

PERFORMANCE EVALUATION OF MAGNETIC FLUX DENSITY BASED
MAGNETIC RESONANCE ELECTRICAL IMPEDANCE TOMOGRAPHY
RECONSTRUCTION ALGORITHMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÖKHAN EKER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2009

Approval of the Thesis:

**PERFORMANCE EVALUATION OF MAGNETIC FLUX DENSITY
BASED MAGNETIC RESONANCE ELECTRICAL IMPEDANCE
TOMOGRAPHY RECONSTRUCTION ALGORITHMS**

submitted by **GÖKHAN EKER** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İsmet Erkmek
Head of Department, **Electrical and Electronics Engineering** _____

Prof. Dr. B. Murat Eyübođlu
Supervisor, **Electrical and Electronics Engineering Dept., METU** _____

Examining Committee Members

Prof. Dr. Nevzat Güneri Gençer
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. B. Murat Eyübođlu
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Kemal Leblebiciođlu
Electrical and Electronics Engineering Dept., METU _____

Assist. Prof. Dr. Yeşim Serinađaođlu Doğrusöz
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Yusuf Ziya İder
Electrical and Electronics Engineering Dept., Bilkent University _____

Date: 03.09.2009

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Gökhan, EKER

Signature:

ABSTRACT

PERFORMANCE EVALUATION OF MAGNETIC FLUX DENSITY BASED MAGNETIC RESONANCE ELECTRICAL IMPEDANCE TOMOGRAPHY RECONSTRUCTION ALGORITHMS

Eker, Gökhan

MSc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. B. Murat Eyüboğlu

September 2009, 141 pages

Magnetic Resonance Electrical Impedance Tomography (MREIT) reconstructs images of electrical conductivity distribution based on magnetic flux density (\vec{B}) measurements. Magnetic flux density is generated by an externally applied current on the object and measured by a Magnetic Resonance Imaging (MRI) scanner. With the measured data and peripheral voltage measurements, the conductivity distribution of the object can be reconstructed. There are two types of reconstruction algorithms. First type uses current density distributions to reconstruct conductivity distribution. Object must be rotated in MRI scanner to measure three components of magnetic flux density. These types of algorithms are called J-based reconstruction algorithms. The second type of reconstruction algorithms uses only one component of magnetic flux density which is parallel to the main magnetic field of MRI scanner. This eliminates the need of subject rotation. These types of algorithms are called B-based reconstruction algorithms.

In this study four of the B-based reconstruction algorithms, proposed by several research groups, are examined. The algorithms are tested by different computer models for noise-free and noisy data. For noise-free data, the algorithms work successfully. System SNR 30, 20 and 13 are used for noisy data. For noisy data the performance of algorithm is not as satisfactory as noise-free data. Twice differentiation of z component of \vec{B} (B_z) is used for two of the algorithms. These algorithms are very sensitive to noise. One of the algorithms uses only one differentiation of B_z so it is immune to noise. The other algorithm uses sensitivity matrix to reconstruct conductivity distribution.

Keywords: electrical impedance tomography, magnetic resonance imaging, conductivity reconstruction, B based reconstruction algorithms.

ÖZ

MANYETİK AKI YOĞUNLUĞU TABANLI MANYETİK REZONANS ELEKTRİKSEL EMPEDANS TOMOGRAFİSİ GERİÇATIM ALGORİTMALARININ PERFORMANS DEĞERLENDİRMESİ

Eker, Gökhan

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. B. Murat Eyüboğlu

Eylül 2009, 141 sayfa

Manyetik Rezonans Elektrik Empedans Tomografisi (MREET), bir elektriksel iletkenlik dağılımı görüntüsünü manyetik akı yoğunluğu (\vec{B}) ölçümlerine dayalı olarak geri çatismaktadır. Bir objeye dışarıdan akım uygulandığında bir manyetik akı yoğunluğu meydana gelmekte ve bu manyetik akı yoğunluğu Manyetik Rezonans Görüntüleme (MRG) tarayıcısıyla ölçülmektedir. Bir cismin içindeki iletkenlik dağılımı, ölçülen data ve dış kısımdan yapılan voltaj ölçümleri ile geri çatılabilmektedir. İki çeşit geriçatım algoritması bulunmaktadır. Birinci çeşit, iletkenlik dağılımını geri çatmak için akım yoğunluğu dağılımını kullanmaktadır. Bu çeşitte cisim, manyetik akı yoğunluğunun üç bileşenini ölçmek için MRG tarayıcısı içinde döndürülmektedir. Bu çeşit algoritmalara J-tabanlı geriçatım algoritmaları denir. İkinci çeşit geriçatım algoritmaları, manyetik akı yoğunluğunun MRG tarayıcısının ana manyetik alanı ile paralel olan bileşenini kullanır. Bu cismin döndürülmesi gereksinimini devre dışı bırakmaktadır. Bu çeşit algoritmalara B-tabanlı geriçatım algoritmaları denir. Bu çalışmada değişik araştırma gurupları tarafından önerilen dört B-tabanlı geriçatım algoritması

incelenmiştir. Algoritmalar gürültüsüz ve gürültülü durum için değişik bilgisayar modelleriyle test edilmiştir. Gürültüsüz durumda algoritmalar başarıyla çalışmaktadır. Gürültülü durum için sistem SNR'ı 30, 20 ve 13 olarak kabul edilmiştir. Algoritmaların gürültülü durum performansları gürültüsüz durum performansları kadar başarılı değildir. İki algoritmada \vec{B} 'nin z bileşeninin (B_z) iki kez türevi kullanılmıştır. Bu algoritmalar gürültüye duyarlıdır. Bir algoritmanın bir kez türev kullanması, algoritmanın gürültüye karşı bağımsızlık kazanmasını sağlamıştır. Diğer algoritma ise iletkenlik dağılımını geri çözmek için duyarlılık matrisi kullanmaktadır.

Anahtar Kelimeler: elektriksel empedans tomografisi, manyetik rezonans görüntüleme, iletkenlik geri çözüme, B tabanlı geriçatım algoritmaları.

ACKNOWLEDGEMENTS

Firstly, I would like to thank my family for having given me this great opportunity. This study has brought the biggest and most important changes to my life. I sincerely thank you for all your support, understanding, and patience.

I would also like to express my gratitude to my supervisor Prof. Dr. Murat Eyübođlu for his guidance throughout the study.

I would also like to thank all my lab mates: Emre, Evren, Rasim, Tankut, Ali, Ümit, Ceren for your long chats. Especially thanks to Emre, Evren, Rasim for your supports on my studies.

I would like to thank to TUBITAK-BIDEB for their supports for my thesis study.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	vi
ACKNOWLEDGEMENTS	viii
TABLE OF CONTENTS	ix
LIST OF TABLES.....	xii
LIST OF FIGURES	xiv
CHAPTER	
1. INTRODUCTION	1
1.1 A Summary of MREIT Studies	2
1.2 Objectives of the Thesis.....	4
1.3 Organization of the Thesis	5
2. THEORY	6
2.1 Introduction	6
2.2 The Forward Problem of MREIT	6
2.3 Extraction of Magnetic Flux Density from MR Images	11
2.4 The Inverse Problem of MREIT.....	13
2.5 Classification of the Reconstruction Algorithms	15
3. MAGNETIC FLUX DENSITY BASED RECONSTRUCTION ALGORITHMS	17
3.1 Reconstruction by Harmonic B_z algorithm	17
3.1.1 Introduction.....	17
3.1.2 Problem Definition.....	18
3.1.3 Implementation	20
3.2 Reconstruction by Variational Gradient B_z algorithm.....	24
3.2.1 Introduction.....	24

3.2.2	Problem Definition.....	24
3.2.3	Implementation	26
3.3	Reconstruction by Sensitivity Matrix algorithm	30
3.3.1	Introduction.....	30
3.3.2	Problem Definition.....	30
3.3.3	Implementation	32
3.4	Reconstruction by Algebraic reconstruction algorithm	35
3.4.1	Introduction.....	35
3.4.2	Problem Definition.....	35
3.4.3	Implementation	38
4.	SIMULATION AND COMPARISON	41
4.1	Introduction	41
4.2	Computer models.....	41
4.2.1	Computer Model-1	42
4.2.2	Computer Model-2	43
4.2.3	Computer Model-3	44
4.2.4	Computer Model-4	45
4.2.5	Thorax Model	46
4.3	Phantom used in experiments.....	47
4.4	Noise used for computer models	48
4.5	Error calculation and stopping criteria.....	50
4.6	Spatial Resolution.....	51
4.7	Simulation Results for Magnetic Flux Density Based Reconstruction Algorithms	52
4.7.1	Reconstruction by Harmonic B_z algorithm.....	52
4.7.1.1	Results for Model-1	53
4.7.1.2	Results for Model- 2	58
4.7.1.3	Results for Model- 3	61
4.7.1.4	Results for Model- 4	62
4.7.1.5	Results for Thorax Model	65

4.7.2	Reconstruction by Variational Gradient B_z Algorithm.	66
4.7.2.1	Results for Model- 1	66
4.7.2.2	Results for Model- 2	72
4.7.2.3	Results for Model- 3	75
4.7.2.4	Results for Model-4	76
4.7.2.5	Results for Thorax Model	79
4.7.3	Reconstruction by Sensitivity Matrix.....	81
4.7.3.1	Results for Model- 1	81
4.7.3.2	Results for Model-2	87
4.7.3.3	Results for Model-3	90
4.7.3.4	Results for Model-4	91
4.7.3.5	Results for Thorax Model	94
4.7.4	Reconstruction by Algebraic Reconstruction Algorithm ...	96
4.7.4.1	Results for Model- 1	96
4.7.4.2	Results for Model-2	101
4.7.4.3	Results for Model-3	103
4.7.4.4	Results for Model-4	105
4.7.4.5	Results for Thorax Model	108
4.8	Comparison of Four Magnetic Flux Density Based Algorithms...	109
4.8.1	Noise-free data.....	109
4.8.2	Noisy data	117
4.9	Experimental Results for Magnetic Flux Density Based Reconstruction Algorithms	121
4.10	Comparison of Simulation and Experimental Results.....	130
5.	CONCLUSIONS AND FUTURE WORK.....	134
5.1	Conclusions	134
5.2	Future work	138
	REFERENCES	139

