results the high accuracy of the classifier). Generally, two parameters, σ and γ, should be optimized to affect learning and generalization of LS-SVM with RBF kernel.

There are many complex algorithms available for the optimization of hyper-parameters, namely, particle swarm optimization-based hyper-parameters selection (Guo, Liang, Wu, & Wang, 2006), non-parametric noise estimator method (Lendasse, Ji, Reyhani, & Verleysen, 2005), and grid search method (Zhang, Li, & Zhong, 2009). Other than these excessively iterative methods, pilot run is also used to find the values of these parameters by trial and error. Mainly, for reducing the generalization error, the previously published schemes used different cross validation methods. The extreme case of cross validation is Left-One-Out (LOO), it provides almost an unbiased estimate of the generalization error but it is computationally very expensive. Meanwhile, the *k*-fold cross validation gives an excellent estimate of the generalization error at low cost (Duan, Keerthi, & Poo, 2003). In the proposed classifier, less mathematically complex, comparatively less time consuming, and intelligent algorithm as shown below (Algorithm 1) is used with *k*-fold cross validation. This algorithm gives the optimized value of the RBF-sigma (a more necessitated parameter for LS-SVM with RBF kernel) with *k*-fold cross validation, which makes more accurate, reliable, and generalized classifier.

Algorithm 1. Pseudocode of the RBF-sigma optimized value with *k*-fold cross validation method.

Step 1: Initialization: $L_{\min}$, $\varepsilon_{\min}$, σ_l, σ_u, k, $i = 1$, $L_s = 1$ and $L = 1$
Step 2: Input one set of brain MR images from *k*-fold datasets $k_{data}(i)$
Step 3: $\sigma_l = \sigma_l + L$
Step 4: Validation for certain parameter (σ_l)
Step 5: Calculate ε
Step 6: if $\varepsilon < \varepsilon_{\min}$
Obtain the certain parameter, $\sigma_k = \sigma_l$
if $i < k$
$k = k + 1$, $L = 0$ and goto Step 2
else
goto Step7
endif
else
if $\sigma_l > \sigma_u$
if $L > L_{\min}$
$L_s = \frac{L_s}{10}$, $i = 1$ and $L = 1$, update the value of σ_l and σ_u by lowest error interval and goto Step 3
else
$i = i + 1$ and $L = L + L_s$ goto Step 3
endif
endif
endif
Step 7: $\sigma_{opt} = \frac{1}{k}\sum_{i=1}^{k}\sigma_k(i)$, obtain optimized and *k*-fold cross validated value of RBF-sigma

In this algorithm:

$L_{\min}$	:	Minimum tolerance in parameter value
$\varepsilon_{\min}$	:	Tolerance, i.e., the minimum number of tests fail
σ_l	:	Lower bound of the parameter
σ_u	:	Upper bound of the parameter
k	:	Number of folds for cross validation

5.2.5 A Graphical Implementation of the Proposed Classifier

A GUI (Graphical User Interface) has been developed for the proposed classifier. The proposed classifier technique is developed using wavelet toolbox, image processing toolbox, and statistics toolbox of MATLAB software. The code can be tested or executed on any MATLAB compatible computer platform. This program provides an interactive tool for MR image classification. The GUI of the proposed classifier is shown in Figure 5.3.

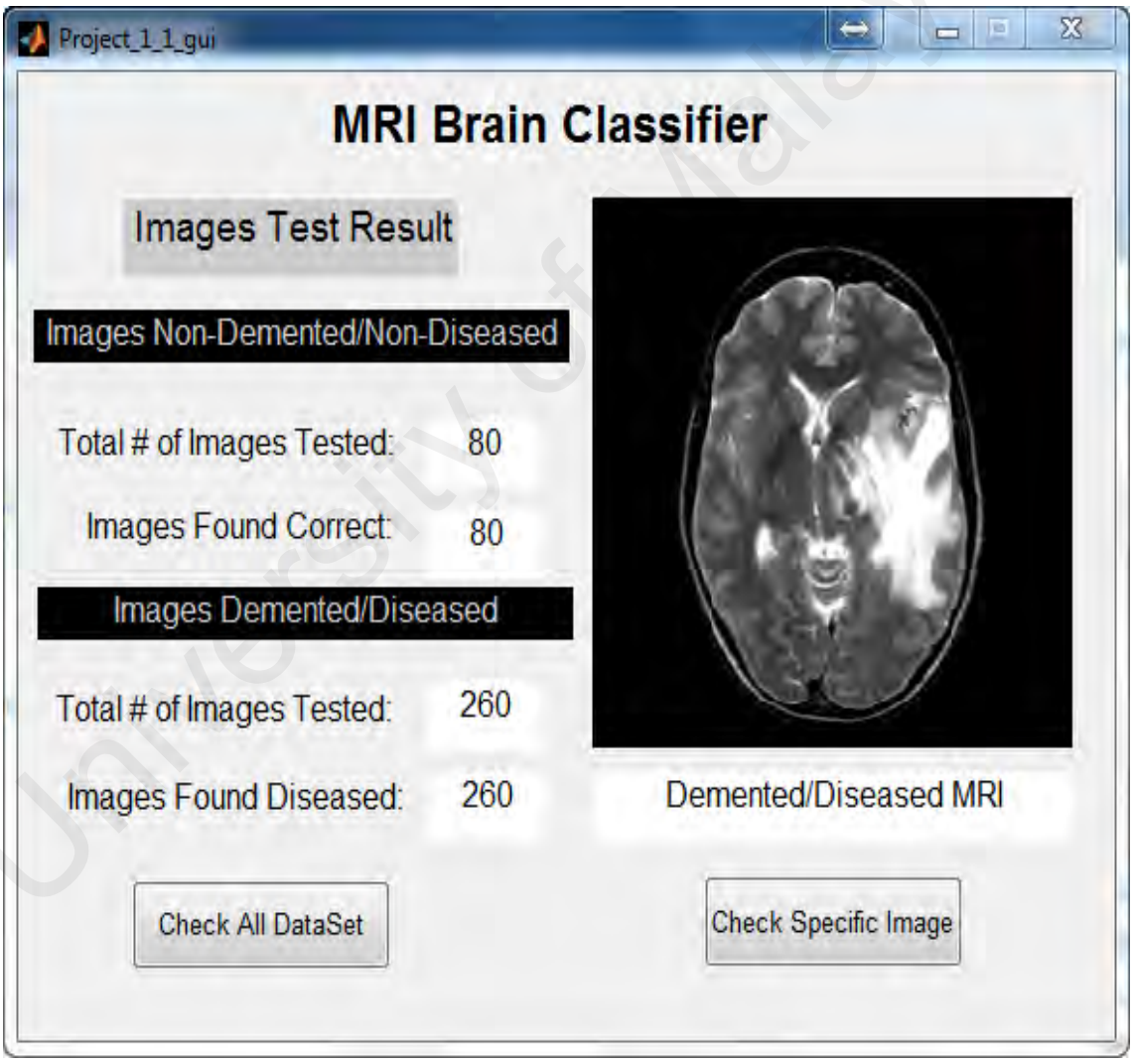


Figure 5.3: The GUI of a proposed classifier.

The GUI has following inputs and outputs:

Input Buttons:

i. **Check All DataSet:** To classify all the images of 340 patients' scans (database).
ii. **Check All DataSet:** To classify any specific image. Click on the button then window browser will open for selecting the query image.

Output Fields:

iii. **Total # of Images Tested:** The total number of images value is shown in the "Total # of Images Tested" text boxes.
iv. **Images Found Correct:** The total number of images classified as normal is mentioned in the "Images Found Correct" text box.
v. **Images Found Diseased:** "Images Found Diseased" text box will display the total number of images classified as abnormal.
vi. **Query Image with Result:** On the right side of the GUI, query image is shown with its classification result.

5.2.6 Experimental Setup and Dataset

The benchmark MRI database was gathered from 'OASIS' and 'Harvard Medical School' MRI databases. The collected database consists of human brain MR images. Both the datasets consist of T1-weighted and T2-weighted MR brain images in the axial plane. The scan parameters used for these datasets were Voxel res: $1.0 \times 1.0 \times 1.25\,(mm)$, Rect. FOV: 256/256, TR: 9.7 (*ms*), TE: 4.0 (*ms*), TI: 20.0 (*ms*), and Flip Angle: 10°. All the subjects are right-handed and include both men and women scans. The dimensions of the images are 256×256 in a plane-resolution. The dataset consists of 340 patients'

brain MRI scans with the demographic and clinical details of the patient. These details include age, gender, Clinical Dementia Rating (CDR), Mini Mental State Examination (MMSE), and different test parameters.

The brain MR images are divided into two groups. The first group (Group-1) contains the images with 11 types of brain diseases, which have been widely used as a benchmark dataset in the literature. This group consists of healthy brain MR images along with the following brain diseases: Glioma, sarcoma, Alzheimer's disease, Alzheimer's disease with visual agnosia, Pick's disease, Huntington's disease, Meningioma, chronic subdurnal hematoma, multiple sclerosis, cerebral toxoplasmosis, and herpes encephalitis. The second more generalized benchmark dataset group (Group-2) has 24 types of diseases in total, among which, 11 diseases are the same as the previous group (Group-1) along with healthy brain MRIs. The 13 new forms of abnormal images having the following diseases: metastatic bronchogenic carcinoma, metastatic adenocarcinoma, motor neuron disease, cerebral calcinosis, AIDS dementia, Lyme encephalopathy, Creutzfeldt-Jakob disease, hypertensive encephalopathy, multiple embolic infarctions, cerebral haemorrhage, cavernous angioma, vascular dementia, and fatal stroke. The Group-2 dataset is more general with 24 different diseases, which leads to test the classifier more comprehensively. The sample of each disease is shown in Figure 5.4.

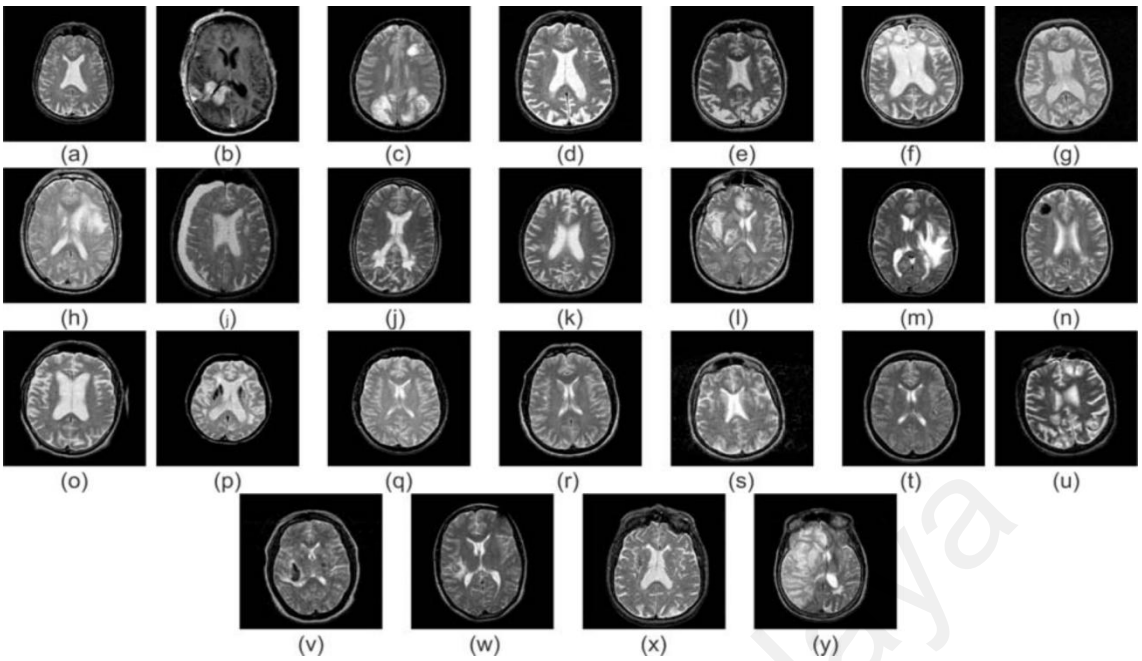


Figure 5.4: Sample images of various diseases in brain MRI dataset: (a) Normal brain (b) Glioma (c) Sarcoma (d) Alzheimer's disease (e) Alzheimer's disease with visual agnosia (f) Pick's disease (g) Huntington's disease (h) Meningioma (i) Chronic subdurnal hematoma (j) Multiple sclerosis (k) Cerebral toxoplasmosis (l) Herpes encephalitis (m) Metastatic bronchogenic carcinoma (n) Metastatic adenocarcinoma (o) Motor neuron disease (p) Cerebral calcinosis (q) AIDS dementia (r) Lyme encephalopathy (s) Creutzfeldt-Jakob disease (t) Hypertensive encephalopathy (u) Multiple embolic infarctions (v) Cerebral haemorrhage (w) Cavernous angioma (x) Vascular dementia (y) fatal stroke.

The demographic information about the dataset is shown in Table 5.2. There are a total of 255 brain MR images in Group-1, which includes 220 abnormal and 35 normal MR images. Group-2 consists of 260 abnormal and 80 normal brain images (total of 340 brain MR images).

Table 5.2: Demographic information.

Group	Normal/Healthy	Abnormal/Demented
Age (mean, range) at MRI scan	68.89 (33-94)	76.65 (62-96)
Sex (F/M) (ratio)	5/2	13/9
MMSE score (mean, range)	29.09 (25-30)	26.79 (14-30)

Table 5.3: Settings of training and validation images for dataset groups.

Groups	Total no. of images		Total no. of training images		Total no. of validation images	
	Normal	Abnormal	Normal	Abnormal	Normal	Abnormal
Group-1	35	220	28	177	7	43
Group-2	80	260	27	100	53	160

Table 5.3 describes the settings of the training and validation images for the data groups. Validation images consist of a number of images including diseased and normal brain images.

5.2.7 Performance Measures

The confusion matrix is widely used to determine the performance of the classifiers. The possible outcomes of the proposed classifier can be described as:

TP (True Positive) : Number of Abnormal images correctly diagnosed

TN (True Negative) : Number of Normal images correctly classified

FP (False Positive) : Number of Normal images classified as Abnormal

FN (False Negative) : Number of Abnormal images diagnosed as Normal

Sensitivity: is the probability of the test finding the abnormal case among all the abnormal cases.

$$\frac{TP}{TP + FN} \tag{5.13}$$

Specificity: is the probability of the test finding the normal case among all normal cases.

$$\frac{TN}{TN + FP} \tag{5.14}$$

Accuracy: is the fraction of test results predicted as correct among all the cases.

$$\frac{TP + TN}{TP + FN + TN + FP} \tag{5.15}$$

5.3 Results and Discussion

In order to evaluate the performance of the proposed system in terms of feature reduction efficiency, sensitivity, specificity, accuracy, time analysis, comparisons with different high-tech schemes, and computation complexity, several experiments were performed on the benchmark datasets of brain MRI. Before comparing the proposed method with other schemes, some methodological aspects are described here.