LIST OF TABLES

TABLES

4.1	Conductivity values of computer model-1	42
4.2	Number of pixels in the gap, for different positions.....	44
4.3	Square conductivities for different cases and contrast levels.....	46
4.4	Conductivity values of Thorax Model	47
4.5	Experimental model's conductivity values	47
4.6	Material ratios for different conductivity values.	48
4.7	Reconstructed error values after iteration 7	54
4.8	Number of pixels in the gap, for different positions.....	58
4.9	Reconstructed error values for thorax model	66
4.10	Reconstructed error values of all regions, for iteration 1 and iteration 9 .	69
4.11	Number of pixels in the gap, for different positions.....	73
4.12	Reconstructed error values for thorax model.....	79
4.13	Reconstructed error values of all regions, noise free data	82
4.14	Reconstructed error values of all regions for noise free and noisy data ...	83
4.15	Mean reconstructed conductivity values of all regions for noise free and noisy data	85
4.16	Mean reconstructed conductivity values of all regions for two cases (for noise free data).....	86
4.17	Number of pixels in the gap, for different positions.....	87
4.18	Mean Conductivity values of squares.....	90
4.19	Reconstructed error values for thorax model	94
4.20	Reconstructed error values after 2 iterations.....	97
4.21	Number of pixels in the gap, for different positions.....	101
4.22	Reconstructed error values for thorax model.....	109

4.23	Reconstructed error values of all regions for noise free data.....	111
4.24	Number of pixels in the gap, for different positions.....	113
4.25	Reconstructed error values of all regions for noise free data.....	119
4.26	Error values of regions for Sensitivity Matrix algorithm.....	123
4.27	Error values of regions for Sensitivity Matrix algorithm.....	127
4.28	Error values of all regions for VGBz algorithm and Algebraic reconstruction algorithm	129
4.29	Experimental phantoms' conductivity values	130
5.1	The parameters used for performance chart.....	137

LIST OF FIGURES

FIGURES

2.1	The object with attached electrodes and MR triggered current source.....	7
2.2	Mesh elements used for FEM.....	9
2.3	The position of object in MRI scanner to extract B_z	12
3.1	Vertical and Horizontal current injection patterns	20
3.2	Flowchart for Harmonic B_z algorithm	23
3.3	Flowchart for VGB_z algorithm	29
3.4	Flowchart for Sensitivity Matrix reconstruction algorithm	34
3.5	Flowchart for Algebraic reconstruction algorithm	40
4.1	Computer model-1 with different conductivity regions.....	42
4.2	Current injection patterns	43
4.3	Computer model-2 with different conductivity regions.....	43
4.4	Computer model-3 with different conductivity regions.....	44
4.5	Computer model-4 with different conductivity regions.....	45
4.6	Thorax Model with different conductivity regions.....	46
4.7	Experimental phantom	48
4.8	FWHM on the PSF of the middle row of the imaging slice.....	52
4.9	Reconstruction by Harmonic B_z algorithm for Model-1 (noise free data).	54
4.10	Reconstructed conductivity errors by Harmonic B_z algorithm for Model-1 (noise free data)	55
4.11	Reconstruction by Harmonic B_z algorithm for Model-1 (all images for iteration 7)	57
4.12	Comparison of reconstruction errors for Harmonic B_z algorithm, for different SNR values and no noise data, for Model-1.	58
4.13	Reconstruction by Harmonic B_z algorithm for Model-2, position-4.....	59

4.14	PSF for middle row of imaging slice by Harmonic B_z algorithm for Model-2, position-1	60
4.15	FWHM values according to change of gap between two squares for all positions	60
4.16	Reconstruction by Harmonic B_z algorithm for Model-3(after iteration8)	61
4.17	Reconstruction by Harmonic B_z algorithm for Model-4,contrast level-2.	62
4.18	Reconstruction by Harmonic B_z algorithm for Model-4,contrast level-2.	63
4.19	Error plots of the square for 10 different contrast levels	63
4.20	Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels.....	64
4.21	Reconstruction by Harmonic B_z algorithm for Thorax Model.....	65
4.22	Reconstruction by VGB_z algorithm for Model-1 (noise free data)	67
4.23	Effect of electrode-edge ratio (noise free data)	68
4.24	Reconstructed conductivity errors by VGB_z algorithm for Model-1 (noise free data)	70
4.25	Reconstruction by VGB_z algorithm for Model-1	71
4.26	Comparison of reconstruction errors for VGB_z algorithm, for different SNR values and no noise data, for Model-1.....	72
4.27	Reconstruction by VGB_z algorithm for Model-2, position-4, for conductive case.....	73
4.28	Reconstruction by VGB_z algorithm for Model-2, position-4, for resistive case.....	74
4.29	PSF for middle row of imaging slice by VGB_z algorithm for Model-2, position-1.....	74
4.30	FWHM values according to change of gap between two squares for all positions	75
4.31	Reconstruction by VGB_z algorithm for Model-3, (after iteration 8).....	76
4.32	Reconstruction by VGB_z algorithm for Model-4, contrast level-2.....	77
4.33	Error plots of the square for 10 different contrast levels.	78
4.34	Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels	79
4.35	Reconstruction by VGB_z algorithm for Thorax Model.....	80
4.36	Reconstruction by Sensitivity Matrix for Model-1.....	82
4.37	Reconstruction by Sensitivity Matrix for Model-1.....	84

4.38	Reconstructed conductivity distributions for two cases (noise-free data)	86
4.39	Reconstruction by Sensitivity Matrix for Model-2, position-4	88
4.40	PSF for middle row of imaging slice by Sensitivity Matrix algorithm for Model-2, position-1	89
4.41	FWHM values according to change of gap between two squares for all positions	89
4.42	Reconstruction by Sensitivity Matrix for Model-3	91
4.43	Reconstruction by Sensitivity Matrix for Model-4, contrast level-2	92
4.44	Error plots of the square for 10 different contrast levels	93
4.45	Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels	94
4.46	Reconstruction by Sensitivity Matrix algorithm for Thorax Model	95
4.47	Reconstruction by Algebraic Reconstruction algorithm for Model-1 (noise free data)	97
4.48	Reconstructed conductivity errors by Algebraic Reconstruction algorithm for Model-1 (noise free data)	98
4.49	Comparison of reconstruction errors for Algebraic Reconstruction algorithm, for different SNR values and no noise data, for Model-1	99
4.50	Reconstruction by Algebraic Reconstruction algorithm for Model-1 (all images for iteration 2)	100
4.51	Reconstruction by Algebraic Reconstruction algorithm for Model-2, position-4	102
4.52	PSF for middle row of imaging slice by Algebraic Reconstruction algorithm for Model-2, position-1	102
4.53	FWHM values according to change of gap between two squares for all positions	103
4.54	Reconstruction by Algebraic Reconstruction algorithm for Model-3, (after iteration 2)	104
4.55	Reconstruction by Algebraic Reconstruction algorithm for Model-4, contrast level-2	105
4.56	Reconstruction by Algebraic Reconstruction algorithm for Model-4, contrast level-2	106
4.57	Error plots of the square for 10 different contrast levels	106
4.58	Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels	107

4.59	Reconstruction by Algebraic reconstruction algorithm, Thorax Model .	108
4.60	Reconstructed conductivity distributions for Model-1 (noise free data)	110
4.61	Comparison of three iterative algorithms according to the total error values and iteration numbers.....	112
4.62	Calculated FWHM according to the positions of two square elements..	114
4.63	Comparison of mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels, for all reconstruction algorithms.....	115
4.64	Reconstructed conductivity distributions for Thorax Model (noise-free data)	116
4.65	Reconstructed conductivity distributions for Model-1 (SNR 30)	118
4.66	Reconstructed conductivity distributions for Thorax Model (SNR 30) .	120
4.67	Experimental B_z data.....	122
4.68	Reconstructed conductivity distributions for experimental phantom.....	123
4.69	Reconstructed conductivity distributions for experimental phantom.....	125
4.70	Experimental B_z data.....	126
4.71	Reconstructed conductivity distributions for experimental phantom.....	127
4.72	Reconstructed conductivity distributions for experimental phantom.....	128
4.73	Reconstructed conductivity distributions for <i>Phantom-1</i>	131
4.74	Reconstructed conductivity distributions for <i>Phantom-2</i>	132
5.1	Performance chart for four B-based reconstruction algorithms.	137

CHAPTER 1

INTRODUCTION

The electrical conductivity imaging is one of the topics of researches on imaging. The biological tissues have different conductivity values according to their types. Their conductivity values change with the physiological and pathological states of the tissues. Some tissues have anisotropic conductivity. But in this thesis used conductivity values are isotropic.

Different methods are used to measure the electrical conductivity of tissues inside the body. Some methods are invasive methods. The tissue must be placed between two plate electrodes [1]. These methods are difficult methods in practice. For this reason a non-invasive measurement method is developed by Henderson and Webster in 1978 [2]. Electrical Impedance Tomography (EIT) is used to reconstruct cross-sectional conductivity images. In 1984, Barber and Brown [3] gave the information to use this method in medical areas. EIT is based on potential measurements [1]. Electrodes are attached on the surface of body and current is applied to these electrodes. Some methods use coil to induce a current distribution inside the body [4]. According to the applied current, a current distribution occurs inside the body. Peripheral voltage measurements are used. These measurements are taken from the surface electrodes, for different combinations of electrodes. The conductivity distribution of tissues inside the body is found with these different voltage measurements. But the measured data can be affected by the electrode positions and geometry of object [5]. Also EIT reconstructs conductivity image with relatively low spatial resolution and accuracy [1]. This is a disadvantage for EIT. To solve these problems and reconstruct good conductivity images Magnetic Resonance Imaging (MRI) is

used with EIT. With an MRI system, magnetic flux density distribution in the imaging region can be measured easily. For this purpose a new imaging modality to measure current densities by MRI is developed [6]. This technique is called Magnetic Resonance-Current Density Imaging (MR-CDI).

1.1 A Summary of MREIT Studies

A new imaging method which is called Magnetic Resonance Electrical Impedance Tomography (MREIT) is explained by Zhang in 1992 [7]. This new method uses conventional voltage measurements and magnetic flux densities to produce conductivity distribution of an object. It has a high sensitivity to the conductivity distribution and also it provides good spatial resolution. An external current is applied to the object by less number of electrodes than EIT. This externally applied current produces a current density distribution, \vec{J} , inside the object and generates a magnetic flux density distribution, \vec{B} , which can be measured by a MRI scanner. Conductivity distribution of an object can be easily calculated using \vec{B} , \vec{J} and measured boundary voltages.