This work used fast DWT with some modifications, which reduces the computation time. "Haar" wavelet transform is used for wavelet decomposition. The DWT decomposition configuration extracts the main features and also reduces the size of the brain MRI, which is initially 256×256 and reduces to 32×32. The PCA block uses these extracted feature vectors and gathers the high variance components. In the proposed scheme, high success rate is achieved by using only 8 principal components. For classification purposes, the classifier was trained by only 0.012% and 0.78% of the original brain MRI and approximation components of the wavelet features, respectively. Therefore, due to this method, the system not only achieved 99.9% feature reduction but also retained its high accuracy. This feature reduction achievement with a higher correctness rate is remarkably impressive than the other state-of-the-art (Das et al., 2013) brain MRI classification techniques. The performance of the proposed method related to the number of principal components used was evaluated by using different values of the principal components in the experiments. Figure 5.5 shows the

performance evaluation in terms of sensitivity, specificity, and accuracy against the number of principal components used by the classifier. A number of the features may increase the complexity of the machine learning system to classify between two groups, which eventually decrease the sensitivity and/or specificity of the system. It is easily found that our proposed system works efficiently by using only 8 principal components for image presentation. The sensitivity, specificity, and accuracy were computed by observing the values of *TP*, *TN*, *FN*, and *FP* outcomes during the experiments. Figure 5.6 shows Receiver Operating Characteristic (ROC) curves for evaluating the classification accuracy of the proposed system. The proposed system correctly classified the MR images of Group-1 and Group-2 with an average Area under the Curve (AUC) of 100%, with 0% standard deviation.

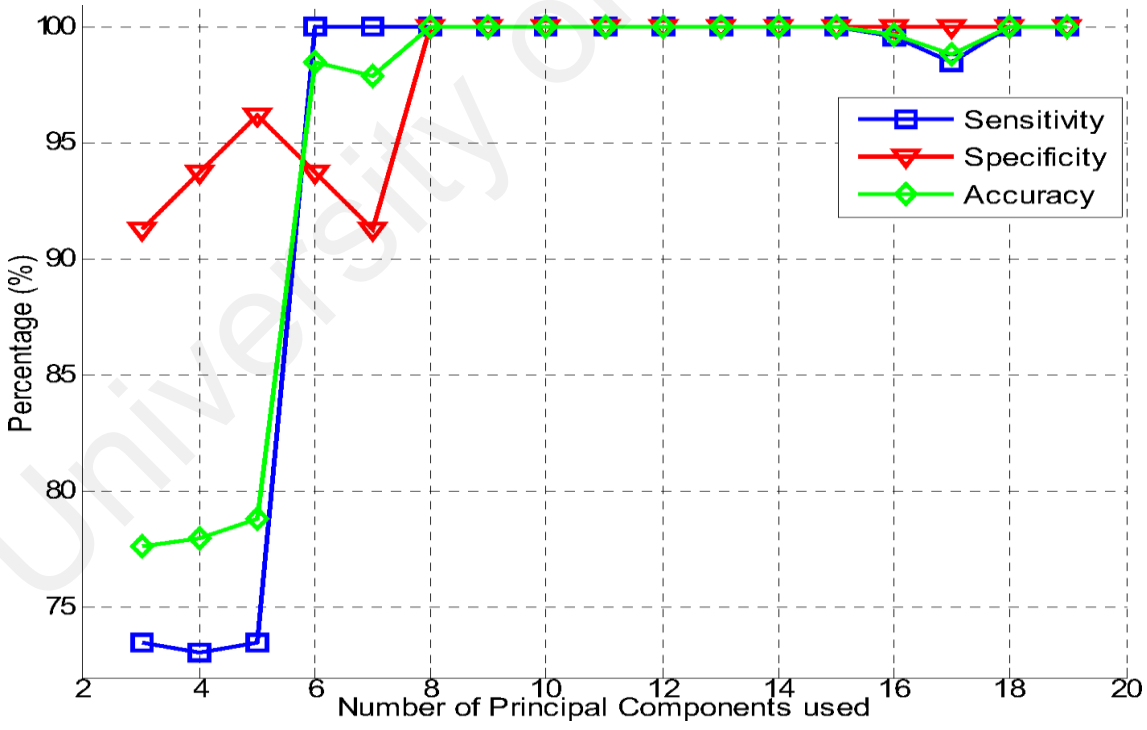


Figure 5.5: Sensitivity, specificity, and accuracy with respect to the number of principal components used.

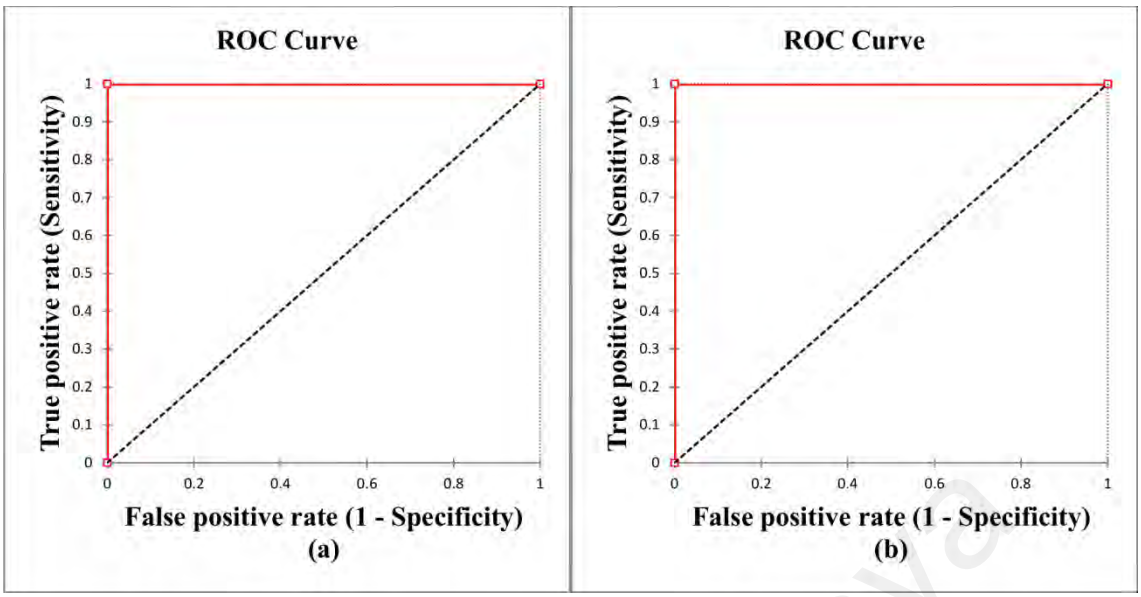


Figure 5.6: ROC curves of performance evaluation: (a) Group-1 and (b) Group-2.

For classification, LS-SVM is used with RBF kernel. Normally, excessive computations and experiments are performed to get the best value of the hyper-parameters of the kernel. In recent published approaches, to estimate the suitable value of the parameters of the function, such as the order d in Homogeneous Polynomial (HPOLY) and Inhomogeneous Polynomial (IPOLY) kernel (Zhang & Wu, 2012), the scaling factor γ in Gaussian Radial Basis (GRB) kernel (Zhang & Wu, 2012), and the kernel and regularization parameters in LS-SVM (Das et al., 2013), trial and error method is used iteratively by changing the value of the parameters manually. It took hundreds of experiments to find the value of the kernel parameters. However, this method is cumbersome, time consuming and wasting human resources. In this study, a simple algorithm is therefore used to find the optimized value of the parameter that makes this system intelligent. It reduces the time as well as discovers the optimized value of the parameter. The optimal value of the kernel parameter σ is determined by the proposed algorithm, i.e., 13.9, when keeping $\gamma = 1$. This optimized value is achieved at the cost of only a few experimentations. This value is not only optimized by the proposed algorithm but also generalized by the use of k-fold cross validation in the algorithm. The

algorithm used k = 5 for k-fold cross validation, to minimize the generalization error. Through all the experiments, the ranges of the kernel parameter $\sigma \in [13.5, 15]$ and the regularization parameter $\gamma \in [1, 3]$ are obtained using the proposed algorithm on which maximum accuracy rate can be achieved. By using these optimized values of the parameters, the proposed system achieved 100% accuracy while testing different benchmark dataset groups.

The clinical efficiency of the proposed classifier in terms of exactness is proven by the confusion matrix given in Table 5.4 (A: Actual, C: Classified). The Group-1 dataset has actual 28+7 normal images from training and validation datasets respectively, and all are classified as normal by the proposed classifier. Abnormal images (177+43) of a Group-1 dataset are all classified as abnormal by the proposed classifier. Similarly, the Group-2 dataset has actual 27+100 normal images from training and validation datasets respectively, and the proposed system classified them all as normal. Abnormal images section of the Group-2 dataset has total of 53+160 (training + validation) images and all these and all these images are examined by the proposed technique and classified as abnormal.

Table 5.4: Confusion matrix of the proposed classifier.

Group-1 Dataset		**Normal (C)**	**Abnormal (C)**
	Normal (A)	28+7[4]	0
	Abnormal (A)	0	177+43[4]
Group-2 Dataset		**Normal (C)**	**Abnormal (C)**
	Normal (A)	27+100[4]	0
	Abnormal (A)	0	53+160[4]

[4] Images (Training + Validation)

The performance of this work was compared with the latest 14 state-of-the-art brain MRI classification techniques, which were examined for the same MRI datasets (axial plane) and on the same platform. The proposed classifier was trained by axial plane brain MR images. However, different slices or planes (location) can also be classified by changing the training datasets accordingly. The comparison results were gathered from these experiments, as presented in Table 5.5. It indicates that the highest accuracy is obtained by the proposed classifier system than the existing techniques. It can be observed that the accuracy of RT + PCA + LS-SVM + RBF (Das et al., 2013) is gradually decreasing when huge and versatile datasets are used, although it uses complex feature decomposition method. DWT + SOM (Chaplot et al., 2006) has the worst performance in terms of accuracy. The schemes proposed in (El-Dahshan et al., 2010), use the lowest feature dimension (lesser than the proposed technique) for classification. However, Table 5.5 reveals that the proposed classifier in (El-Dahshan et al., 2010), is less efficient than the proposed design in terms of correctness rate.

The feature dimension of (Chaplot et al., 2006), (4761 feature/image), is the worst case and also leads to the computationally complex system. The methods described in (Zhang, Dong, et al., 2011; Zhang et al., 2010; Zhang & Wu, 2012; Zhang, Wu, et al., 2011), use low features and show improved results in brain MRI classification. But, these methods are computationally complex because of using various complex weight optimization techniques. However, the proposed scheme requires only 8 feature vectors and obtains better accuracy. Moreover, the memory consumption is also reduced using only 8 dimensional feature vectors, which increases the efficiency of the classifier. The reconstructed images of the proposed architecture (in CHAPTER 3) are also correctly classified by the proposed classifier, so it also provides evidence that the reconstructed images from the under-sampled data using the proposed SENSE architecture can be classified by such automated medical decision support systems.

Table 5.5: Performance comparison using two different dataset groups.

Scheme	Feature Dimension	Accuracy (%)	
		Group-1	Group-2
DWT + SOM (Chaplot et al., 2006)	4761	91.65	88.23
DWT + SVM + linear (LIN) (Chaplot et al., 2006)	4761	94.05	90.29
DWT + SVM + POLY (Chaplot et al., 2006)	4761	96.37	91.18
DWT + SVM + RBF (Chaplot et al., 2006)	4761	96.18	90.88
DWT + PCA + forward neural network (FNN) (El-Dahshan et al., 2010)	7	95.29	90.59
DWT + PCA + kNN (El-Dahshan et al., 2010)	7	96.79	91.47
DWT + PCA + FNN + adaptive chaotic particle swarm optimization (ACPSO) (Zhang et al., 2010)	19	97.38	94.41
DWT + PCA + FNN + scaled conjugate gradient (SCG) (Zhang, Dong, et al., 2011)	19	97.14	93.53
DWT + PCA + FNN + scaled chaotic artificial bee colony (SCABC) (Zhang, Wu, et al., 2011)	19	97.81	94.71
DWT + PCA + kernel SVM (KSVM) + LIN (Zhang & Wu, 2012)	19	94.29	90.59
DWT + PCA + KSVM + HPOLY (Zhang & Wu, 2012)	19	95.61	91.47
DWT + PCA + KSVM + IPOLY (Zhang & Wu, 2012)	19	97.73	93.53
DWT + PCA + KSVM + GRB (Zhang & Wu, 2012)	19	98.82	94.11
RT + PCA + LS-SVM + RBF (Das et al., 2013)	9	99.39	96.47
Fast DWT + PCA + LS-SVM + RBF (Proposed)	8	100	100

5.3.1 Time Analysis Comparison

One other important performance measure is computation time to evaluate the classifier. The time taken for the LS-SVM parameter optimization is very low and training time is just 0.047s. The parameters of the LS-SVM keep unchanged after training. All 340 images are tested through the proposed classifier and the computation time on all the stages (feature extraction, feature reduction, and classification) is recorded. For each brain MRI of 256×256 size, the proposed system consumes the average computation time in feature extraction, feature reduction, and LS-SVM classification which come out 0.0019s, 0.016s, and 0.0027s, respectively. The recent fastest version of the classifier (Zhang & Wu, 2012) executing on the same platform consumes 0.0068s, 0.017s, and 0.0029s for feature extraction, feature reduction, and SVM classification, respectively. Figure 5.7 shows the obtained results, which indicates that our proposed method improves the computation time by 71%, 3%, and 4% on feature extraction stage, feature reduction stage, and classification stage, respectively. The total average computation time for testing the 256×256 size brain MR image is about 0.02076s (feature extraction time + feature reduction time + classification time), which has a significant impact on clinical decision support systems.

From the above discussed results and comparisons, it is clear that the proposed system has significantly high efficiency among all state-of-the-art literature works. Moreover, it is robust and it consumes lower computation time. It has also been demonstrated that the proposed system works equally efficiently with different sizes of datasets and various disease classes.

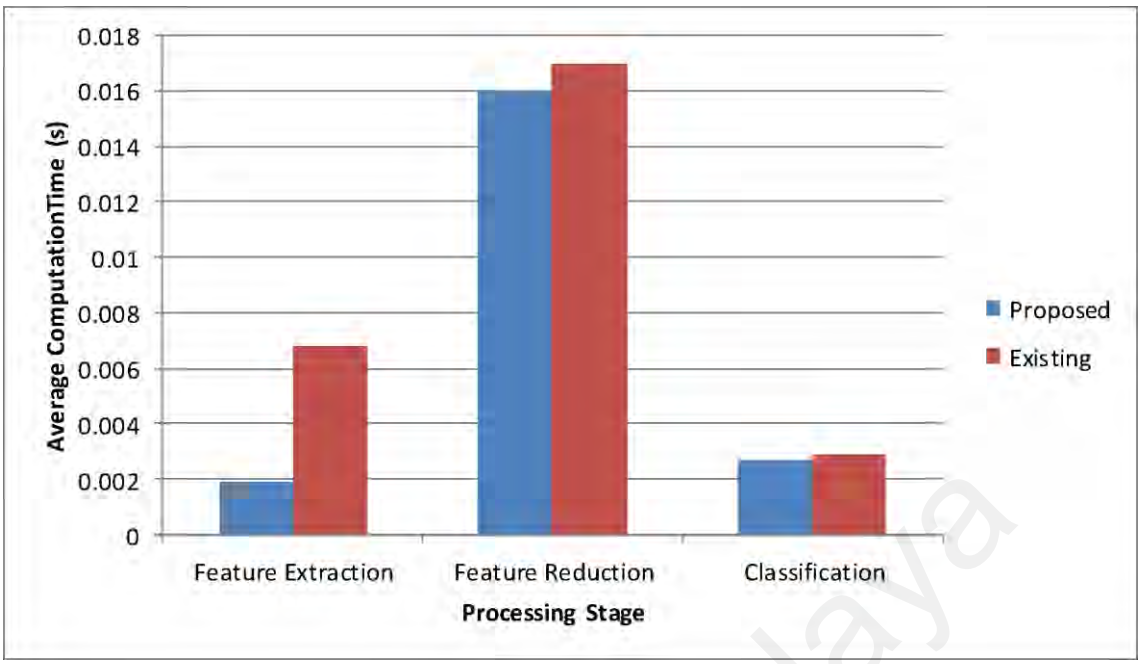


Figure 5.7: Time analysis comparison.