Some method, used in MREIT, use \vec{J} to reconstruct conductivity distribution. To determine \vec{J} , the x, y and z component of magnetic flux density is used. For this purpose, the object must be rotated in MRI scanner. Ider *et al* [8] used integration along the Equipotential lines and Cartesian grid lines. In 2002 Kwon *et al* [9] used an algorithm which is called J-substitution algorithm. Özdemir *et al* [10] explained the Equipotential projection based MREIT and made same experiments. In 2009 Boyacıoğlu and Eyüboğlu [11] used a hybrid algorithm, which uses J-substitution and filtered Equipotential projection.

Other methods use only one component of \vec{B} to reconstruct a conductivity distribution. The component of \vec{B} is chosen with the same direction of main magnetic field of MRI system. It is always in z direction. By this way the need of object rotation inside a MRI scanner is eliminated. In 1998, Ider and Birgül [12] proposed a method which uses sensitivity matrix. A sensitivity matrix is calculated according to the change of each element's conductivity value. Birgül

et al [13] made experimental studies and they used sensitivity matrix relation between conductivity and z component of \vec{B} , B_z . In 2003, Seo *et al* [14] proposed a new reconstruction algorithm which is based on the measurement of B_z . In 2003, Oh *et al* [15] improved the technique explained by Seo [14]. The name of this technique is Harmonic B_z reconstruction algorithm. They used the relation between Laplacian of B_z and derivatives of conductivity. Their algorithm worked successfully for noise free data. But for noisy data the algorithm did not work successfully for low Signal-to-Noise Ratio (SNR) because of twice differentiation of noisy B_z data. For this reason, Park *et al* [16] developed a new algorithm called Variational Gradient B_z algorithm to improve the noise performance. They differentiated B_z data only once. By this way, they differentiated noise only once. The errors near the boundaries spread over the inner parts of object in Variational Gradient B_z algorithm. In order to deal with this problem, Kwon *et al* [17] proposed enhanced version of the previous algorithm. The boundary information was effectively used in this new algorithm. Also in 2004, Ider and Onart [18] proposed a new algorithm which uses B_z . In this algorithm non-linear matrix equations were used to reconstruct resistivity. At least two current injection patterns were necessary. In 2008, Jeon *et al* [19] proposed an algorithm which is called local Harmonic B_z algorithm. The algorithm deals with the defects caused by problematic regions of object. They used segmentation method to extract problematic regions. Huixian *et al* [20] explained an algorithm which uses only B_z . The conductivity distribution of object at oblique slice can be imaged with their algorithms.

All B-based reconstruction algorithms explained above use only one component of \vec{B} that eliminates the object rotation in MRI scanner.

1.2 Objectives of the Thesis

Several studies are made to measure the conductivity distribution of a region inside the body or object. MREIT is one of the best methods to take these measurements. Different research groups use different reconstruction algorithms to produce a conductivity image. Each research group use different methods, different models and different currents. Some groups use three components of \vec{B} to calculate \vec{J} . But the other groups use only one component of \vec{B} . By this way, they eliminate object rotation inside MRI scanner. These kinds of methods are called B-based reconstruction algorithms.

The objectives of this thesis are:

- To classify the magnetic flux density based reconstruction algorithms.
- To examine how magnetic flux density based reconstruction algorithms work.
- To implement some of the magnetic flux density based reconstruction algorithms.
- To compare the implemented algorithms for different conditions.
- To take experimental data for testing each algorithm.

In this thesis, four different B-based reconstruction algorithms are studied. First the Harmonic B_z algorithm [15], which uses the relation between Laplacian of B_z and derivative of conductivity distribution, is examined. The second algorithm is Variational Gradient B_z algorithm [16]. The difference of this algorithm from the previous algorithm is the one B_z differentiation. Then Sensitivity matrix algorithm [12] is examined. The calculated sensitivity matrix is used to reconstruct conductivity distribution for this algorithm. The last algorithm is Algebraic reconstruction algorithm [18]. In this algorithm, matrix equations were used to reconstruct resistivity.

1.3 Organization of the Thesis

In Chapter 2, the forward and inverse problems in MR-EIT are explained and formulated. The formulas which are used for extracting magnetic flux density from an image, taken from an MRI scanner, are explained. Also the methods used in reconstruction are classified. In Chapter 3, the magnetic flux density based reconstruction algorithms, which are used in this thesis, are explained and formulations for these reconstruction algorithms are derived. In Chapter 4, four computer models, which are used to test the performance of each algorithm, are explained. Also experimental phantom, used to take experimental data, is explained. The results taken from reconstruction algorithms and comparison for these algorithms are given. The experimental results are also given in this chapter. Finally in Chapter 5, a brief summary, conclusions and future work are given.

CHAPTER 2

THEORY

2.1 Introduction

In this chapter, the forward problem for MREIT is explained in the first section. The forward problem solutions are the important parts of the iterative algorithms. The reconstructed conductivity distributions are used as an input for forward problem during the iterations. For this reason the formulations for MREIT forward problem are shown. The Finite Element Method (FEM) is explained briefly. Since, the usage of elements in object is important to produce good results. Then the procedure to extract magnetic flux density distribution from an MRI data is explained. In the next section, the inverse problem and formulations used for inverse problem are explained. The conductivity distribution in the object is obtained by the inverse problem. Finally, the types of MREIT reconstruction algorithms are given.

2.2 The Forward Problem of MREIT

In MREIT the generated magnetic flux density is used to produce the conductivity distribution inside the object. Surface electrodes are used to inject current. According to the injected current, current density appears inside the object. But this current density can not be measured directly with MRI scanner. The magnetic flux density induced by the injected current is used since; magnetic flux density is the only parameter that can be measured by MRI scanner. The forward problem solutions are the important parts of the iterative algorithms. The

reconstructed conductivity distributions are used as an input for forward problem during the iterations.

The forward problem uses boundary conditions and known conductivity values to calculate the potential values. A current is injected from the surface electrodes which are placed on the boundary of the object. The object's inner conductivity values are isotropic which changes between zero and positive infinity. This injected current is triggered with an MRI system. According to the injected current, a current density is generated inside the object. The object, attached electrodes and MRI triggered current source shown in Figure 2.1.

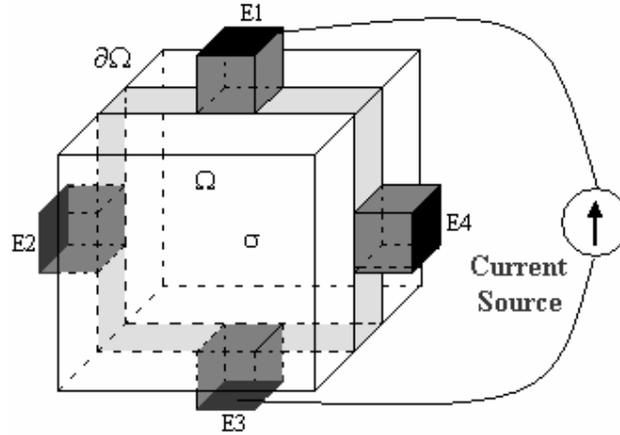


Figure 2.1: The object with attached electrodes and MR triggered current source.

As seen in Figure 2.1, the object has a volume, Ω , a boundary, $\partial\Omega$, an inner conductivity distribution, σ and four electrodes, E1, E2, E3, E4. The current apply interval is short for making the conductivity distribution time independent during the pulses.

Boundary Value Problem (BVP) is used to calculate potential distribution inside the object. The boundary conditions are important for solution of BVP. Neumann boundary condition, which is defined in equation (2.1), is used in forward problem.

$$\sigma \frac{\partial \phi}{\partial n} = \begin{cases} J & \text{on positive current electrode} \\ -J & \text{on negative current electrode} \\ 0 & \text{elsewhere} \end{cases} \quad (2.1)$$

where, n is the outward unit vector on $\partial\Omega$ and J is applied current.

The relation between conductivity, σ and potential distribution, ϕ is shown in equation (2.2).

$$\nabla \cdot (\sigma \nabla \phi) = 0 \quad \text{in } \Omega \quad (2.2)$$

where, Ω is the volume of the object. As seen from equation (2.2), we can calculate the potential distribution inside an object, if we know the inner conductivity distribution of the object.

We can find the potential distribution with different techniques like Finite Element Method (FEM), Finite Difference Method (FDM), *etc.* In this thesis, FEM is used to find the potential distribution inside the object for 2-dimensional case. Also FEM can be used for 3-dimensional case. The detailed explanations for 3-dimensional case can be found in Onart [21].

In FEM, the imaging slice is separated into small cells. By this way more points are used to find the potential distributions inside the object. These small cells are called as the mesh elements of the object. These mesh elements are shown in Figure 2.2.

If we increase the mesh element number of model, the solution of potential distribution will be more accurate. Since, the potential distribution is taken from more than one mesh element for a pixel. Each pixel uses more information in this case.

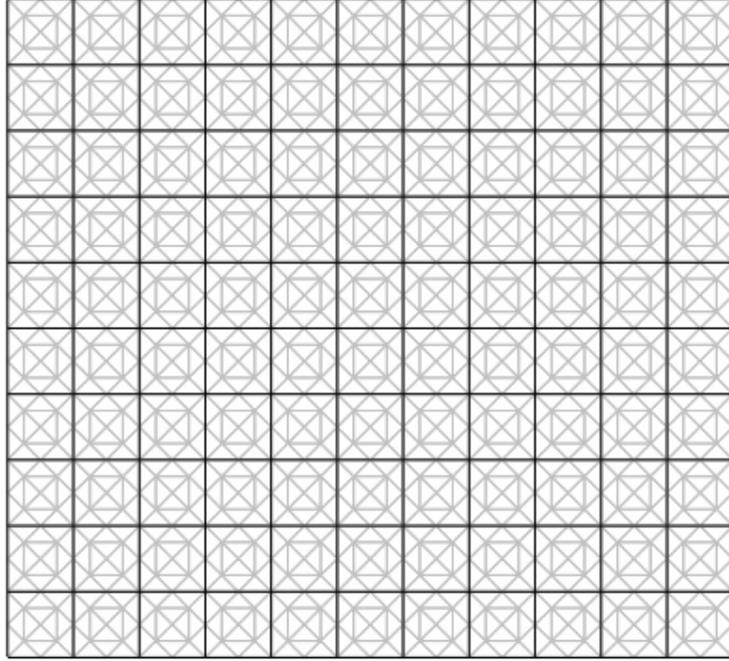


Figure 2.2: Mesh elements used for FEM.