5.4 Summary

In this chapter, the computer based medical decision support system is proposed for the automatic classification of brain MR slices as normal or abnormal. This automated system is designed using fast DWT, PCA, and LS-SVM method, which gives a promising accuracy in classifying the human brain as healthy or diseased. According to the experimental results, the proposed approach yields better performance in terms of the minimum number of principal components used, sensitivity, specificity, classification accuracy, and time analysis, when compared to other popular methods available in the recent literature. The results show that the proposed method is an accurate and robust classifier. The classification performance of this research work on different dataset groups with various diseases shows that it has an impressive generalization capability. The time analysis results prove that the proposed automated intelligent classifier can easily meet the real time diagnosis challenges. The reconstructed images generated by the proposed FPGA based SENSE implementation

are also classified by the classifier. The proposed classifier also classifies the reconstructed images accurately. Furthermore, it is easy to operate, non-invasive, efficient, and computationally inexpensive.

CHAPTER 6: CONCLUSION AND FUTURE WORKS

In this research, the main objective was to investigate and develop a real-time MRI SENSE reconstruction system with classifier to reduce the time required for an MRI and to facilitate the medical practitioner to reach the final decision as early as possible. A novel parameterized FPGA implementation of real-time SENSE reconstruction was presented in this thesis. The results showed a successful implementation of real-time SENSE reconstruction, which led to reducing the computation time, memory usage and transmission cost. The achieved reconstruction results maintained significant values of the artefact power and mean SNR in the reconstructed images. Furthermore, the proposed FPGA based system also provides low power and portable solution as additional features which are very useful for upcoming modern portable MRI scanners. The overall proposed system is also capable to classify the MR images with high accuracy after rapid reconstruction.

The proposed method is computationally efficient solution for reducing the reconstruction time, memory usage and transmission cost. The computation time in the proposed system is significantly lower than the conventional platform implementation (such as CPU or GPU) and it merely consumes $0.054\,ms$ and $0.049\,ms$ for SENSE reconstruction of acceleration factor 2 and 3 datasets, respectively. Moreover, it efficiently manages the memory usage and reduces the transmission cost by on-coil reconstruction. The percentage reduction achieved by the proposed system is 98.6% and 87.5% for memory usage and transmission costs, respectively. Furthermore, the proposed system can reconstruct the images without transferring all the raw data to the memory banks of the MRI workstation, thereby, reducing the coaxial noise.

The proposed FPGA based implementation for real-time SESNE reconstruction (which can be embedded right on the receiver coil data acquisition system) provides a low power and portable platform solution. This platform provides real-time reconstruction before new MRI slice acquisition, which enables the radiographer to check the scan deficiencies (if any) on run-time. Hence,

The new FPGA based real-time SENSE reconstruction system gives promising results on different sensitivity maps (pre-scan and E-maps). The results prove that the proposed design implementation is capable to produce artefact free reconstruction results when using E-maps based sensitivity maps. This technique enables a successful application of SENSE reconstruction where coil sensitivities are hard to estimate by conventional pre-scan method. Smaller values of AP and good mean SNR ensure the quality of the reconstructed images produced by the proposed system.

A new approach for MR image classification was also proposed for validation of the reconstructed images and classification of the brain MR images. It is an optimal solution for brain MRI classification. The results showed higher accuracy for classifying an MRI scan as abnormal or healthy. The proposed algorithm also provides a good alternative to more computationally complex and demanding classification methods. The overall classifier performance indicates that the proposed automated classification tool can be easily integrated in the health care system, which can assist the general practitioners to reach the final decision. Moreover, this intelligent system can be used for classification of images with different pathological conditions, types, and disease status. Finally, the proposed system has the potential to produce accurate reconstructions and classification so it can be used as a significant tool in clinical practice.

6.1 Limitations of the study

This is a novel study for FPGA based real-time SENSE reconstruction with an aim to develop a low power, high throughput and portable system. In literature, not much FPGA based implementation of pMRI reconstruction was available to compare its findings. However, different other platforms (such as multi-core CPU and/or GPU) based implementations for pMRI reconstruction was compared with the proposed system. The proposed system is an application specific parameterized architectural design for SENSE algorithm, which is limited to a maximum of 8 receiver coils and an acceleration factor of 2 and 3. Furthermore, the proposed real-time system has the potential to reconstruct the images right on the receiver coil data acquisition system which is very close to MRI, therefore, appropriate measures must be needed to suppress the interference of the unwanted signals.

In addition, the suggested classification scheme in this thesis has the limitation that it was only validated for brain MR images. However, the proposed intelligent system has the capability to classify any body parts MRI scans, once it is properly trained by appropriate datasets.

6.2 Future Works

In future, this work will be employed for higher acceleration factors as well. The proposed architecture is scalable; so it can be easily developed for higher acceleration factors and higher number of receiver coils as well on upcoming advanced FPGAs. The computational time of the FPGA based SENSE implementation could be further decreased by using advance parallel processing and pipelining techniques, which eventually increases the throughput of the system. Furthermore, other reconstruction algorithms (e.g., GRAPPA, CS and CG-SENSE) can also be modelled in a reconfigurable hardware similar to this work. The combination of CS and pMRI can

also be exploited with such reconfigurable hardware to achieve higher acceleration factors, thus reducing the MRI scan time even further.

REFERENCES

Bauer, S., Wiest, R., Nolte, L.-P., & Reyes, M. (2013). A survey of MRI-based medical image analysis for brain tumor studies. *Physics in Medicine and Biology, 58*(13), R97-R129.

Birk, M., Zapf, M., Balzer, M., Ruiter, N., & Becker, J. (2014). A comprehensive comparison of GPU- and FPGA-based acceleration of reflection image reconstruction for 3D ultrasound computer tomography. *Journal of Real-Time Image Processing, 9*(1), 159-170.

Blaimer, M., Breuer, F., Mueller, M., Heidemann, R. M., Griswold, M. A., & Jakob, P. M. (2004). SMASH, SENSE, PILS, GRAPPA: how to choose the optimal method. *Topics in Magnetic Resonance Imaging, 15*(4), 223-236.

Bydder, M., Larkman, D. J., & Hajnal, J. V. (2002a). Combination of signals from array coils using image-based estimation of coil sensitivity profiles. *Magnetic Resonance in Medicine, 47*(3), 539-548.

Bydder, M., Larkman, D. J., & Hajnal, J. V. (2002b). Generalized SMASH imaging. *Magnetic Resonance in Medicine, 47*(1), 160-170.

Cawley, G. C., & Talbot, N. L. (2004). Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. *Neural Networks, 17*(10), 1467-1475.

Chaplot, S., Patnaik, L., & Jagannathan, N. (2006). Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network. *Biomedical Signal Processing and Control, 1*(1), 86-92.

Chen, G. C., & Juang, C. F. (2013). Object Detection Using Color Entropies and a Fuzzy Classifier. *IEEE Computational Intelligence Magazine, 8*(1), 33-45.

Chiuchişan, I., & Cerlincă, M. (2013). Implementation of Real-Time System for Medical Image Processing using Verilog Hardware Description Language. *Proceedings of the 9th International Conference on Cellular and Molecular Biology, Biophysics and Bioengineering (BIO'13) 5125*, 66-69.

Chu, W., Ong, C. J., & Keerthi, S. S. (2005). An improved conjugate gradient scheme to the solution of least squares SVM. *IEEE Transactions on Neural Networks, 16*(2), 498-501.

Cong, J., Sarkar, V., Reinman, G., & Bui, A. (2011). Customizable domain-specific computing. *IEEE Design & Test of Computers, 28*(2), 6-15.

Cooley, C. Z., Stockmann, J. P., Armstrong, B. D., Sarracanie, M., Lev, M. H., Rosen, M. S., & Wald, L. L. (2015). Two-dimensional imaging in a lightweight portable MRI scanner without gradient coils. *Magnetic Resonance in Medicine, 73*(2), 872-883.

Dalal, I. L., & Fontaine, F. L. (2006). A Reconfigurable FPGA-based 16-Channel Front-End for MRI. *The Fortieth Asilomar Conference on Signals, Systems and Computers* (ACSSC '06) (pp. 1860-1864). Pacific Grove, CA: IEEE.

Das, S., Chowdhury, M., & Kundu, M. K. (2013). Brain MR image classification using multiscale geometric analysis of ripplet. *Progress In Electromagnetics Research, 137*, 1-17.

de Zwart, J. A., van Gelderen, P., Kellman, P., & Duyn, J. H. (2002). Application of sensitivity-encoded echo-planar imaging for blood oxygen level-dependent functional brain imaging. *Magnetic Resonance in Medicine, 48*(6), 1011-1020.

Duan, K., Keerthi, S. S., & Poo, A. N. (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing, 51*, 41-59.

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification.* John Wiley & Sons.

Eklund, A., Dufort, P., Forsberg, D., & LaConte, S. M. (2013). Medical image processing on the GPU – Past, present and future. *Medical Image Analysis, 17*(8), 1073-1094.

El-Dahshan, E.-S. A., Hosny, T., & Salem, A.-B. M. (2010). Hybrid intelligent techniques for MRI brain images classification. *Digital Signal Processing, 20*(2), 433-441.

Faisal, A., Parveen, S., Badsha, S., Sarwar, H., & Reza, A. W. (2013). Computer assisted diagnostic system in tumor radiography. *Journal of Medical Systems, 37*(3), 1-10.

Fan, J. L., & Zhao, F. (2007). Two-dimensional Otsu's curve thresholding segmentation method for gray-Level images. *Dianzi Xuebao(Acta Electronica Sinica), 35*(4), 751-755.

Gebhardt, P., Wehner, J., Weissler, B., Botnar, R., Marsden, P., & Schulz, V. (2016). FPGA-based RF interference reduction techniques for simultaneous PET–MRI. *Physics in Medicine and Biology, 61*(9), 3500.

Griswold, M. A., Heidemann, R. M., & Jakob, P. M. (2003). Direct parallel imaging reconstruction of radially sampled data using GRAPPA with relative shifts. *Proceedings of the 11th Annual Meeting of the ISMRM,* 2349.

Griswold, M. A., Jakob, P. M., Heidemann, R. M., Nittka, M., Jellus, V., Wang, J., . . . Haase, A. (2002). Generalized autocalibrating partially parallel acquisitions (GRAPPA). *Magnetic Resonance in Medicine, 47*(6), 1202-1210.

Griswold, M. A., Jakob, P. M., Nittka, M., Goldfarb, J. W., & Haase, A. (2000). Partially parallel imaging with localized sensitivities (PILS). *Magnetic Resonance in Medicine, 44*(4), 602-609.

Guo, X., Liang, Y., Wu, C., & Wang, C. (2006). PSO-based hyper-parameters selection for LS-SVM classifiers. In *International conference on Neural Information Processing* (pp. 1138-1147). Berlin: Springer.

Hansen, M. S., Atkinson, D., & Sorensen, T. S. (2008). Cartesian SENSE and k-t SENSE reconstruction using commodity graphics hardware. *Magnetic Resonance in Medicine, 59*(3), 463-468.

Hansen, M. S., & Sørensen, T. S. (2013). Gadgetron: an open source framework for medical image reconstruction. *Magnetic Resonance in Medicine, 69*(6), 1768-1776.

Heidemann, R. M., Griswold, M. A., Haase, A., & Jakob, P. M. (2001). VD-AUTO-SMASH imaging. *Magnetic Resonance in Medicine, 45*(6), 1066-1074.

Heidemann, R. M., Griswold, M. A., Seiberlich, N., Krüger, G., Kannengiesser, S. A., Kiefer, B., . . . Jakob, P. M. (2006). Direct parallel image reconstructions for spiral trajectories using GRAPPA. *Magnetic Resonance in Medicine, 56*(2), 317-326.

Holland, D., Liu, C., Song, X., Mazerolle, E., Stevens, M., Sederman, A., . . . Beyea, S. (2013). Compressed sensing reconstruction improves sensitivity of variable density spiral fMRI. *Magnetic Resonance in Medicine, 70*(6), 1634-1643.

Hugger, T., Zahneisen, B., LeVan, P., Lee, K. J., Lee, H.-L., Zaitsev, M., & Hennig, J. (2011). Fast undersampled functional magnetic resonance imaging using nonlinear regularized parallel image reconstruction. *PLoS ONE, 6*(12), e28822.

Jakob, P. M., Grisowld, M. A., Edelman, R. R., & Sodickson, D. K. (1998). AUTO-SMASH: a self-calibrating technique for SMASH imaging. *Magnetic Resonance Materials in Physics Biology and Medicine, 7*(1), 42-54.

Ji, J. X., Son, J. B., & Rane, S. D. (2007). PULSAR: A Matlab toolbox for parallel magnetic resonance imaging using array coils and multiple channel receivers. *Concepts in Magnetic Resonance Part B, 31*(1), 24-36.

Klöppel, S., Stonnington, C. M., Barnes, J., Chen, F., Chu, C., Good, C. D., . . . Roberts, C. C. (2008). Accuracy of dementia diagnosis—a direct comparison between radiologists and a computerized method. *Brain, 131*(11), 2969-2974.

Kose, K., & Haishi, T. (2011). High resolution NMR imaging using a high field yokeless permanent magnet. *Magnetic Resonance in Medicine Sci, 10*(3), 159-167.

Kowalik, G. T., Steeden, J. A., Pandya, B., Odille, F., Atkinson, D., Taylor, A., & Muthurangu, V. (2012). Real-time flow with fast GPU reconstruction for continuous assessment of cardiac output. *Journal of Magnetic Resonance Imaging, 36*(6), 1477-1482.

Kressler, B., Spincemaille, P., Prince, M. R., & Wang, Y. (2006). Reduction of reconstruction time for time-resolved spiral 3D contrast-enhanced magnetic

resonance angiography using parallel computing. *Magnetic Resonance in Medicine, 56*(3), 704-708.

Kumar, P., Sharma, N., & Rana, A. (2012). Handwritten Character Recognition using Different Kernel based SVM Classifier and MLP Neural Network (A COMPARISON). *International Journal of Computer Applications, 53*(11), 25-31.

Kyriakos, W. E., Panych, L. P., Kacher, D. F., Westin, C. F., Bao, S. M., Mulkern, R. V., & Jolesz, F. A. (2000). Sensitivity profiles from an array of coils for encoding and reconstruction in parallel (SPACE RIP). *Magnetic Resonance in Medicine, 44*(2), 301-308.

Larkman, D. J., & Nunes, R. G. (2007). Parallel magnetic resonance imaging. *Physics in Medicine and Biology, 52*(7), R15-R55.

Lendasse, A., Ji, Y., Reyhani, N., & Verleysen, M. (2005). LS-SVM hyperparameter selection with a nonparametric noise estimator. In *International conference on Artificial Neural Networks: Formal Models and Their Applications* (pp. 625-630). Berlin: Springer.

Li, L., & Wyrwicz, A. M. (2015). Design of an MR image processing module on an FPGA chip. *Journal of Magnetic Resonance, 255*, 51-58.

Li, S., Chan, C., Stockmann, J. P., Tagare, H., Adluru, G., Tam, L. K., . . . Peters, D. C. (2015). Algebraic reconstruction technique for parallel imaging reconstruction of undersampled radial data: Application to cardiac cine. *Magnetic Resonance in Medicine, 73*(4), 1643-1653.

Liang, D., Liu, B., Wang, J., & Ying, L. (2009). Accelerating SENSE using compressed sensing. *Magnetic Resonance in Medicine, 62*(6), 1574-1584.

Lustig, M., Donoho, D. L., & Pauly, J. M. (2007). Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine, 58*(6), 1182-1195.

Lustig, M., Donoho, D. L., Santos, J. M., & Pauly, J. M. (2008). Compressed sensing MRI. *IEEE Signal Processing Magazine, 25*(2), 72-82.

Lustig, M., & Pauly, J. M. (2010). SPIRiT: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space. *Magnetic Resonance in Medicine, 64*(2), 457-471.

Magalhães, F., Sousa, R., Araújo, F. M., & Correia, M. V. (2013). Compressive Sensing Based Face Detection without Explicit Image Reconstruction Using Support Vector Machines. In *International Conference Image Analysis and Recognition* (pp. 758-765). Berlin: Springer.

Maitra, M., & Chatterjee, A. (2006). A Slantlet transform based intelligent system for magnetic resonance brain image classification. *Biomedical Signal Processing and Control, 1*(4), 299-306.

Maji, P., Chanda, B., Kundu, M. K., & Dasgupta, S. (2007). Deformation correction in brain MRI using mutual information and genetic algorithm. In *International Conference on Computing: Theory and Applications, 2007. ICCTA'07* (pp. 372-376). Kolkata: IEEE.

Matoug, S., Abdel-Dayem, A., Passi, K., Gross, W., & Alqarni, M. (2012). Predicting Alzheimer's disease by classifying 3D-Brain MRI images using SVM and other well-defined classifiers. In *Journal of Physics: Conference Series 341*, 012019. IOP Publishing.

McKenzie, C. A., Yeh, E. N., Ohliger, M. A., Price, M. D., & Sodickson, D. K. (2002). Self-calibrating parallel imaging with automatic coil sensitivity extraction. *Magnetic Resonance in Medicine, 47*(3), 529-538.

McKhann, G., Drachman, D., Folstein, M., Katzman, R., Price, D., & Stadlan, E. M. (1984). Clinical diagnosis of Alzheimer's disease Report of the NINCDS-ADRDA Work Group* under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease. *Neurology, 34*(7), 939.

McRobbie, D. W., Moore, E. A., Graves, M. J., & Prince, M. R. (2006). *MRI from Picture to Proton.* New York, NY: Cambridge University Press.

Mwangi, B., Ebmeier, K. P., Matthews, K., & Steele, J. D. (2012). Multi-centre diagnostic classification of individual structural neuroimaging scans from patients with major depressive disorder. *Brain, 135*(5), 1508-1521.

Nam, S., Hong, S. N., Akçakaya, M., Kwak, Y., Goddu, B., Kissinger, K. V., . . . Nezafat, R. (2014). Compressed sensing reconstruction for undersampled breath-hold radial cine imaging with auxiliary free-breathing data. *Journal of Magnetic Resonance Imaging, 39*(1), 179-188.

Omer, H. (2012). *Parallel MRI: Tools and Applications.* (PhD Dissertation). Imperial College, London.

Omer, H., & Dickinson, R. (2010). A graphical generalized implementation of SENSE reconstruction using Matlab. *Concepts in Magnetic Resonance Part A, 36*(3), 178-186.

Otazo, R., Kim, D., Axel, L., & Sodickson, D. K. (2010). Combination of compressed sensing and parallel imaging for highly accelerated first-pass cardiac perfusion MRI. *Magnetic Resonance in Medicine, 64*(3), 767-776.

Patil, N., Shelokar, P., Jayaraman, V., & Kulkarni, B. (2005). Regression models using pattern search assisted least square support vector machines. *Chemical Engineering Research and Design, 83*(8), 1030-1037.

Pawar, K., Egan, G., & Zhang, J. (2015). Multichannel Compressive Sensing MRI Using Noiselet Encoding. *PLoS ONE, 10*(5), e0126386.

Pereira, K., Athanas, P., Lin, H., & Feng, W. (2011). Spectral Method Characterization on FPGA and GPU Accelerators. In *International Conference on*

Reconfigurable Computing and FPGAs (ReConFig) (pp. 487-492). Cancun: IEEE.

Prasad, P. V. (2006). *Magnetic resonance imaging: methods and biologic applications*. Humana Press.

Pratx, G., & Xing, L. (2011). GPU computing in medical physics: A review. *Medical Physics, 38*(5), 2685-2697.

Pruessmann, K. P., Weiger, M., Börnert, P., & Boesiger, P. (2001). Advances in sensitivity encoding with arbitrary k-space trajectories. *Magnetic Resonance in Medicine, 46*(4), 638-651.

Pruessmann, K. P., Weiger, M., Scheidegger, M. B., & Boesiger, P. (1999). SENSE: sensitivity encoding for fast MRI. *Magnetic Resonance in Medicine, 42*(5), 952-962.

Robson, P. M., Grant, A. K., Madhuranthakam, A. J., Lattanzi, R., Sodickson, D. K., & McKenzie, C. A. (2008). Comprehensive quantification of signal-to-noise ratio and g-factor for image-based and k-space-based parallel imaging reconstructions. *Magnetic Resonance in Medicine, 60*(4), 895-907.

Sarty, G. E. (2015). Cyclic generalized projection MRI. *Magnetic Resonance Imaging, 33*(3), 304-311.

Saybasili, H., Herzka, D. A., Seiberlich, N., & Griswold, M. A. (2014). Real-time imaging with radial GRAPPA: Implementation on a heterogeneous architecture for low-latency reconstructions. *Magnetic Resonance Imaging, 32*(6), 747-758.

Saybasili, H., Kellman, P., Griswold, M. A., Derbyshire, J. A., & Guttman, M. A. (2009). HTGRAPPA: Real-time B1-weighted image domain TGRAPPA reconstruction. *Magnetic Resonance in Medicine, 61*(6), 1425-1433.

Scapaticci, R., Di Donato, L., Catapano, I., & Crocco, L. (2012). A feasibility study on microwave imaging for brain stroke monitoring. *Progress In Electromagnetics Research B, 40*, 305-324.

Seiberlich, N., Ehses, P., Duerk, J., Gilkeson, R., & Griswold, M. (2011). Improved radial GRAPPA calibration for real-time free-breathing cardiac imaging. *Magnetic Resonance in Medicine, 65*(2), 492-505.

Seiberlich, N., Lee, G., Ehses, P., Duerk, J. L., Gilkeson, R., & Griswold, M. (2011). Improved temporal resolution in cardiac imaging using through-time spiral GRAPPA. *Magnetic Resonance in Medicine, 66*(6), 1682-1688.

Sengur, A. (2008). An expert system based on principal component analysis, artificial immune system and fuzzy k-NN for diagnosis of valvular heart diseases. *Computers in Biology and Medicine, 38*(3), 329-338.

Shahzad, H., Sadaqat, M., Hassan, B., Abbasi, W., & Omer, H. (2016). Parallel MRI Reconstruction Algorithm Implementation on GPU. *Applied Magnetic Resonance, 47*(1), 1-9.

Sodickson, D. K. (2000). Tailored SMASH image reconstructions for robust in vivo parallel MR imaging. *Magnetic Resonance in Medicine, 44*(2), 243-251.

Sodickson, D. K., & Manning, W. J. (1997). Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays. *Magnetic Resonance in Medicine, 38*(4), 591-603.

Stone, S. S., Haldar, J. P., Tsao, S. C., Hwu, W.-m., Sutton, B. P., & Liang, Z.-P. (2008). Accelerating advanced MRI reconstructions on GPUs. *Journal of Parallel and Distributed Computing, 68*(10), 1307-1318.

Strang, G. (2009). *Introduction to Linear Algebra* (4th ed.). Massachusetts: Wellesley Cambridge Press.

Sundararajan, P. (2010). High Performance Computing Using FPGAs. Retrieved 01-Jan-2016, from http://www.xilinx.com/support /documentation/ white _papers/wp375_HPC_Using_FPGAs.pdf

Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters, 9*(3), 293-300.

Tang, W., Sun, H., & Wang, W. (2012). A digital receiver module with direct data acquisition for magnetic resonance imaging systems. *Review of Scientific Instruments, 83*(10), 104701.

Tayler, A. B., Holland, D. J., Sederman, A. J., & Gladden, L. F. (2011). Time resolved velocity measurements of unsteady systems using spiral imaging. *Journal of Magnetic Resonance, 211*(1), 1-10.

Uecker, M., Hohage, T., Block, K. T., & Frahm, J. (2008). Image reconstruction by regularized nonlinear inversion—joint estimation of coil sensitivities and image content. *Magnetic Resonance in Medicine, 60*(3), 674-682.

Uecker, M., Lai, P., Murphy, M. J., Virtue, P., Elad, M., Pauly, J. M., . . . Lustig, M. (2014). ESPIRiT—an eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA. *Magnetic Resonance in Medicine, 71*(3), 990-1001.

Van Gestel, T., Suykens, J. A., Baesens, B., Viaene, S., Vanthienen, J., Dedene, G., . . . Vandewalle, J. (2004). Benchmarking least squares support vector machine classifiers. *Machine Learning, 54*(1), 5-32.

Vapnik, V. (2000). *The nature of statistical learning theory*: Springer Science & Business Media.

Walsh, D. O., Gmitro, A. F., & Marcellin, M. W. (2000). Adaptive reconstruction of phased array MR imagery. *Magnetic Resonance in Medicine, 43*(5), 682-690.

Walther, J. S. (1971). A unified algorithm for elementary functions. In *Proceedings of spring joint computer conference* (pp. 379-385). New York: ACM.

Wang, B., Wu, T., Yan, F., Li, R., Xu, N., & Wang, Y. (2009). RankBoost Acceleration on both NVIDIA CUDA and ATI Stream platforms. In *International Conference on Parallel and Distributed Systems (ICPADS)* (pp. 284-291). Shenzhen: IEEE.

Wang, F.-F., & Zhang, Y.-R. (2011). The support vector machine for dielectric target detection through a wall. *Progress In Electromagnetics Research Letters, 23*, 119-128.

Wang, J., Kluge, T., Nittka, M., Jellus, V., Kuehn, B., & Kiefer, B. (2001). *Parallel acquisition techniques with modified SENSE reconstruction mSENSE.* Paper presented at the Proceedings of the First Würzburg Workshop on Parallel Imaging Basics and Clinical Applications, Würzburg, Germany.

Wang, Y., He, Y., Shan, Y., Wu, T., Wu, D., & Yang, H. (2010). Hardware computing for brain network analysis. In *2nd Asia Symposium on Quality Electronic Design (ASQED)* (pp. 219-222). Penang: IEEE.

Weissler, B., Gebhardt, P., Lerche, C. W., Wehner, J., Solf, T., Goldschmidt, B., . . . Perkuhn, M. (2014). MR compatibility aspects of a silicon photomultiplier-based PET/RF insert with integrated digitisation. *Physics in Medicine and Biology, 59*(17), 5119.

Westbrook, C. (2013). *Handbook of MRI technique.* John Wiley & Sons.

Xu, N.-Y., Cai, X.-F., Gao, R., Zhang, L., & Hsu, F.-H. (2007). Fpga-based accelerator design for rankboost in Web search engines. In *International Conference on Field-Programmable Technology (ICFPT)* (pp. 33-40). Kitakyushu: IEEE.

Yamamoto, S., Watabe, T., Watabe, H., Aoki, M., Sugiyama, E., Imaizumi, M., . . . Hatazawa, J. (2011). Simultaneous imaging using Si-PM-based PET and MRI for development of an integrated PET/MRI system. *Physics in Medicine and Biology, 57*(2), N1.

Yeh, E. N., McKenzie, C. A., Ohliger, M. A., & Sodickson, D. K. (2005). Parallel magnetic resonance imaging with adaptive radius in k-space (PARS): Constrained image reconstruction using k-space locality in radiofrequency coil encoded data. *Magnetic Resonance in Medicine, 53*(6), 1383-1392.

Ying, L., & Sheng, J. (2007). Joint image reconstruction and sensitivity estimation in SENSE (JSENSE). *Magnetic Resonance in Medicine, 57*(6), 1196-1202.

Zhang, W., Li, C., & Zhong, B. (2009). LSSVM parameters optimizing and non-linear system prediction based on cross validation. In *Fifth International Conference on Natural Computation (ICNC'09)* (pp. 531-535). Tianjin: IEEE.

Zhang, Y., Dong, Z., Wu, L., & Wang, S. (2011). A hybrid method for MRI brain image classification. *Expert Systems With Applications, 38*(8), 10049-10053.

Zhang, Y., Wang, S., & Wu, L. (2010). A novel method for magnetic resonance brain image classification based on adaptive chaotic PSO. *Progress In Electromagnetics Research, 109*, 325-343.

Zhang, Y., & Wu, L. (2011). Crop Classification by forward neural network with adaptive chaotic particle swarm optimization. *Sensors, 11*(5), 4721-4743.

Zhang, Y., & Wu, L. (2012). An MR brain images classifier via principal component analysis and kernel support vector machine. *Progress In Electromagnetics Research, 130*, 369-388.

Zhang, Y., Wu, L., & Wang, S. (2011). Magnetic resonance brain image classification by an improved artificial bee colony algorithm. *Progress In Electromagnetics Research, 116*, 65-79.

Zhuo, L., & Prasanna, V. K. (2005). Sparse matrix-vector multiplication on FPGAs. In *Proceedings of the 2005 ACM/SIGDA 13th international symposium on Field-programmable gate arrays* (pp. 63-74). California: ACM.

Zotev, V. S., Volegov, P. L., Matlashov, A. N., Espy, M. A., Mosher, J. C., & Kraus Jr, R. H. (2008). Parallel MRI at microtesla fields. *Journal of Magnetic Resonance, 192*(2), 197-208.