But there is a disadvantage of increasing the mesh element number. If we increase the element number, we will need more computation time to solve the forward problem.

After finding the potential distribution, the equation (2.3) is used to find the electric field intensity.

$$\vec{E} = -\nabla\phi \quad (2.3)$$

The relation between electric field intensity and current density is shown below.

$$\vec{J} = \sigma\vec{E} = -\sigma\nabla\phi \quad (2.4)$$

As explained before the injected current generates magnetic flux density inside the object. This magnetic flux density is important for MREIT problems. Biot-

Savart law is used to generate magnetic flux density from calculated current density. This relation is shown in equation (2.5).

$$\vec{B}(r) = \frac{\mu_0}{4\pi} \int_{\Omega} \vec{J}(r') \times \frac{(r-r')}{|r-r'|^3} dv \quad (2.5)$$

where, μ_0 is the permeability of free space, r is the unit vector from origin to source point (x, y, z) , r' is the unit vector from origin to field point (x', y', z') .

As explained before, small mesh elements are used in FEM approach. Equation (2.5) gives the magnetic flux density on a point in space. In our studies, we need discrete calculation of magnetic flux density in the center of each mesh element of the model. For the i^{th} element the current density value is multiplied with the area of mesh element. Also at the center of each element there is a point current source, \vec{I} . The direction of \vec{I} is the same direction of current density produced in the mesh element. The generated magnetic field at any field point is calculated with equation (2.6).

$$\vec{B} = \frac{\mu_0}{4\pi} \sum_{i=1}^m \frac{\vec{I} \times \vec{R}}{R^3} \quad (2.6)$$

In equation (2.6), m is the number of mesh elements, \vec{R} is defined from source point (x, y, z) to field point (x', y', z') , R is the magnitude of \vec{R} . By this way, discretization can be made and \vec{B} can be calculated at the center of each mesh element. The effect of each element on itself is assumed negligible to prevent singularity problems.

Forward problem solutions are the important parts of the iterative algorithms. First an initial conductivity distribution is assumed. According to this distribution potential values are calculated for this uniform case. Then the conductivity distribution is calculated. This calculated conductivity distribution used as the

input for the forward problem. Then the forward problem is solved. To terminate iterations, the change in the conductivity distribution is examined at iterations.

2.3 Extraction of Magnetic Flux Density from MR Images

The magnetic flux density, which is generated by externally applied current, can not be measured directly with an MRI scanner. For this reason some computations are used to extract the magnetic flux density distribution from a MR image.

For magnetic flux density computation, the difference between phase for no current case and the phase for applied current case are used.

When we apply a current, a phase shift occurs. The magnetization for no current case is shown in equation (2.7).

$$\vec{M}_{nc}(x, y) = M(x, y) e^{(j\gamma B t + j\Phi_{nc})} \quad (2.7)$$

After applying an external current on the imaging object, a phase shift occurs. This phase shift is shown in equation (2.8). This phase shift is added to the equation for no current case. The new equation is shown in equation (2.9).

$$\Phi_J = \gamma B_z^J(x, y) T_C \quad (2.8)$$

$$\vec{M}_J(x, y) = M(x, y) e^{(j\gamma B t + j\gamma B_z^J(x, y) T_C + j\Phi_{nc})} \quad (2.9)$$

The ratio of equation (2.9) and (2.7) gives the following equation.

$$\frac{\vec{M}_J(x, y)}{\vec{M}_{nc}(x, y)} = e^{(j\gamma B_z^J(x, y) T_C)} = e^{j\theta(x, y)} \quad (2.10)$$

where, γ is the gyromagnetic ratio, T_C current applied time per excitation.

We can extract magnetic flux density as:

$$B_z^J(x, y) = \frac{\theta(x, y)}{\gamma T_C} \quad (2.11)$$

This extracted B_z is a component of magnetic flux density, which is parallel to the main magnetic field of MRI scanner. The illustration for the position of object to extract B_z is shown in Figure 2.3.

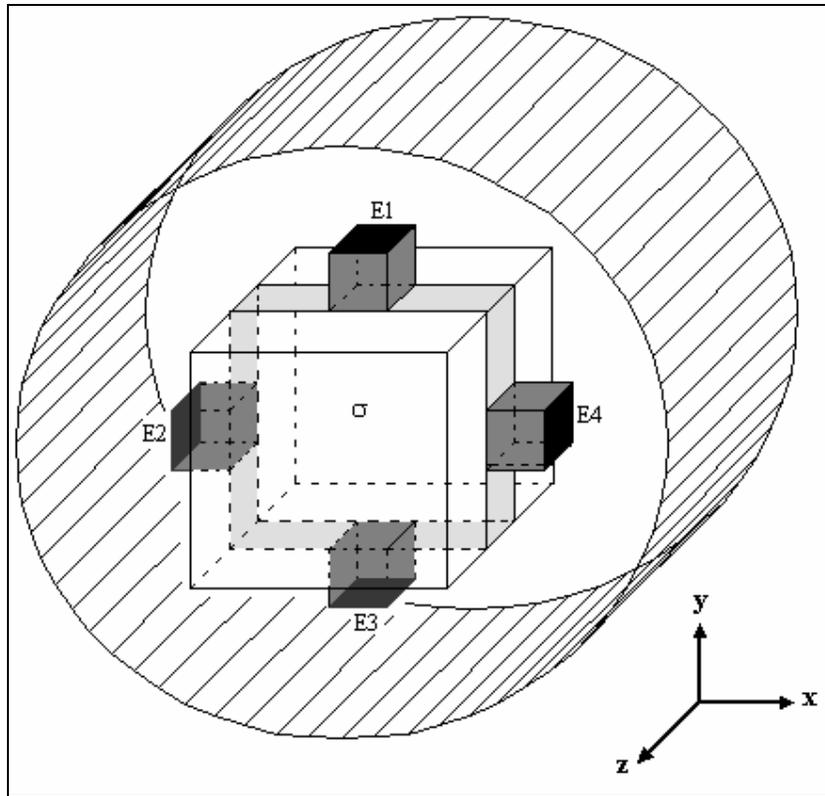


Figure 2.3: The position of object in MRI scanner to extract B_z .

To extract the other components of \vec{B} the object must be rotated inside the MRI scanner. And the explained procedures must be used for each position. In this thesis, only z component of \vec{B} is used. Therefore there is no need to rotate the object for the reconstruction algorithms used in this thesis.

2.4 The Inverse Problem of MREIT

As explained in the previous section, according to the externally applied current a magnetic flux density is generated inside the object. In MREIT, the main aim is to find the unknown conductivity distribution inside the object. The generated magnetic flux density is used to calculate the conductivity distribution of an object. Also some voltage measurement taken on the boundary of the object can be used during the reconstruction processes. The forward problem solutions are used as the input for inverse problem.

Reconstruction algorithms which are used in this thesis use only z-component of magnetic flux density. This component is parallel to the main magnetic field of MRI scanner. For each reconstruction algorithms, B_z data must be calculated first.

As explained by Scott [6], for inverse problem solutions it is assumed that the current flows in a region constant conductivity, σ , and permeability, μ . This satisfies the following equations.

$$\vec{J} = \sigma \vec{E} \quad (2.12)$$

where \vec{E} is the electric field intensity.

$$\vec{B} = \mu \vec{H} \quad (2.13)$$

$$\nabla \times \vec{E} = 0 \quad (2.14)$$

$$\nabla \cdot \vec{B} = 0 \quad (2.15)$$

$$\vec{E} = -\nabla \Phi \quad (2.16)$$

The magnetic field intensity, \vec{H} , of a vector can be written as:

$$\nabla \times \nabla \times \vec{H} = \nabla(\nabla \cdot \vec{H}) - \nabla^2 \vec{H} \quad (2.17)$$

The equations shown above are used to produce equation (2.18) from equation (2.17).

$$\nabla^2 \vec{H} = \vec{J} \times \frac{\nabla \sigma}{\sigma} \quad (2.18)$$

This equation shows the relation between magnetic flux intensity, current density and conductivity. By using equation (2.12), (2.13), (2.16) and (2.18), the relation between magnetic flux density, conductivity and potential distribution can be found as:

$$\frac{\nabla^2 \vec{B}}{\mu_0} = -\nabla \phi \times \nabla \sigma \quad (2.19)$$

where, $\nabla^2 \vec{B}$ is the Laplacian of \vec{B} , ϕ is the potential distribution.

This relation is used in magnetic flux density based reconstruction algorithms. Using Laplacian \vec{B} means, second order derivatives of \vec{B} are utilized. But laplacian operator has a disadvantage for noisy cases. It differentiates noise twice.

Some of the reconstruction algorithms uses the relation between \vec{B} , \vec{J} and natural logarithm of resistivity, R [8]. They assumed equation (2.14) is satisfied. The relation is shown as:

$$\nabla R \times \vec{J} = -\nabla \times \vec{J} \quad (2.20)$$

R is defined as $\rho = e^R$. Using the fact that $\vec{J} = \nabla \times \vec{B} / \mu_0$, equation (2.20) can be written as:

$$\nabla R \times \vec{J} = -\nabla \left(\frac{\nabla \cdot \vec{B}}{\mu_0} \right) + \frac{\nabla^2 \vec{B}}{\mu_0} \quad (2.21)$$

The divergence of \vec{B} is equal to zero. So we can write equation (2.21) as:

$$\nabla R \times \vec{J} = \frac{\nabla^2 \vec{B}}{\mu_0} \quad (2.22)$$

Equation (2.22) gives the relation between resistivity, magnetic flux density and current density.

For reconstruction algorithms used in this study, only the z-component of magnetic flux density is used. For this reason, equation (2.19) and (2.22) are used with only z components.

The details of reconstruction algorithms' inverse problem procedures are explained in chapter 3.

2.5 Classification of the Reconstruction Algorithms

There are several methods which are used to reconstruct resistivity or conductivity distribution. Each reconstruction algorithm use different parameters and conditions. Some methods are iterative.