LIST OF PUBLICATIONS

Journal Papers:

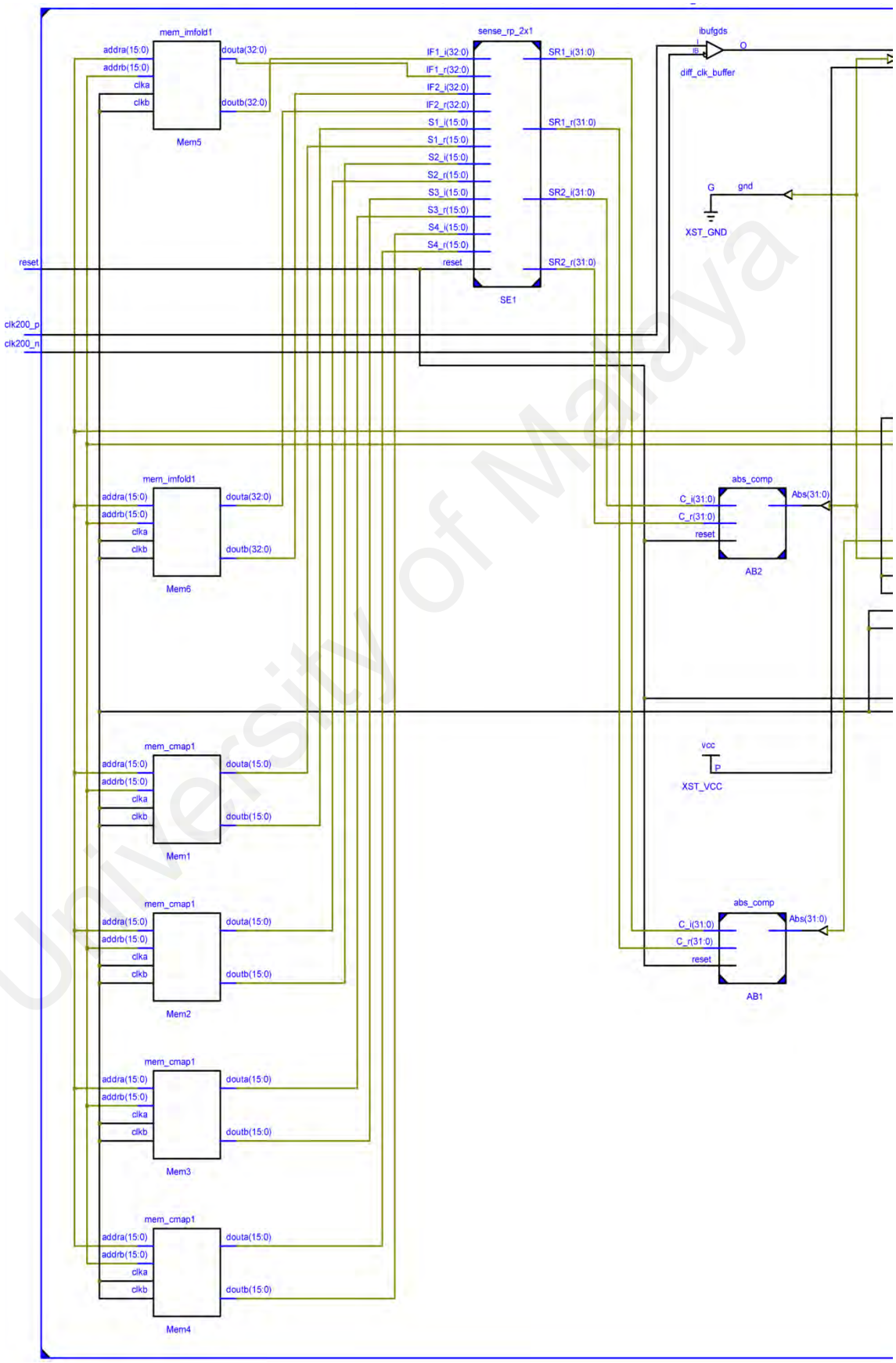
1. **Siddiqui, M. F.**, Reza, A. W., & Kanesan, J. (2015). An Automated and Intelligent Medical Decision Support System for Brain MRI Scans Classification. *PloS one*, *10*(8), e0135875.
2. **Siddiqui, M. F.,** Reza, A. W., Kanesan, J., & Ramiah, H. (2014). Investigation of a novel common subexpression elimination method for low power and area efficient DCT architecture. *The Scientific World Journal*, *2014*.
3. **Siddiqui, M. F.**, Reza, A. W., Shahzad, H., Omer, H., & Kanesan, J. (2016). Implementation of a real-time SENSE reconstruction on an FPGA. Journal of Real-Time Image Processing. (Under Review).
4. **Siddiqui, M. F.**, Reza, A. W., Shafique, A., Omer, H., & Kanesan, J. (2016). FPGA implementation of real-time SENSE reconstruction using pre-scan and E-maps sensitivities. Magnetic Resonance Imaging. (Under Review).
5. **Siddiqui, M. F.**, Sheikh, M., Reza, A. W., & Kanesan, J. (2016). Computer Aided Multi-Class Decision Support System to Diagnose Diseases in Brain MRI. Journal of Biomedical Informatics. (Under Review).

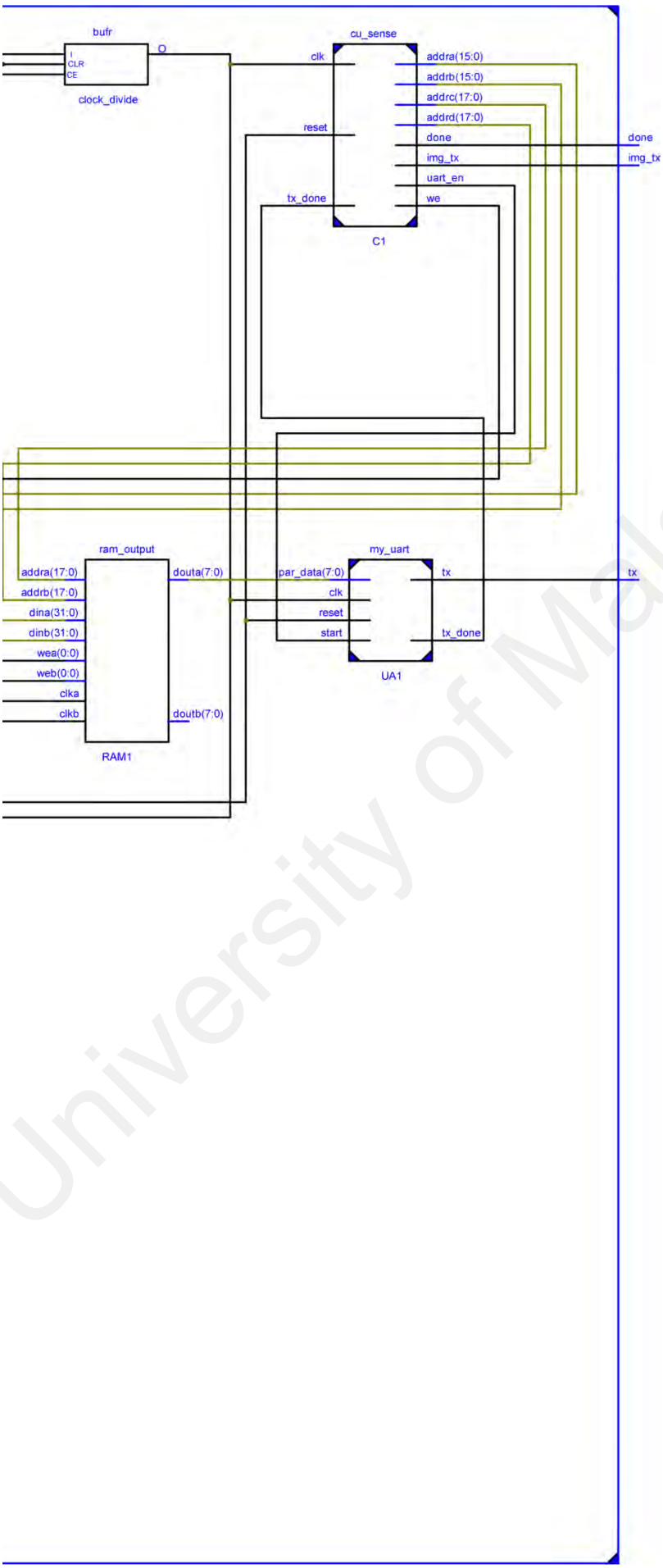
Conferences:

1. **Muhammad Faisal Siddiqui**, Abubakr Shafique, Yousif Rauf Javed, Talha Ahmad Khan, Hamza Naeem Mughal, Ahmed Wasif Reza, Hammad Omer, and Jeevan Kanesan. (2016). Real-time SENSE reconstruction using pre-scan and E-maps sensitivities. In *ISMRM (International Society for Magnetic Resonance in Medicine) 2016 – 24th Annual Meeting and Exhibition*. Singapore: ISMRM.

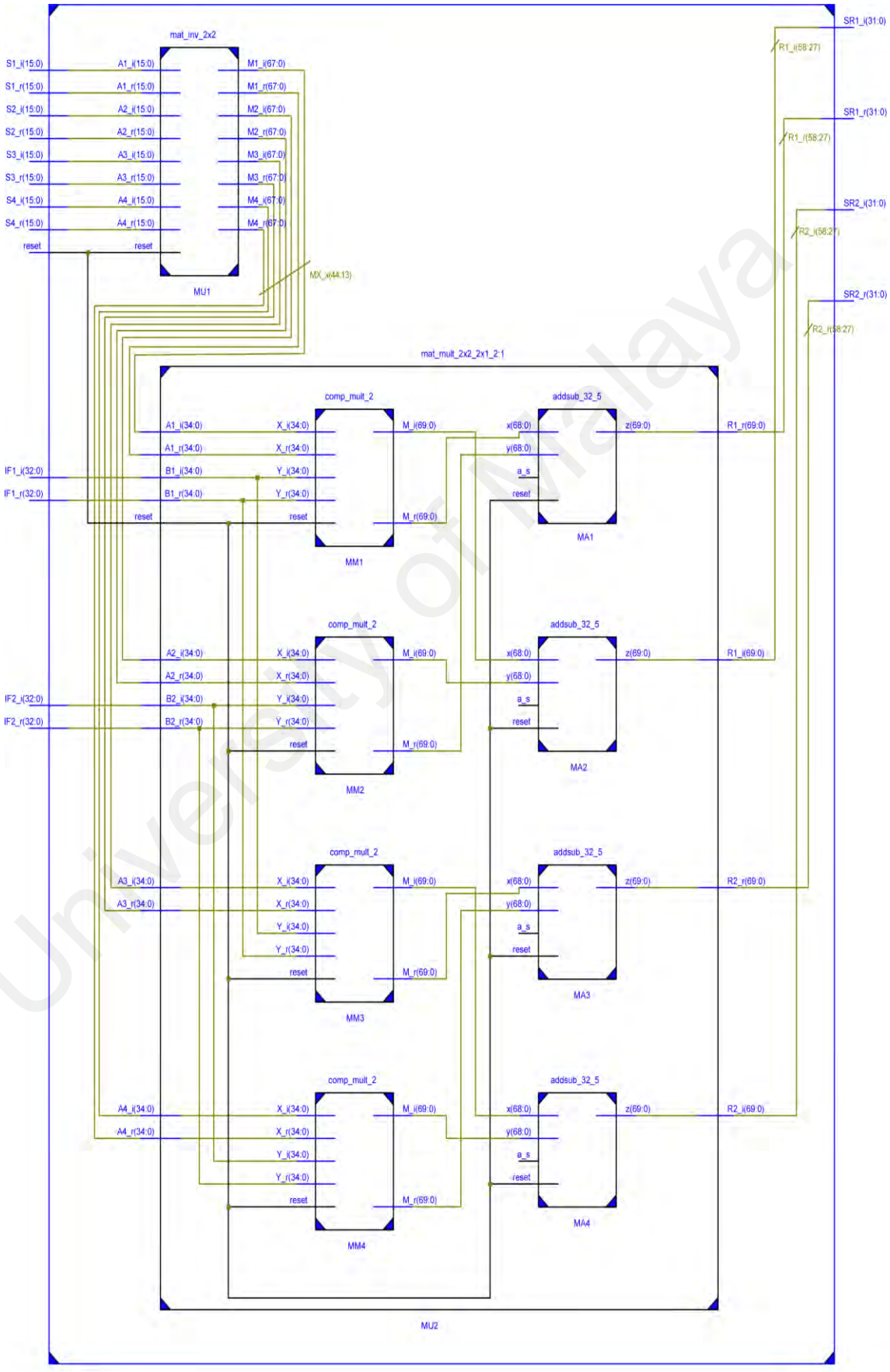
2. **Muhammad Faisal Siddiqui**, Ahmed Wasif Reza, Hammad Omer, , Abubakr Shafique, Yousif Rauf Javed, Talha Ahmad Khan. (2015). Parameterized architecture design of SENSE for real-time reconstruction. In *ESMRMB (European Society for Magnetic Resonance in Medicine and Biology) 2015 Congress - 32nd Annual Scientific Meeting* (pp. S216-S217). Edinburgh/UK: ESMRMB.
3. **Muhammad Faisal Siddiqui**, Ahmed Wasif Reza, Jeevan Kanesan, and Hammad Omer. (2014). A New Parameterized Architectural Design for SENSE Reconstruction. In *3rd International Conference on Computer Engineering & Mathematical Sciences 2014 (ICCEMS 2014)* (pp. 335-338). Langkawi, Malaysia: S&K.

APPENDIX A1 – REGISTER TRANSFER LOGIC (RTL) SCHEMATIC DIAGRAM OF THE PROPOSED SYSTEM (TOP LEVEL)