The reconstruction algorithms are divided into two main groups. The first group of algorithms uses current density distributions to reconstruct conductivity. This methods use three components of measured magnetic flux density to produce current density. For this reason, these kind of reconstruction algorithms are called Current Density Based (J-based) reconstruction algorithms or Type-I reconstruction algorithms.

The other groups of algorithms use only one component of magnetic flux density which is parallel to the main magnetic field. The subject rotation is not needed

because of the use of only one component of magnetic flux density. For this reasons the reconstruction algorithms are called as Magnetic Flux Density Based (B-based) reconstruction algorithms or Type-II reconstruction algorithms.

Four of the B-based reconstruction algorithms are examined in this thesis.

CHAPTER 3

MAGNETIC FLUX DENSITY BASED RECONSTRUCTION ALGORITHMS

Some of the magnetic flux density based MR-EIT reconstruction algorithms use only one component of magnetic field, which is parallel to the main magnetic field direction. In this chapter four magnetic density based reconstruction algorithms are explained.

3.1 Reconstruction by Harmonic B_z algorithm

3.1.1 Introduction

In this section, one of the magnetic flux density based reconstruction algorithms which is called Harmonic B_z reconstruction algorithm is explained. The Harmonic B_z reconstruction algorithm is a reconstruction algorithm which uses only one component of magnetic flux density. In this thesis z-direction is used as the direction of the main magnetic field.

An applied external current produces a magnetic flux density $\vec{B}^j = (B_x^j, B_y^j, B_z^j)$ inside an object. In 2003 Seo *et al* [14] proposed a new reconstruction algorithm which is based on the measurement of only one component of magnetic flux density, B_z . By this method, they eliminate subject rotation inside a Magnetic

Resonance Imaging (MRI) scanner. This reconstruction algorithm uses Laplacian of B_z , $\nabla^2 B_z$.

In 2003, Oh *et al* [15] improved the technique explained in the previous part. They use Laplacian of B_z . This is a disadvantage for this algorithm. Because, using $\nabla^2 B_z$ means making twice differentiation of noisy B_z . This algorithm calculates the derivatives of conductivity, $\nabla\sigma$, to produce a conductivity distribution.

3.1.2 Problem Definition

An object Ω is placed inside an MRI scanner to measure the z-component of magnetic flux density, [15]. P different surface electrodes are used to produce $N = P(P-1)/2$ independent electrode pairs to apply current. Let the injection current between the j^{th} pair of electrodes be I^j for $j=1, \dots, N$. This injected current produces a current density, $\vec{J}^j = (J_x^j, J_y^j, J_z^j)$, inside the object. Applied external current and produced current density, generates a magnetic flux density $\vec{B}^j = (B_x^j, B_y^j, B_z^j)$ inside the object. As a result, equation $\nabla \times \vec{B}^j = \mu_0 \vec{J}^j$ is used for conductive object. The magnetic flux density increases the MRI phase image. This increment is proportional to the produced magnetic flux density component which is parallel to the main magnetic field of the MRI scanner.

In Harmonic B_z reconstruction algorithm, the object conductivities are assumed to be isotropic.

A potential, ϕ^j , is produced according to the injected current (I^j , $j=1, \dots, N$). This potential is a solution of the following Neumann Boundary Problem (NBP).

$$\begin{cases} \nabla \cdot (\sigma \nabla \phi^j) = 0 & \text{in } \Omega \\ -\sigma \nabla \phi^j \cdot \hat{n} = g^j & \text{on } \partial\Omega \end{cases} \quad (3.1)$$

where, \hat{n} is the outward unit normal vector, g^j is the normal component of current density which is produced with the injected current on the boundary ($\partial\Omega$). This normal component is also defined with equation (3.2).

$$\int_e g^j ds = \pm I^j \quad (3.2)$$

where, e is the current injection electrode. The sign of current depends on the direction of current, and g^j is zero on the regions of the boundary which are not in contact with e .

If the conductivity distribution (σ), externally applied current (I^j) and electrode configuration on the object are known, the ϕ^j in equation (3.1) will be calculated by using the Finite Element Method (FEM).

Equation (3.3) is defined by Scott [6]. This equation can be used for Harmonic B_z algorithm.

$$\nabla^2 \vec{B} = -\mu_0 \nabla \phi \times \nabla \sigma \quad (3.3)$$

The z component of this equation can be written as:

$$\frac{1}{\mu_0} \nabla^2 B_z^j = \left(\frac{\partial \sigma}{\partial x}, \frac{\partial \sigma}{\partial y} \right) \cdot \left(\frac{\partial \phi^j}{\partial y}, -\frac{\partial \phi^j}{\partial x} \right) \quad j = 1, \dots, N \quad (3.4)$$

As seen from equation (3.4), the x and y gradients of ϕ^j and Laplacian of B_z is used to find the x and y derivatives of σ .

Matrix form of equation (3.4) is shown as:

$$\mathbf{Us} = \mathbf{b} \quad (3.5)$$

where,

$$\mathbf{U} = \begin{bmatrix} \frac{\partial \phi^1}{\partial y} & -\frac{\partial \phi^1}{\partial x} \\ \vdots & \vdots \\ \frac{\partial \phi^N}{\partial y} & -\frac{\partial \phi^N}{\partial x} \end{bmatrix} \quad \mathbf{s} = \begin{bmatrix} \frac{\partial \sigma}{\partial x} \\ \frac{\partial \sigma}{\partial y} \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} \nabla^2 B_z^1 \\ \vdots \\ \nabla^2 B_z^N \end{bmatrix} \quad (3.6)$$

3.1.3 Implementation

In this thesis, two current injection patterns ($N=2$), which are shown in Figure 3.1, are used. One of them is vertical, the other is horizontal. This means that these current injection patterns are orthogonal to each other. Each injection current is equal to 20 mA.

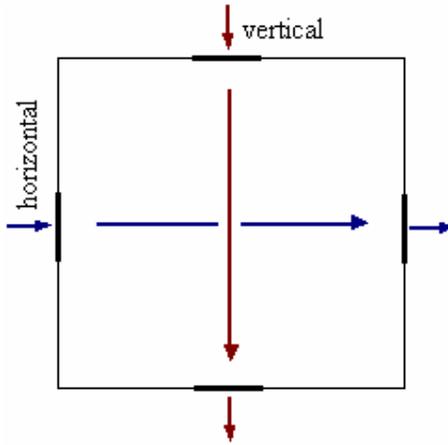


Figure 3.1: Vertical and Horizontal current injection patterns.

First, the Laplacian of B_z ($\nabla^2 B_z$) is calculated according to the change of B_z along the x, y and z direction. Central difference, which is shown in equation (3.7), is used for calculation of $\nabla^2 B_z$.

$$f''(x_0) = \frac{f(x_0 + h) - 2f(x_0) + f(x_0 - h)}{h^2} \quad (3.7)$$

For computation of $\nabla^2 B_z$ at the i^{th} pixel, equation (3.8) is used. This equation is also called three-point difference.

$$\begin{aligned} \nabla^2 B_z(x_i, y_i, z_i) = & \frac{B_z(x_{i+1}, y_i, z_i) - 2B_z(x_i, y_i, z_i) + B_z(x_{i-1}, y_i, z_i)}{(\Delta x)^2} + \\ & + \frac{B_z(x_i, y_{i+1}, z_i) - 2B_z(x_i, y_i, z_i) + B_z(x_i, y_{i-1}, z_i)}{(\Delta y)^2} + \\ & + \frac{B_z(x_i, y_i, z_{i+1}) - 2B_z(x_i, y_i, z_i) + B_z(x_i, y_i, z_{i-1})}{(\Delta z)^2} \end{aligned} \quad (3.8)$$

In equation (3.8), Δx , Δy and Δz are the dimensions of each pixel in the x, y and z directions.

The ϕ^j values are calculated for a uniform conductivity distribution. Then the x-derivative and y-derivative of ϕ^j are calculated using the forward difference formulation, which is shown in equation (3.9).

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} \quad (3.9)$$

For each current injection pattern, the Laplacian of B_z and x-gradient, y-gradient of potential are calculated. These values are calculated for each pixel in the imaging slice. Then using equation (3.5), the x-derivative and y-derivative of conductivity distribution can be calculated.

After calculation of conductivity derivatives, the line integral is used to produce the conductivity distribution. For this case, we assume that a pixel's conductivity value is known. This is the initial pixel. The reconstructed conductivity distribution can be calculated by using the initial conductivity value and calculated derivatives of σ . The potential is measured from a point on the edge

of imaging slice. This measured potential value is used for scaling. Equation (3.10) is used to terminate the iterative Harmonic B_z algorithm.

$$\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \varepsilon \quad (3.10)$$

where, ε is the defined tolerance value. As seen from equation (3.10) the reconstructed conductivity distributions for m^{th} iteration and $(m+1)^{\text{th}}$ iteration are used to terminate the iterations.

After finding the conductivity distribution, current density image can be found by using equation (3.11).

$$\vec{J}^j = -\sigma \nabla \phi^j \quad (3.11)$$

The Harmonic B_z iterative algorithm has the following steps:

Step-1: For $m=0$, assume an initial conductivity distribution, σ_0 .

Step-2: ϕ_{m+1}^j is calculated by solving the following NBP for each current injection pattern ($j=1, \dots, N$).

$$\begin{cases} \nabla \cdot (\sigma_m \nabla \phi_{m+1}^j) = 0 & \text{in } \Omega \\ -\sigma_m \nabla \phi_{m+1}^j \cdot \hat{n} = g^j & \text{on } \partial\Omega. \end{cases} \quad (3.12)$$

Step-3: Calculate the σ_{m+1} by using the equation (3.5). Then scale this conductivity distribution using the voltage measurements on the edge of the object.

Step-4: If $\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \varepsilon$ for a predefined tolerance value, current density image can be calculated or iterations can be terminated. Otherwise, set $m = (m+1)$ and go back to Step 2.

The flowchart for iterative Harmonic B_z algorithm is shown below.

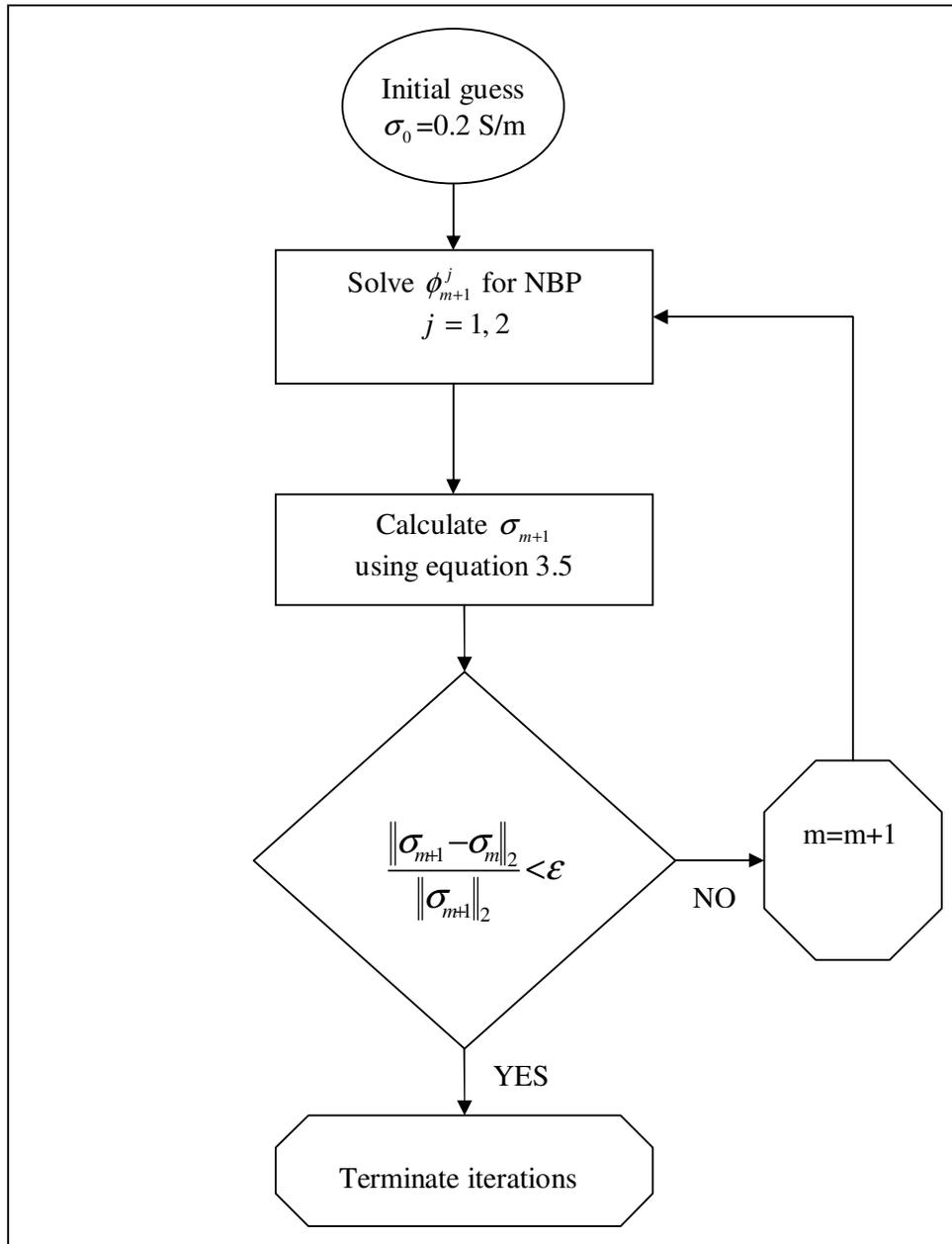


Figure 3.2: Flowchart for Harmonic B_z algorithm.

3.2 Reconstruction by Variational Gradient B_z algorithm

3.2.1 Introduction

In this section, one of the magnetic flux density based reconstruction algorithms which is called Variational Gradient B_z (VGB $_z$) reconstruction algorithm is explained. The Harmonic B_z reconstruction algorithm is a reconstruction algorithm which uses only one component of magnetic flux density. In this thesis, z-direction is used as the direction of the main magnetic field.

In 2003, Seo *et al* [14] and Oh *et al* [15] proposed a reconstruction algorithm which is based on the measurement of only one component of magnetic flux density, B_z . It is called as Harmonic B_z reconstruction algorithm. This algorithm is based on the calculation of $\nabla^2 B_z$. But this algorithm is very sensitive to noise because of the twice differentiation of noisy data.

In 2004, Park *et al* [16] proposed a new reconstruction algorithm which uses only one differentiation of B_z . The name of this algorithm is Variational Gradient B_z reconstruction algorithm.

3.2.2 Problem Definition

The conductivity distribution of the target object is isotropic for VGB $_z$ reconstruction algorithm.

As explained in Section 3.1.2, an applied external current generates a magnetic flux density, $\vec{B}^j = (B_x^j, B_y^j, B_z^j)$, inside the object. The z-component of this generated magnetic flux density is important for VGB $_z$ reconstruction algorithm.

A potential ϕ^j is produced according to the injected current (I^j , $j=1, \dots, N$). This potential is a solution of the NBP which is shown in equation (3.13).

$$\begin{cases} \nabla \cdot (\sigma \nabla \phi^j) = 0 & \text{in } \Omega \\ -\sigma \nabla \phi^j \cdot \hat{n} = g^j & \text{on } \partial\Omega, \end{cases} \quad (3.13)$$

where, \hat{n} is the outward unit normal vector, g^j is the normal component of current density, produced with the injected current, on the boundary ($\partial\Omega$).

There is a relation between magnetic flux density and current density distribution. This relation is shown in equation (3.14).

$$\nabla \times \vec{B} = \mu_0 \vec{J} \quad \text{and} \quad \vec{J} = -\sigma \nabla \phi \quad (3.14)$$

This equation can be written in integral form.

$$\int_{\Omega} \nabla \times \vec{B} \cdot \vec{\Psi} \, d\mathbf{r} = -\mu_0 \int_{\Omega} \sigma \nabla \phi \cdot \vec{\Psi} \, d\mathbf{r} \quad \text{for all } \vec{\Psi} \in [L^2(\Omega)]^3 \quad (3.15)$$

where, $L^2(\Omega)$ is the set of square integrable function in domain Ω . The x component, y component and z component of \vec{B} is placed inside this equation. But we need only z component of \vec{B} . For this reason we must eliminate B_x and B_y with suitable constraints on $\vec{\Psi}$.

For this purpose, the $\vec{\Psi}$ is chosen as $\vec{\Psi} = (\frac{\partial \psi}{\partial y}, -\frac{\partial \psi}{\partial x}, 0)$ with $\vec{\Psi} \in T$. The definition of T is shown below.

$$T := \left\{ \psi \in H^1(\Omega) : \text{for } \mathbf{r} \in \partial\Omega, \begin{cases} (i) \frac{\partial \psi}{\partial y}(\mathbf{r}) = 0 = \frac{\partial \psi}{\partial x}(\mathbf{r}) \text{ if } \mathbf{v} \cdot \vec{e}_3 \neq 0 \\ (ii) \frac{\partial \psi}{\partial z}(\mathbf{r}) = 0 \text{ if } \mathbf{v}(\mathbf{r}) \times \vec{e}_3 \neq 0 \end{cases} \right\} \quad (3.16)$$

where, $\vec{e}_3 = (0,0,1)$ and $H^1(\Omega)$ is the square integrable functions in Ω .

With these constraints and equation (3.15), the following equation which has only z component of \vec{B} , can be found.

$$\int_{\Omega} \sigma L \tilde{\nabla} \phi \cdot \tilde{\nabla} \psi \, d\mathbf{r} = \frac{1}{\mu_0} \int_{\Omega} \nabla B_z \cdot \nabla \psi \, d\mathbf{r} \quad \forall \psi \in T \quad (3.17)$$

where $\tilde{\nabla} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ and $L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$

For each pixel, equation (3.17) can be written as:

$$\sigma \left(\frac{\partial \phi}{\partial y} - \frac{\partial \phi}{\partial x} \right) = \frac{1}{\mu_0} \left(\frac{\partial B_z}{\partial y} + \frac{\partial B_z}{\partial x} \right) \quad (3.18)$$

For each current injection pattern, equation (3.18) is used to find the conductivity distribution of imaging slice.

3.2.3 Implementation

In this thesis, two current injection patterns ($N=2$) are used. One of them is vertical, the other is horizontal. This means these currents are orthogonal to each other. Each injection current is equal to 20 mA.

First the sum of x-gradient and y-gradient of B_z , $\left(\frac{\partial B_z}{\partial y} + \frac{\partial B_z}{\partial x} \right)$ is calculated according to the change of B_z along the x-direction and y-direction. Forward difference, which is shown in equation (3.19), is used for the calculation of $\left(\frac{\partial B_z}{\partial y} + \frac{\partial B_z}{\partial x} \right)$.

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} \quad (3.19)$$

The ϕ^j values are calculated for a uniform conductivity distribution. Then the x-derivative and y-derivative of ϕ^j are calculated using the forward difference formulation, which is shown in equation (3.19).

For each current injection pattern, the x and y gradients of B_z and potential are calculated. These values are calculated for each pixel in the imaging slice. Then using equation (3.18), the conductivity distribution can be calculated.

The potential is measured from a point on the edge of imaging slice. This measured potential value is used for scaling. To terminate the iterative VGB_z algorithm, equation (3.20) is used.

$$\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \varepsilon \quad (3.20)$$

where ε is the defined tolerance value. As seen from equation 3.20 the reconstructed conductivity distributions for m^{th} iteration and $(m+1)^{\text{th}}$ iteration are used to terminate the iterations.

The VGB_z iterative algorithm has the following steps:

Step-1: For $m=0$, assume an initial conductivity distribution σ_0 .

Step-2: ϕ_{m+1}^j is calculated by solving the following NBP for each current injection pattern ($j=1, \dots, N$).

$$\begin{cases} \nabla \cdot (\sigma_m \nabla \phi_{m+1}^j) = 0 & \text{in } \Omega \\ -\sigma_m \nabla \phi_{m+1}^j \cdot \hat{n} = g^j & \text{on } \partial\Omega \\ \int_{\partial\Omega} \phi_m^j = 0 \end{cases} \quad (3.21)$$

Step-3: Calculate the σ_{m+1} by using the equation (3.22).

$$\int_{\Omega} \sigma_{m+1} \begin{bmatrix} (L \tilde{\nabla} u^1) \cdot \tilde{\nabla} \psi \\ (L \tilde{\nabla} u^2) \cdot \tilde{\nabla} \psi \end{bmatrix} d\mathbf{r} = \begin{bmatrix} \frac{1}{\mu_0} \int_{\Omega} \nabla B_z^1 \cdot \nabla \psi d\mathbf{r} \\ \frac{1}{\mu_0} \int_{\Omega} \nabla B_z^2 \cdot \nabla \psi d\mathbf{r} \end{bmatrix} \quad \forall \psi \in T \quad (3.22)$$

Then, scale this conductivity distribution using the voltage measurements on the edge of the subject.