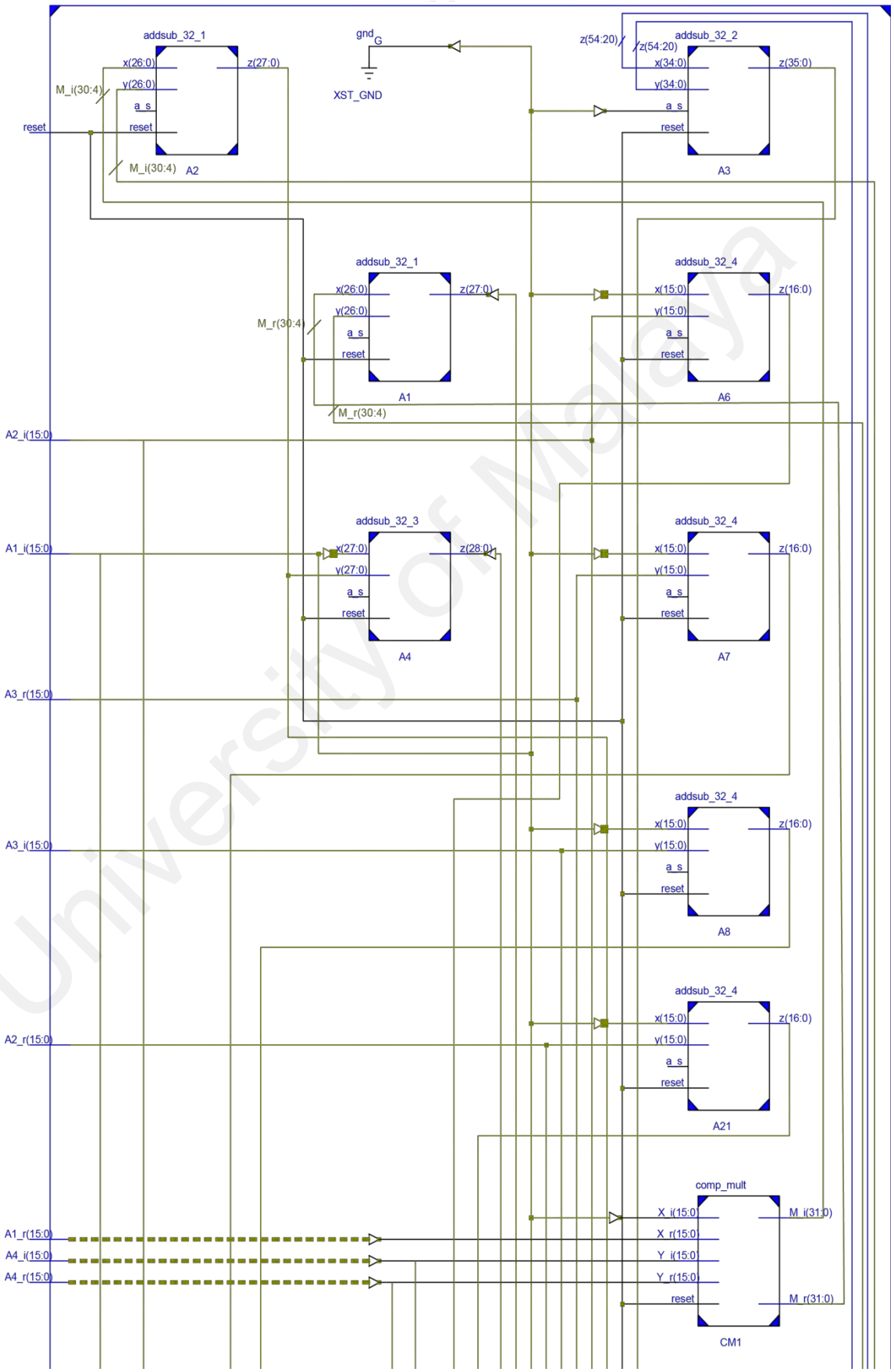


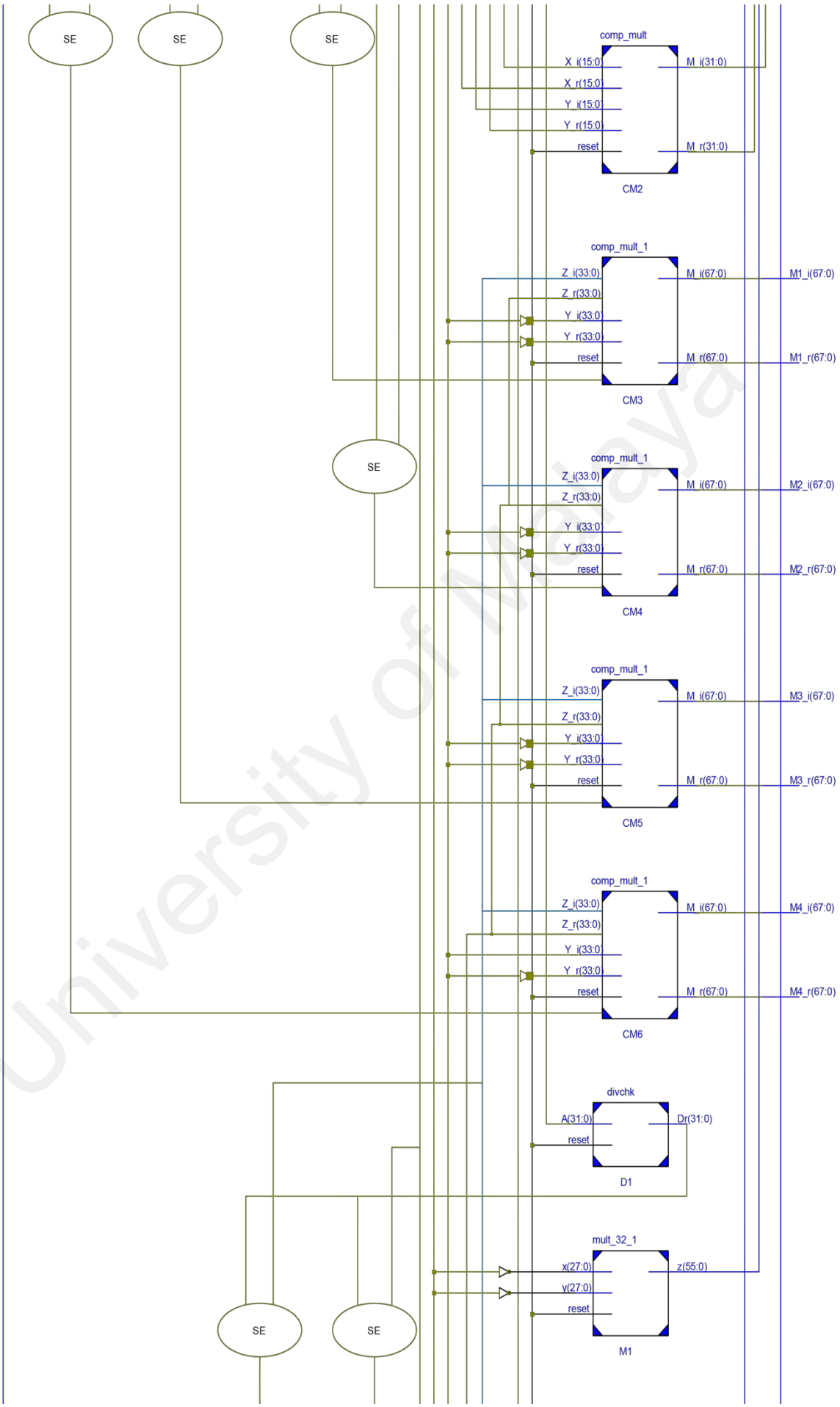


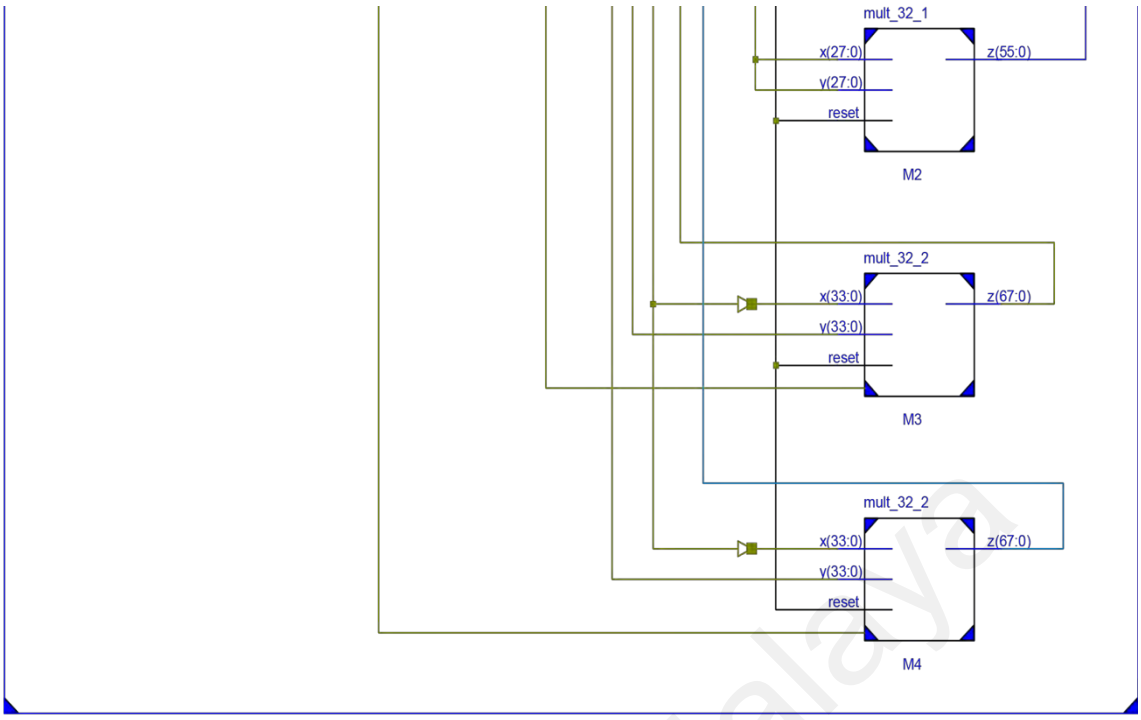
APPENDIX A2 – REGISTER TRANSFER LOGIC (RTL) SCHEMATIC DIAGRAM OF THE PROPOSED SENSE RECONSTRUCTION MODULE



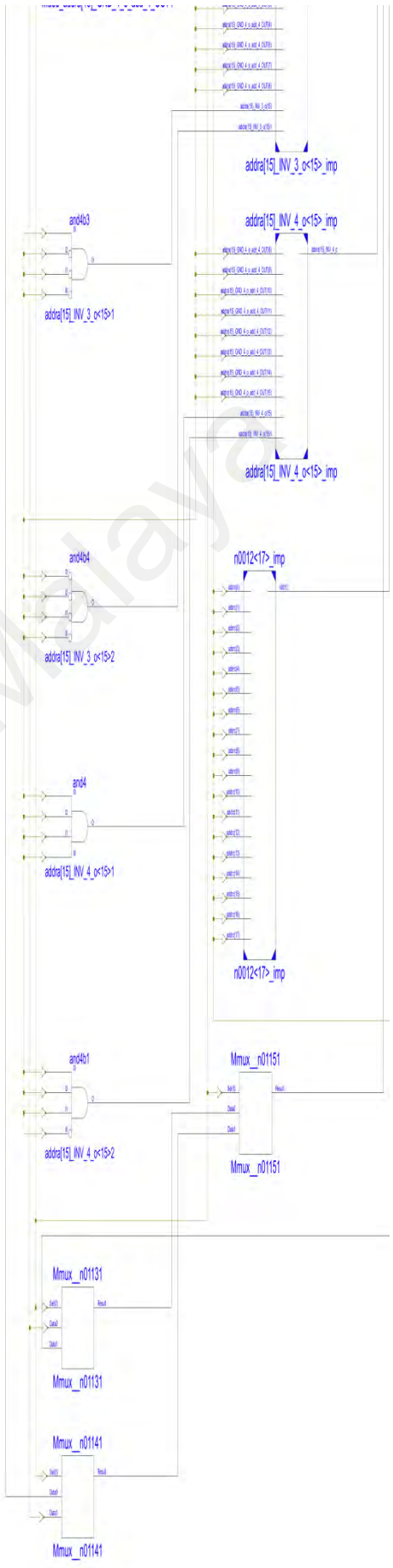
APPENDIX A3 – REGISTER TRANSFER LOGIC (RTL) SCHEMATIC DIAGRAM OF THE PROPOSED MATRIX INVERSION MODULE