Step-4: If $\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \varepsilon$ for a predefined tolerance value, terminate

the iterations. Otherwise, set $m = (m+1)$ and go back to Step 2.

The flowchart for iterative VGB_z is shown in Figure 3.3.

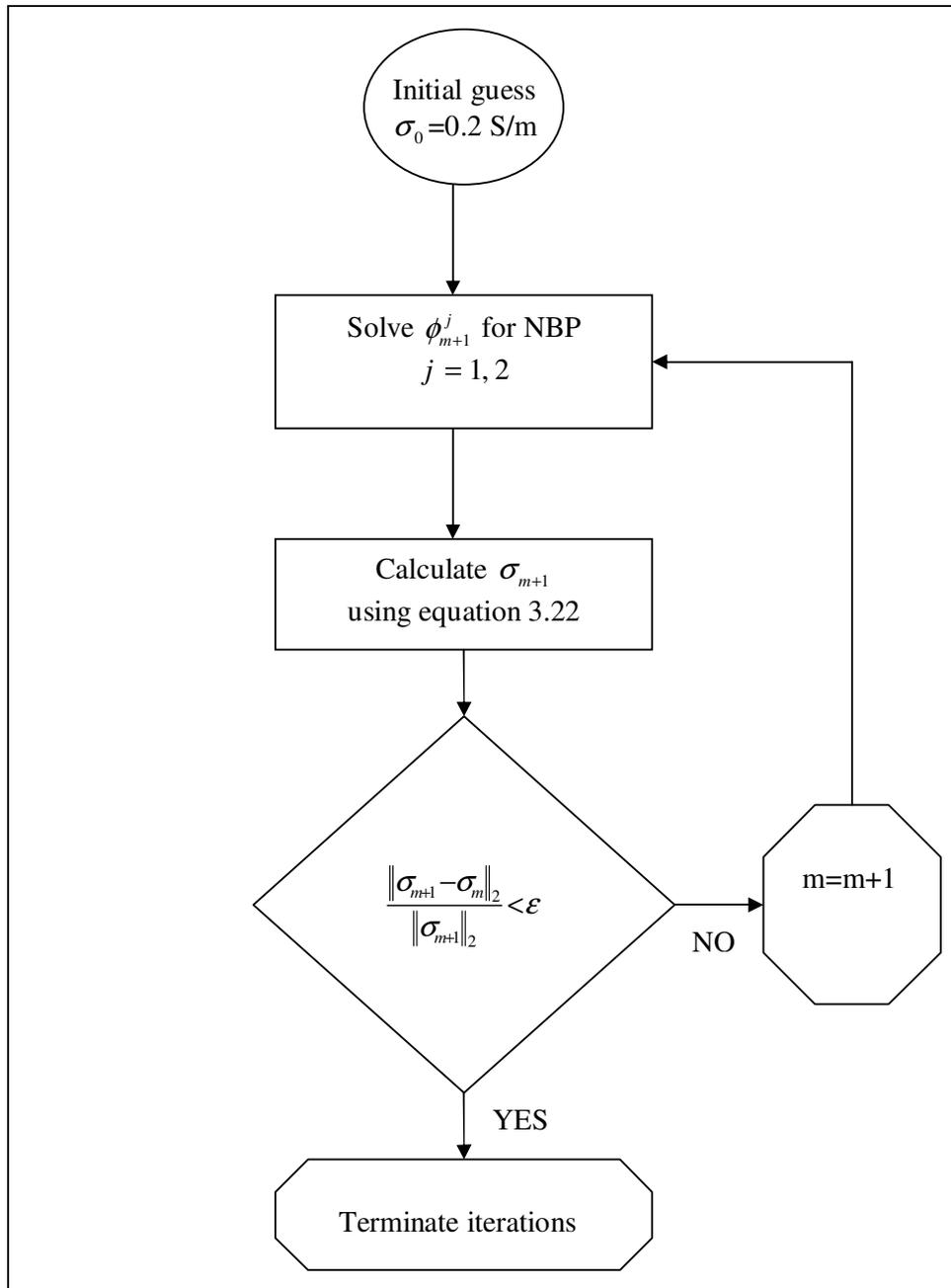


Figure 3.3: Flowchart for VGB_z algorithm.

3.3 Reconstruction by Sensitivity Matrix algorithm

3.3.1 Introduction

In this section, one of the magnetic flux density based reconstruction algorithms which is called Sensitivity Matrix reconstruction algorithm is explained. The Sensitivity Matrix reconstruction algorithm is a reconstruction algorithm which uses only one component of magnetic flux density.

In 1998, Ider and Birgül [12] use Sensitivity Matrix for reconstructing the conductivity distribution. Sensitivity matrix is related to the change of magnetic flux density according to the change of conductivity values of each element in the object. This algorithm uses perturbation in magnetic field measurement and Sensitivity matrix to find the perturbation in the conductivity values of each element.

3.3.2 Problem Definition

The conductivity values used in this algorithm is isotropic. As explained in Section 3.1.2, an applied external current generates a magnetic flux density, $\vec{B}^j = (B_x^j, B_y^j, B_z^j)$, inside the object. The z-component of this generated magnetic field is important for Sensitivity Matrix reconstruction algorithm.

This algorithm is based on producing a Sensitivity matrix. If we know the change characteristic of B_z for an initial conductivity distribution and the perturbation of conductivity, we will easily find the B_z data for the real conductivity distribution. This is shown in equation (3.23).

$$B_z(\sigma) \approx B_z(\sigma_0) + \frac{\partial B_z}{\partial \sigma} \cdot \Delta\sigma \quad (3.23)$$

where, $\Delta\sigma$ is equal to the difference between real conductivity distribution (σ) and initial conductivity distribution (σ_0).

Equation 3.23 can be written in a matrix form which is shown below.

$$\Delta \mathbf{b} = \mathbf{S} \Delta \boldsymbol{\sigma} \quad (3.24)$$

where, $\Delta \mathbf{b}$ is a vector which includes the perturbations in magnetic field measurements, $\Delta \boldsymbol{\sigma}$ is a vector which includes perturbation in the conductivity values of object's elements and \mathbf{S} is a matrix which is called sensitivity matrix. These are shown below.

$$\Delta \mathbf{b} = [\Delta b_1 \quad \Delta b_2 \quad \dots \quad \Delta b_m]^T \quad (3.25)$$

$$\Delta \boldsymbol{\sigma} = [\Delta \sigma_1 \quad \Delta \sigma_2 \quad \dots \quad \Delta \sigma_n]^T \quad (3.26)$$

$$\mathbf{S} = \left(\begin{array}{ccc} \frac{\partial b_1}{\partial \sigma_1} & \dots & \frac{\partial b_1}{\partial \sigma_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial b_m}{\partial \sigma_1} & \dots & \frac{\partial b_m}{\partial \sigma_n} \end{array} \right)_{\sigma=\sigma_0} \quad (3.27)$$

where, m is the number of measurements, n is the number of elements of object.

By producing a sensitivity matrix, we have a characteristic of change in B_z according to the change of conductivity values. To form a sensitivity matrix, all elements are changed with small values. The difference between B_z and $B_z(\sigma_0)$ is divided by the change of conductivity value. If the measurement number and elements number are high, the computation time for sensitivity matrix will be high. The sensitivity matrix is calculated for only the initial conductivity distribution. Then this matrix is used for different cases.

Singular Value Decomposition (SVD) is used to test the performance of the calculated sensitivity matrix. For this reason, we must examine the singular values of the sensitivity matrix. The \mathbf{S} matrix can be written as:

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T \quad (3.28)$$

where, \mathbf{U} is a $m \times m$ orthonormal matrix, \mathbf{V} is a $n \times n$ orthonormal matrix, $\mathbf{\Lambda}$ is a $m \times n$ diagonal matrix. Diagonal elements are the singular values of \mathbf{S} . For testing the performance of this matrix we must calculate the condition number which is equal to the ratio of the maximum singular value and minimum singular value. If the condition number very near to 1, the matrix said well-conditioned. If the condition number is greater than 1, the matrix said to be ill-conditioned.

After calculating the sensitivity matrix, the pseudo inverse of \mathbf{S} is calculated. Then this pseudo inverse and the calculated perturbation in magnetic field measurements are multiplied to find the perturbation in conductivity. This is shown in equation (3.29).

$$\Delta\sigma = \mathbf{S}^\dagger \Delta\mathbf{b} \quad (3.29)$$

where, \mathbf{S}^\dagger is the pseudo inverse of sensitivity matrix.

After finding the conductivity perturbation, the reconstructed conductivity distribution can be calculated.

3.3.3 Implementation

In this study, two orthogonal current injection patterns ($N=2$) are used to inject 20 mA current. The number of measurements is equal to 40. The number of elements of object is equal to 40. So, we must produce a 1600×1600 Sensitivity matrix. Initial conductivity distribution is assumed $\sigma_0 = 0.2$ S/m. For this distribution each element's conductivity is changed 0.02 S/m. For all changing cases, the difference between B_z and $B_z(\sigma_0)$ is divided by the change of conductivity value. The computed sensitivity matrix for one current injection pattern is ill-conditioned. For this reason, the sensitivity matrices can be combined.

$$\mathbf{S}_{\text{combined}} = [\mathbf{S}_1 \quad \mathbf{S}_2]^T \quad (3.30)$$

After computing the combined sensitivity matrix, singular values of matrix $\mathbf{S}_{\text{combined}}$ are examined. The singular values, which are very close to zero, are truncated.

After computing a good sensitivity matrix the equation (3.31) is used to find $\Delta\sigma$.

$$\Delta\sigma = (\mathbf{S}_{\text{combined}})^\dagger \Delta\mathbf{b}_{\text{combined}} \quad (3.31)$$

where, $\Delta\mathbf{b}_{\text{combined}}$ is the combined version of perturbation in B_z for two current injection pattern.