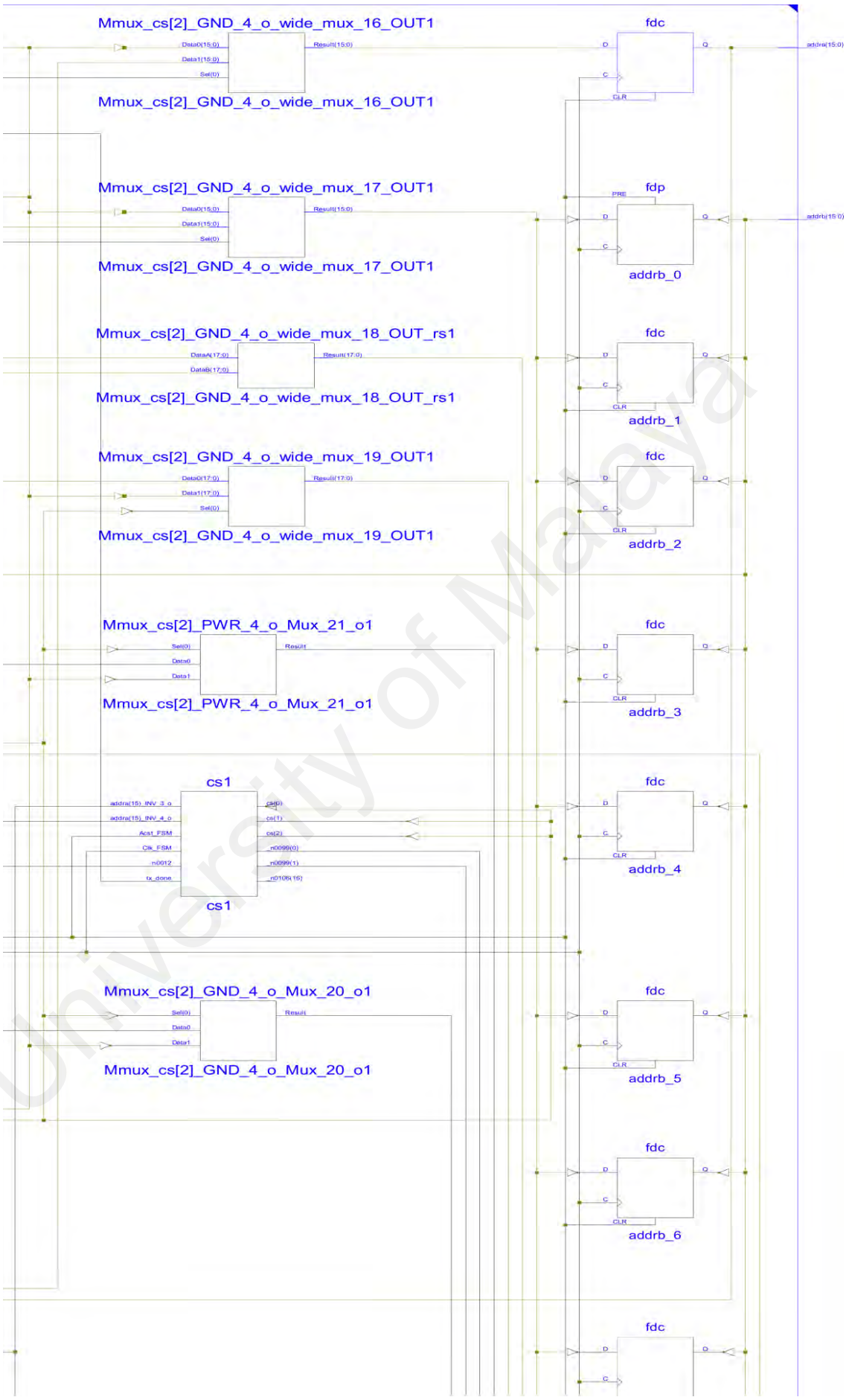


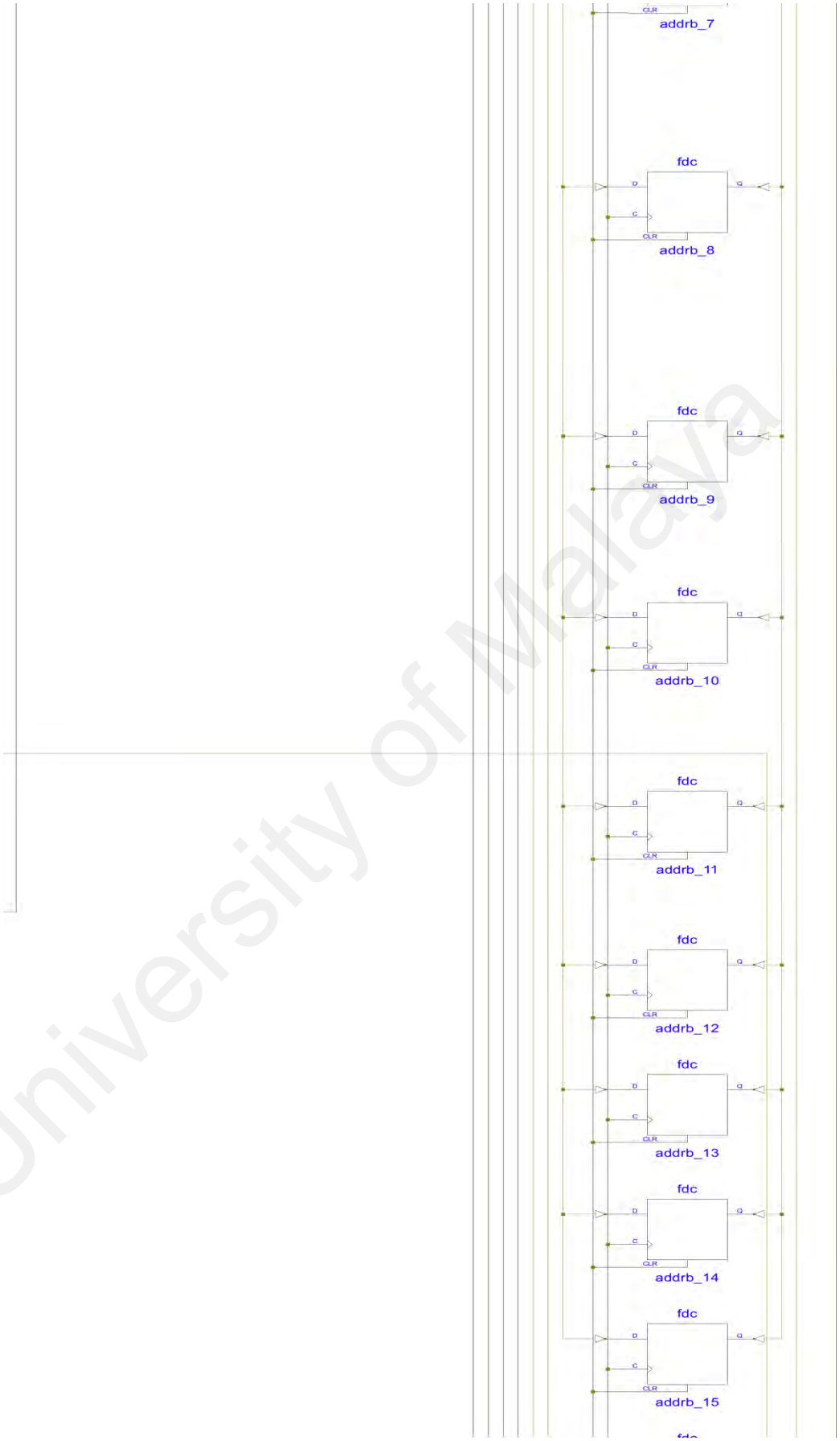


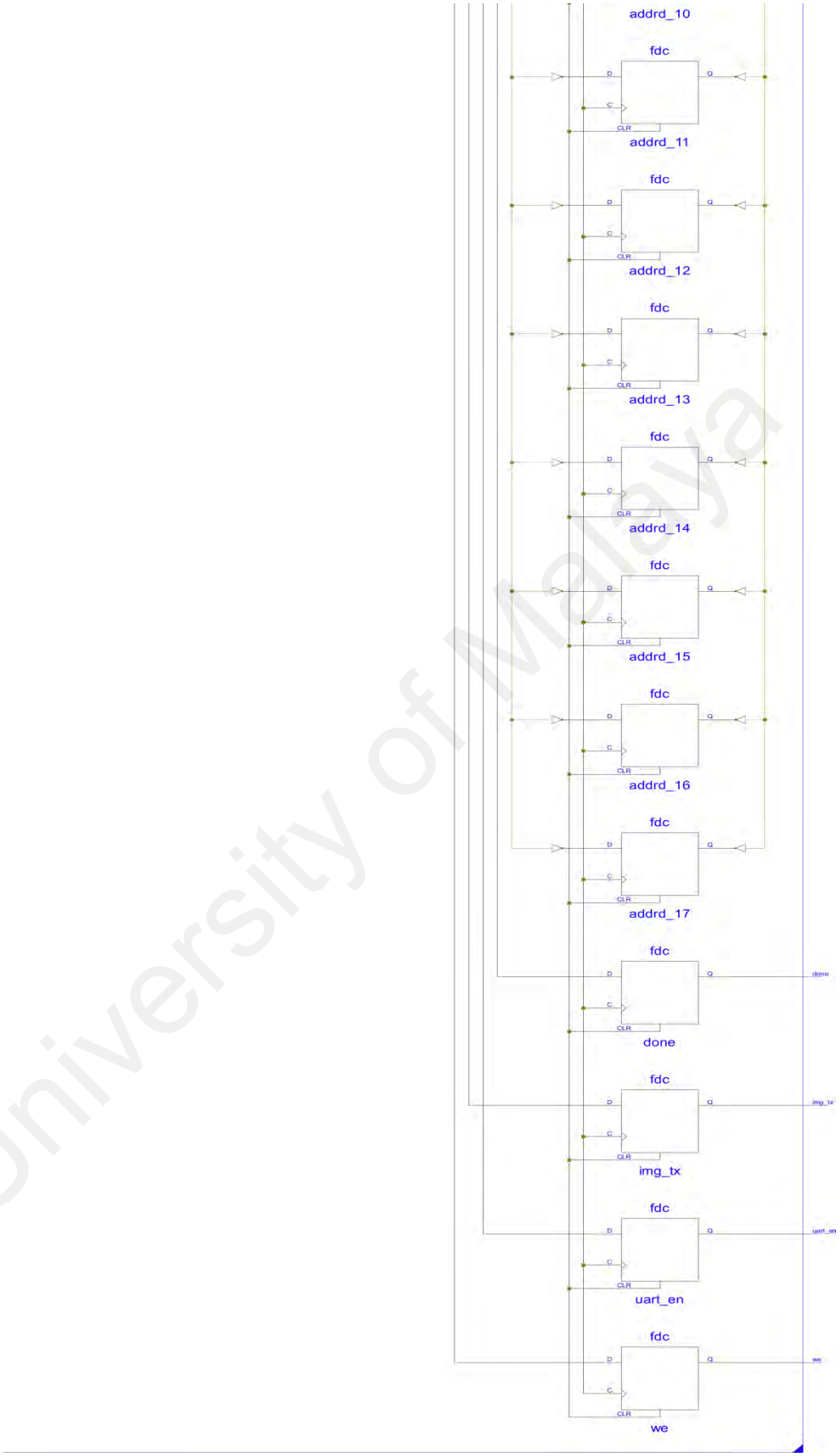


APPENDIX A4 – REGISTER TRANSFER LOGIC (RTL) SCHEMATIC DIAGRAM OF THE PROPOSED ARCHITECTURE









APPENDIX B – VERILOG CODE OF THE PROPOSED SYSTEM

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////
// Module Name:       sense_recon
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////
module sense_recon(tx,done,img_tx,clk200_p,clk200_n,reset);

parameter width= 32;
input reset,clk200_p,clk200_n;
output done,img_tx,tx;
wire [16-1:0]S1_r,S1_i,S2_r,S2_i,S3_r,S3_i,S4_r,S4_i;
wire [width:0]IF1_r,IF1_i,IF2_r,IF2_i;
wire [15 : 0] addra,addrb;
wire [width-1:0]SR1_r,SR1_i,SR2_r,SR2_i;
wire we,uart_en;
wire [17 : 0] addrc,addrd;
//wire [31 : 0] SR1,SR2;
wire [7:0]douta;
wire tx_done;
wire clk,clk200;
IBUFGDS diff_clk_buffer(
      .I(clk200_p),
      .IB(clk200_n),
      .O(clk200));

  // BUFR used to divide by 4 and create a regional clock

  BUFR #(
      .BUFR_DIVIDE("4"),
      .SIM_DEVICE("VIRTEX6"))
  clock_divide (
      .I(clk200),
      .O(clk),
      .CE(1'b1),
      .CLR(1'b0));


cu_sense
C1(we,addra,addrb,addrc,addrd,done,uart_en,img_tx,tx_done,clk,re
set);
//test_u T1(douta,data_in,uart_en,clk,rst);
my_uart UA1(tx,tx_done,douta,uart_en,clk,reset);
sense_rp_2x1
SE1(SR1_r,SR1_i,SR2_r,SR2_i,S1_r,S1_i,S2_r,S2_i,S3_r,S3_i,S4_r,S
4_i,IF1_r,IF1_i,IF2_r,IF2_i,reset);

abs_comp AB1(SR1,SR1_r,SR1_i,reset);
abs_comp AB2(SR2,SR2_r,SR2_i,reset);
```

```
ram_output RAM1(clk,we,addrc,SR1,douta,clk,we,addrd,SR2,);

mem_cmap1 Mem1(clk,addra,S1_r,clk,addrb,S1_i);
mem_cmap1 Mem2(clk,addra,S2_r,clk,addrb,S2_i);
mem_cmap1 Mem3(clk,addra,S3_r,clk,addrb,S3_i);
mem_cmap1 Mem4(clk,addra,S4_r,clk,addrb,S4_i);

mem_imfold1 Mem5(clk,addra,IF1_r,clk,addrb,IF1_i);
mem_imfold1 Mem6(clk,addra,IF2_r,clk,addrb,IF2_i);

endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       cu_sense
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////
module
cu_sense(we,addra,addrb,addrc,addrd,done,uart_en,img_tx,tx_done,
clk,reset);

output [15 : 0] addra,addrb;
output [17 : 0] addrc,addrd;
output we,done,uart_en,img_tx;
input tx_done,clk,reset;

reg [15 : 0] addra,addrb;
reg [17 : 0] addrc,addrd;
reg we,done,uart_en,img_tx;
reg [2:0]cs;

parameter
S0=3'b000,S1=3'b001,S2=3'b010,S3=3'b011,S4=3'b100,S5=3'b101,S6=3
'b110,S7=3'b111;

always @(posedge clk or posedge reset)
begin
     if(reset)
          begin
          addra=16'b0000000000000000;
          addrb=16'b0000000000000001;
          addrc=18'b000000000000000000;
          addrd=18'b000000000000000100;
          we=0; done=0;uart_en=0;img_tx=0;
          cs=S0;
          end
     else
          begin
          case(cs)
          S0:   begin
                    addra=16'b0000000000000000;
                    addrb=16'b0000000000000001;
```

```
                addrc=18'b000000000000000000;
                addrd=18'b000000000000000100;
                we=0; done=0;uart_en=0;img_tx=0;
                cs=S1;
                end
        S1:   begin
                done=0;uart_en=0;img_tx=0;
                addra=addra+2;
                addrb=addrb+2;
                addrc=18'b000000000000000000;
                addrd=18'b000000000000000100;
                if(addra!=16'd64) begin
                      cs=S1; we=0;
                end
                else begin
                      cs=S2; we=1;
                end
                end
        S2:   begin
                addra=addra+2;
                addrb=addrb+2;
                addrc=addrc+8;
                addrd=addrd+8;uart_en=0;img_tx=0;
                if(addra!=16'd65534) begin
                      cs=S2; we=1; done=0;
                end
                else  begin
                      cs=S3; we=0; done=1;
                end
                end
              S3:   begin
                addra=16'b0000000000000000;
                addrb=16'b0000000000000001;
                addrc=18'b000000000000000000;done=1;

addrd=18'b000000000000000000;uart_en=1;img_tx=0;we=0;
                cs=S4;
                end
              S4:   begin
                addra=16'b0000000000000000;
                addrb=16'b0000000000000001;done=1;

addrd=18'b000000000000000000;uart_en=1;we=0;img_tx=0;
                if(tx_done) begin
                      if(addrc!=18'd262143) begin
                            addrc=addrc+1;cs=S4;
                      end
                      else
                            cs=S5;
                end
                else
                      cs=S4;
                end
              S5:   begin
                      addra=16'b0000000000000000;
                      addrb=16'b0000000000000001;
                      addrc=18'b000000000000000000;
                      addrd=18'b000000000000000100;
```

```
                              we=0; done=1;uart_en=0;img_tx=1;
                              cs=S5;
                        end
                  default:   begin
                              addra=16'b0000000000000000;
                              addrb=16'b0000000000000001;
                              addrc=18'b000000000000000000;
                              addrd=18'b000000000000000100;
                              we=0; done=1;uart_en=0;img_tx=1;
                              cs=S5;
                        end

            endcase

      end

end
endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       sense _rp_2x1
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////

module
sense_rp_2x1(SR1_r,SR1_i,SR2_r,SR2_i,S1_r,S1_i,S2_r,S2_i,S3_r,S3
_i,S4_r,S4_i,IF1_r,IF1_i,IF2_r,IF2_i,reset);

parameter width= 32;

output [width-1:0]SR1_r,SR1_i,SR2_r,SR2_i;
input [16-1:0]S1_r,S1_i,S2_r,S2_i,S3_r,S3_i,S4_r,S4_i;
input [width:0]IF1_r,IF1_i,IF2_r,IF2_i;
input reset;

wire [68-1:0]M1_r,M2_r,M3_r,M4_r;
wire [68-1:0]M1_i,M2_i,M3_i,M4_i;
wire [69:0]TSR1_r,TSR1_i,TSR2_r,TSR2_i;

mat_inv_2x2
MU1(M1_r,M1_i,M2_r,M2_i,M3_r,M3_i,M4_r,M4_i,S1_r,S1_i,S2_r,S2_i,
S3_r,S3_i,S4_r,S4_i,reset);

mat_mult_2x2_2x1_2
MU2(TSR1_r,TSR1_i,TSR2_r,TSR2_i,M1_r[61:27],M1_i[61:27],M2_r[61:
27],M2_i[61:27],M3_r[61:27],M3_i[61:27],M4_r[61:27],M4_i[61:27],
{{2{IF1_r[width]}},IF1_r},{{2{IF1_i[width]}},IF1_i},{{2{IF2_r[wi
dth]}},IF2_r},{{2{IF2_i[width]}},IF2_i},reset);

assign SR1_r=TSR1_r[58:27];
assign SR1_i=TSR1_i[58:27]; assign SR2_r=TSR2_r[58:27];
assign SR2_i=TSR2_i[58:27];
```

```
endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       mat_inv_2x2
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////
module
mat_inv_2x2(M1_r,M1_i,M2_r,M2_i,M3_r,M3_i,M4_r,M4_i,A1_r,A1_i,A2
_r,A2_i,A3_r,A3_i,A4_r,A4_i,reset);

parameter width=16;

output [68-1:0]M1_r,M2_r,M3_r,M4_r;
output [68-1:0]M1_i,M2_i,M3_i,M4_i;
input reset;
input [width-1:0]A1_r,A1_i,A2_r,A2_i,A3_r,A3_i,A4_r,A4_i;

wire [31:0]X_r,X_i,Y_r,Y_i;
wire [27:0]W_r,W_i;
wire [width:0]b_mr,b_mi,c_mr,c_mi;
wire [67:0]Z_r,Z_i;
wire [35:0]T3;
wire [28:0]T4;
wire [55:0]r_s,i_s;
wire [32-1:0]Q;

comp_mult CM1(X_r,X_i,A1_r,A1_i,A4_r,A4_i,reset);

comp_mult CM2(Y_r,Y_i,A2_r,A2_i,A3_r,A3_i,reset);

addsub_32 #(27) A1(W_r,X_r[30:4],Y_r[30:4],1'b1,reset);
addsub_32 #(27) A2(W_i,X_i[30:4],Y_i[30:4],1'b1,reset);
mult_32 #(28) M1(r_s,W_r,W_r,reset);

mult_32 #(28) M2(i_s,W_i,W_i,reset);

addsub_32 #(35) A3(T3,r_s[54:20],i_s[54:20],1'b0,reset);
addsub_32 #(28) A4(T4,28'd0,W_i,1'b1,reset);
addsub_32 #(16) A21(b_mr,16'd0,A2_r,1'b1,reset);
addsub_32 #(16) A6(b_mi,16'd0,A2_i,1'b1,reset);
addsub_32 #(16) A7(c_mr,16'd0,A3_r,1'b1,reset);
addsub_32 #(16) A8(c_mi,16'd0,A3_i,1'b1,reset);

divchk #(32) D1(Q,T3[35:4],reset);

mult_32 #(34)
M3(Z_r,{Q[29:0],{4{1'b0}}},{{6{W_r[27]}},W_r[27:0]},reset);
mult_32 #(34)
M4(Z_i,{Q[29:0],{4{1'b0}}},{{5{T4[28]}},T4[28:0]},reset);

#(42)CM3(M1_r,M1_i,Z_r[31:0],Z_i[31:0],{{23{A4_r[15]}},A4_r,{3{1
'b0}}},{{23{A4_i[15]}},A4_i,{3{1'b0}}},reset);
```

```
comp_mult
#(34)CM3(M1_r,M1_i,Z_r[46:13],Z_i[46:13],{{3{A4_r[15]}},A4_r,{15
{1'b0}}},{{3{A4_i[15]}},A4_i,{15{1'b0}}},reset);

comp_mult
#(34)CM4(M2_r,M2_i,Z_r[46:13],Z_i[46:13],{{2{b_mr[15]}},b_mr,{15
{1'b0}}},{{2{b_mi[15]}},b_mi,{15{1'b0}}},reset);

comp_mult
#(34)CM5(M3_r,M3_i,Z_r[46:13],Z_i[46:13],{{2{c_mr[15]}},c_mr,{15
{1'b0}}},{{2{c_mi[15]}},c_mi,{15{1'b0}}},reset);

comp_mult
#(34)CM6(M4_r,M4_i,Z_r[46:13],Z_i[46:13],{{3{A1_r[15]}},A1_r,{15
{1'b0}}},{{3{A1_i[15]}},A1_i,{15{1'b0}}},reset);

endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       mat_mult_2x2_2x1_2
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////

module
mat_mult_2x2_2x1_2(R1_r,R1_i,R2_r,R2_i,A1_r,A1_i,A2_r,A2_i,A3_r,
A3_i,A4_r,A4_i,B1_r,B1_i,B2_r,B2_i,reset);

parameter width=35;

output [2*width-1:0]R1_r,R2_r,R1_i,R2_i;

input reset;
input [width-
1:0]A1_r,A1_i,A2_r,A2_i,A3_r,A3_i,A4_r,A4_i,B1_r,B1_i,B2_r,B2_i;

wire [2*width-
1:0]TM1_r,TM1_i,TM2_r,TM2_i,TM3_r,TM3_i,TM4_r,TM4_i;

comp_mult #(width) MM1(TM1_r,TM1_i,A1_r,A1_i,B1_r,B1_i,reset);

comp_mult #(width) MM2(TM2_r,TM2_i,A2_r,A2_i,B2_r,B2_i,reset);

comp_mult #(width) MM3(TM3_r,TM3_i,A3_r,A3_i,B1_r,B1_i,reset);

comp_mult #(width) MM4(TM4_r,TM4_i,A4_r,A4_i,B2_r,B2_i,reset);

addsub_32 #(2*width-1) MA1(R1_r,TM1_r[2*width-
1:1],TM2_r[2*width-1:1],1'b0,reset);

addsub_32 #(2*width-1) MA2(R1_i,TM1_i[2*width-
1:1],TM2_i[2*width-1:1],1'b0,reset);
```

```
addsub_32 #(2*width-1) MA3(R2_r,TM3_r[2*width-
1:1],TM4_r[2*width-1:1],1'b0,reset);

addsub_32 #(2*width-1) MA4(R2_i,TM3_i[2*width-
1:1],TM4_i[2*width-1:1],1'b0,reset);

endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       abs_comp
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////
module abs_comp(Abs,C_r,C_i,reset);

parameter width=32;

output [width-1:0]Abs;
input reset;
input [width-1:0]C_r,C_i;  //6.26

wire [2*width-1:0]T_r,T_i;
wire [width:0]T_s;

mult_32 #(width) Ma1 (T_r,C_r,C_r,reset);
mult_32 #(width) Ma2 (T_i,C_i,C_i,reset);

addsub_32 #(width) Aa1 (T_s,T_r[60:29],T_i[60:29],1'b0,reset);
sqrt2 Sa1(T_s[width-1:0],Abs);
endmodule

//////////////////////////////////////////////////////////////////
// Module Name:       my_uart
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//////////////////////////////////////////////////////////////////
module my_uart(tx,tx_done,par_data,start,clk,reset);

output tx,tx_done;
input clk,reset,start;
input [7:0]par_data;

ser_tx
S1(.clk(clk),.reset(reset),.start(start),.tx(tx),.par_data(par_d
ata),.tx_done(tx_done));
endmodule
```

```
//////////////////////////////////////////////////////////////////
//
// Module Name:       sense_recon
// Project Name:      MAGNETIC RESONANCE IMAGING SENSE
RECONSTRUCTION SYSTEM USING FPGA
// Target Devices:    Virtex 6 and Virtex 7
//
// Description:       For 8-coils or less
//
// Code Developed by: Muhammad Faisal Siddiqui,
//                    PhD Scholar, University of Malaya,
//                    Kuala Lumpur, Malaysia
//
//////////////////////////////////////////////////////////////////
module sense_recon(SR1_r,SR1_i,SR2_r,SR2_i,clk,reset);

parameter width= 32;

output [32-1:0]SR1_r,SR1_i,SR2_r,SR2_i;
input reset,clk;

wire [16-1:0]S1_r,S1_i,S2_r,S2_i,S3_r,S3_i,S4_r,S4_i;
wire [16-1:0]S5_r,S5_i,S6_r,S6_i,S7_r,S7_i,S8_r,S8_i;
wire [16-1:0]S9_r,S9_i,S10_r,S10_i,S11_r,S11_i,S12_r,S12_i;
wire [16-1:0]S13_r,S13_i,S14_r,S14_i,S15_r,S15_i,S16_r,S16_i;
wire [width+1:0]IF1_r,IF1_i,IF2_r,IF2_i;
wire [width+1:0]IF3_r,IF3_i,IF4_r,IF4_i;
wire [width+1:0]IF5_r,IF5_i,IF6_r,IF6_i;
wire [width+1:0]IF7_r,IF7_i,IF8_r,IF8_i;
wire [15 : 0] addra,addrb;

cu_sense C1(addra,addrb,clk,reset);

Sense_8x2 S1(SR1_r,SR1_i,SR2_r,SR2_i,
S1_r,S1_i,S2_r,S2_i,
S3_r,S3_i,S4_r,S4_i,
S5_r,S5_i,S6_r,S6_i,
S7_r,S7_i,S8_r,S8_i,
S9_r,S9_i,S10_r,S10_i,
S11_r,S11_i,S12_r,S12_i,
S13_r,S13_i,S14_r,S14_i,
S15_r,S15_i,S16_r,S16_i,
IF1_r,IF1_i,IF2_r,IF2_i,
IF3_r,IF3_i,IF4_r,IF4_i,
IF5_r,IF5_i,IF6_r,IF6_i,
IF7_r,IF7_i,IF8_r,IF8_i,
reset);

mem_cmap1 Mem1(clk,addra,S1_r,clk,addrb,S1_i);
mem_cmap2 Mem2(clk,addra,S2_r,clk,addrb,S2_i);
mem_cmap3 Mem3(clk,addra,S3_r,clk,addrb,S3_i);
mem_cmap4 Mem4(clk,addra,S4_r,clk,addrb,S4_i);
mem_cmap5 Mem5(clk,addra,S5_r,clk,addrb,S5_i);
mem_cmap6 Mem6(clk,addra,S6_r,clk,addrb,S6_i);
mem_cmap7 Mem7(clk,addra,S7_r,clk,addrb,S7_i);
mem_cmap8 Mem8(clk,addra,S8_r,clk,addrb,S8_i);

mem_cmap9 Mem9(clk,addra,S9_r,clk,addrb,S9_i);
```

```
mem_cmap10 Mem10(clk,addra,S10_r,clk,addrb,S10_i);
mem_cmap11 Mem11(clk,addra,S11_r,clk,addrb,S11_i);
mem_cmap12 Mem12(clk,addra,S12_r,clk,addrb,S12_i);
mem_cmap13 Mem13(clk,addra,S13_r,clk,addrb,S13_i);
mem_cmap14 Mem14(clk,addra,S14_r,clk,addrb,S14_i);
mem_cmap15 Mem15(clk,addra,S15_r,clk,addrb,S15_i);
mem_cmap16 Mem16(clk,addra,S16_r,clk,addrb,S16_i);


mem_imfold1 MemA(clk,addra,IF1_r,clk,addrb,IF1_i);
mem_imfold2 MemB(clk,addra,IF2_r,clk,addrb,IF2_i);
mem_imfold3 MemC(clk,addra,IF3_r,clk,addrb,IF3_i);
mem_imfold4 MemD(clk,addra,IF4_r,clk,addrb,IF4_i);

mem_imfold5 MemE(clk,addra,IF5_r,clk,addrb,IF5_i);
mem_imfold6 MemF(clk,addra,IF6_r,clk,addrb,IF6_i);
mem_imfold7 MemG(clk,addra,IF7_r,clk,addrb,IF7_i);
mem_imfold8 MemH(clk,addra,IF8_r,clk,addrb,IF8_i);

endmodule
```

APPENDIX C – MATLAB CODE FOR GRAPHICAL USER INTERFACE OF THE PROPOSED CLASSIFIER

```
% Matlab Code for Graphical User Interface of the proposed classifier
% (Developed in Chapter 5)
% Code Developed by:    Muhammad Faisal Siddiqui,
%                       PhD Scholar, Univeristy of Malaya,
%                       Kuala Lumpur, Malaysia

function varargout = Project_1_1_gui(varargin)
% PROJECT_1_1_GUI MATLAB code for Project_1_1_gui.fig
%      PROJECT_1_1_GUI, by itself, creates a new PROJECT_1_1_GUI or
raises the existing
%      PROJECT_1_1_GUI('Property','Value',...) creates a new
PROJECT_1_1_GUI or raises the
%      existing singleton*.  Starting from the left, property value
pairs are
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Project_1_1_gui_OpeningFcn, ...
                   'gui_OutputFcn',  @Project_1_1_gui_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before Project_1_1_gui is made visible.
function Project_1_1_gui_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Project_1_1_gui (see VARARGIN)

% Choose default command line output for Project_1_1_gui


% UIWAIT makes Project_1_1_gui wait for user response (see UIRESUME)
% uiwait(handles.figure1);

list = dir('E:\data\MRI\Project_1_ver1.03-gui\img_dataset\*.gif');
%format of the file
for l = 1:length(list)
img{l} = imread(list(l).name);
% [c{l},s{l}]=wavefast(img{l},3,'haar');    %for all DWT
% [nc,img_a{l}]=wavecut('a',c{l},s{l});          %for approx. part of
DWT
```

```
img_a{l}=my_dwt(double(img{l}));
if(l==1)
    S=cat(3,img_a{l});
else
    S=cat(3,S,img_a{l});
end
end

list2 = dir('E:\data\MRI\Project_1_ver1.03-
gui\img_dataset\non_demented\*.gif');  %format of the file
for m = 1:length(list2)
img{m} = imread(list2(m).name);
% [c{m},s{m}]=wavefast(img{m},3,'haar');    %for all DWT
% [nc,img_a{m}]=wavecut('a',c{m},s{m});          %for approx. part of
DWT
img_a{m}=my_dwt(double(img{m}));
S=cat(3,S,img_a{m});
 l=l+1;
end

[x,r]=imstack2vectors(S);

P_19=princompimg(x,8);  % LS-SVM @ 8 gives 100%

ok_img=m;
%%%%%%%%%%%%%%%%%%

group=-1.*ones(l-ok_img,1);   %Group1 let suppose non-demented images
b=ones(ok_img,1);    %Group2 let suppose demented images
group=[group;b];
train_data=P_19.Y;
train_data=train_data';

%SVMstruct = svmtrain(train_data,group);
SVMstruct =
svmtrain(train_data,group,'Kernel_Function','rbf','rbf_sigma',13.9,'bo
xconstraint',1,'method','LS');
handles.SVMstruct = SVMstruct;
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
%%%%for testing


% --- Outputs from this function are returned to the command line.
function varargout = Project_1_1_gui_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
```

```
% handles    structure with handles and user data (see GUIDATA)



clear list
%%%%for testing
SVMstruct2=handles.SVMstruct;
msg='Test from diseased dataset'
test_diseased=0;
list = dir('E:\data\MRI\Project_1_ver1.03-gui\check_dataset\*.gif');
%format of the file
for l = 1:length(list)
img{l} = imread(list(l).name);
img_a{l}=my_dwt(double(img{l}));
x_test=imstack2vectors(img_a{l});
result = svmclassify(SVMstruct2,x_test');
if (result==-1)
test_diseased=test_diseased+1;
else
    l
end
end
total_img=l;
set(handles.text11,'string',total_img);
test_diseased_correctly_diagnosed=test_diseased;
set(handles.text12,'string',test_diseased_correctly_diagnosed);

clear list
msg='Test from non-diseased dataset'
test_img_ok=0;
list = dir('E:\data\MRI\Project_1_ver1.03-
gui\check_dataset\non_demented\*.gif');  %format of the file
for k = 1:length(list)
img{k} = imread(list(k).name);
img_a{k}=my_dwt(double(img{k}));
x_test=imstack2vectors(img_a{k});
result = svmclassify(SVMstruct2,x_test');
if (result==1)
test_img_ok=test_img_ok+1;
end
end
total_images=k;
set(handles.text7,'string',total_images);
test_non_diseased_correctly_diagnosed=test_img_ok;
set(handles.text8,'string',test_non_diseased_correctly_diagnosed);
% --- Executes during object creation, after setting all properties.
function pushbutton1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
SVMstruct2=handles.SVMstruct;
[filename, pathname] = uigetfile({'*.gif'});
sel_img = imread([pathname filename]);
img_test=my_dwt(double(sel_img));
x_test=imstack2vectors(img_test);
```

```
imshow(sel_img)
result = svmclassify(SVMstruct2,x_test');
if(result==-1)
set(handles.text13,'string','Demented/Diseased MRI' );
else
set(handles.text13,'string','Non-Demented/Non-Diseased MRI' );
end


% --- Executes during object creation, after setting all properties.
function pushbutton2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes during object creation, after setting all properties.
function axes1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: place code in OpeningFcn to populate axes1


% --- Executes during object creation, after setting all properties.
function text7_CreateFcn(hObject, eventdata, handles)  %Num of total
images of correct dataset
% hObject    handle to text7 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes during object creation, after setting all properties.
function text8_CreateFcn(hObject, eventdata, handles)%Num of total
images correct
% hObject    handle to text8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes during object creation, after setting all properties.
function text11_CreateFcn(hObject, eventdata, handles)%Num of total
images of diseased dataset
% hObject    handle to text7 (see GCBO)
% hObject    handle to text11 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes during object creation, after setting all properties.
function text12_CreateFcn(hObject, eventdata, handles)%Num of total
images of diseased
% hObject    handle to text7 (see GCBO)
% hObject    handle to text12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
```

```
% handles    empty - handles not created until after all CreateFcns
called


% --- Executes during object creation, after setting all properties.
function text13_CreateFcn(hObject, eventdata, handles)  %specific
Check
% hObject    handle to text13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called
```