The reconstructed conductivity distribution is calculated using equation (3.32).

$$\sigma = \sigma_{\text{initial}} + \Delta\sigma \quad (3.32)$$

The Sensitivity Matrix reconstruction algorithm has the following steps:

Step-1: Assume an initial conductivity distribution, σ_0 .

Step-2: Compute the combined sensitivity matrix, $\mathbf{S}_{\text{combined}}$, according to the changes on the initial conductivity distribution for all current injection patterns. If necessary use truncation.

Step-3: Find pseudo inverse of \mathbf{S} .

Step-4: Measure B_z for all current injection pattern and calculate the perturbation in the magnetic field measurements, $\Delta\mathbf{b}$.

Step-5: Calculate perturbation in element conductivity values with equation $\Delta\sigma = (\mathbf{S}_{\text{combined}})^\dagger \Delta\mathbf{b}_{\text{combined}}$

Step-6: Calculate conductivity distribution.

The flowchart for Sensitivity Matrix algorithm is shown in Figure 3.4.

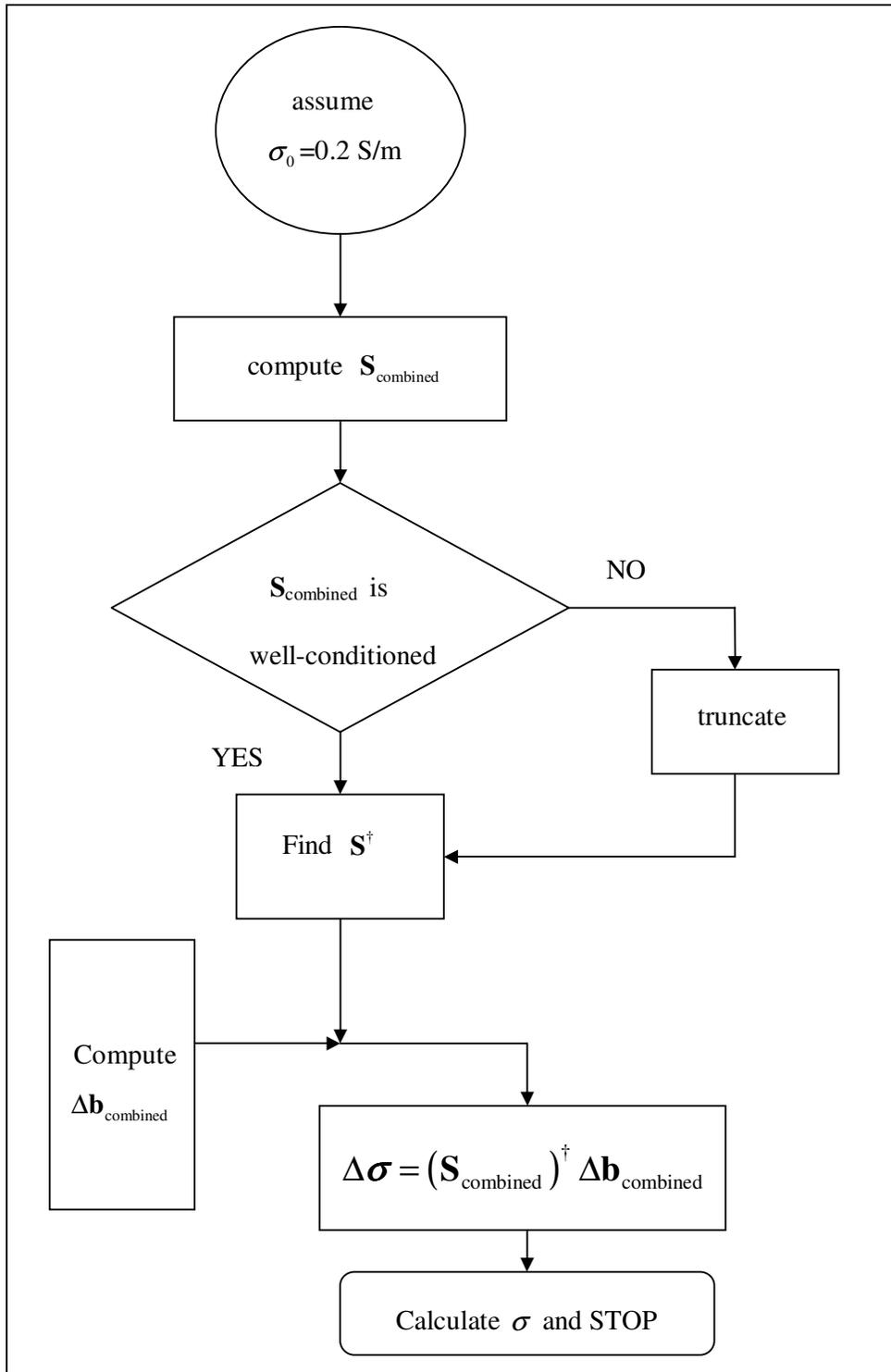


Figure 3.4: Flowchart for Sensitivity Matrix reconstruction algorithm.

3.4 Reconstruction by Algebraic reconstruction algorithm

3.4.1 Introduction

In this section, one of the magnetic flux density based reconstruction algorithms which is called Algebraic reconstruction algorithm is explained. The Algebraic reconstruction algorithm is an iterative reconstruction algorithm which uses only one component of magnetic flux density.

In 2003, Ider *et al* [8] have examined the reconstruction in MREIT and have proposed three methods for numerical implementations. Ider and Onart [18] examined one of the methods, explained in Ider *et al* [8], which uses non-linear matrix equations to find the resistivity distribution of the target object. They use only one component of magnetic flux density which is parallel to the main magnetic field. Also they said that at least two current injection patterns are necessary to reconstruct the resistivity distributions. After finding the resistivity distribution, the conductivity distribution can be found easily.

3.4.2 Problem Definition

The conductivity values used in this algorithm is isotropic. As explained in Section 3.1.2, an applied external current generates a magnetic flux density, $\vec{B}^j = (B_x^j, B_y^j, B_z^j)$, inside the object. The z-component of this generated magnetic flux density is used in Algebraic reconstruction algorithm.

A part of the following equation is used in this algorithm. Because this algorithm uses only B_z .

$$\nabla R \times \vec{J} = \frac{\nabla^2 \vec{B}}{\mu_0} \quad (3.33)$$

In equation (3.33), R is called the natural logarithm of resistivity.

$$\rho = e^R \quad (3.34)$$

The matrix form of equation (3.33) is shown below.

$$\begin{bmatrix} 0 & J_z & -J_y \\ -J_z & 0 & J_x \\ J_y & -J_x & 0 \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial R}{\partial x} \\ \frac{\partial R}{\partial y} \\ \frac{\partial R}{\partial z} \end{bmatrix} = \frac{1}{\mu_0} \begin{bmatrix} \nabla^2 B_x \\ \nabla^2 B_y \\ \nabla^2 B_z \end{bmatrix} \quad (3.35)$$

The row of this equation, which includes B_z , is important for Algebraic reconstruction algorithm. For this reason, we can write equation (3.36).

$$\frac{\partial R}{\partial x} J_y - \frac{\partial R}{\partial y} J_x = \frac{1}{\mu_0} \nabla^2 B_z \quad (3.36)$$

As seen above, the x-derivative and y-derivative of logarithmic resistivity is used. Central difference, which is shown in equation (3.37), is used for calculation of these derivatives.

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} \quad (3.37)$$

Central difference is used for interior regions of the subject. The edge regions forward difference or backward difference must be used.

With these information we can write equation (3.36) in the form of equation (3.38) for $(i,j)^{th}$ element of the imaged object.

$$\frac{R_{(i+1,j)} - R_{(i-1,j)}}{2\Delta x} J_{y(i,j)} - \frac{R_{(i,j+1)} - R_{(i,j-1)}}{2\Delta y} J_{x(i,j)} = \left(\frac{\nabla^2 B_z}{\mu_0} \right)_{(i,j)} \quad (3.38)$$

For computation of $\nabla^2 B_z$ at the i^{th} pixel, equation (3.39) is used. This equation is also called three-point difference.

$$\begin{aligned} \nabla^2 B_z(x_i, y_i, z_i) = & \frac{B_z(x_{i+1}, y_i, z_i) - 2B_z(x_i, y_i, z_i) + B_z(x_{i-1}, y_i, z_i)}{(\Delta x)^2} + \\ & + \frac{B_z(x_i, y_{i+1}, z_i) - 2B_z(x_i, y_i, z_i) + B_z(x_i, y_{i-1}, z_i)}{(\Delta y)^2} + \\ & + \frac{B_z(x_i, y_i, z_{i+1}) - 2B_z(x_i, y_i, z_i) + B_z(x_i, y_i, z_{i-1})}{(\Delta z)^2} \end{aligned} \quad (3.39)$$

In equation (3.38) and equation (3.39), Δx , Δy and Δz are the dimensions of each element in the x, y and z directions.

The number of elements in x and y directions are equal to K and M , respectively. For all elements inside the object, we can write the equation (3.38) as the following matrix form.

$$\mathbf{A} \mathbf{R} = \mathbf{b} \quad (3.40)$$

where, \mathbf{A} is the coefficient matrix with dimensions $(KM \times KM)$, \mathbf{R} is a $(KM \times 1)$ vector which includes the logarithmic resistivity values of all elements, \mathbf{b} is a $(KM \times 1)$ vector which includes the Laplacian of B_z .

Singular Value Decomposition (SVD) is used to test the performance of the calculated coefficient matrix. If necessary, truncation can be made.

The pseudo inverse of \mathbf{A} is calculated. With this pseudo inverse and the calculated Laplacian for magnetic field measurements are multiplied to find the logarithmic resistivity of all elements. This is shown in equation (3.41).

$$\mathbf{R} = \mathbf{A}^\dagger \mathbf{b} \quad (3.41)$$

where, \mathbf{A}^\dagger is the pseudo inverse of coefficient matrix.

After finding the logarithmic resistivity, the reconstructed resistivity distribution is calculated with equation (3.34). Then the conductivity distribution is calculated by equation (3.42).

$$\sigma = \frac{1}{\rho} \quad (3.42)$$

3.4.3 Implementation

For this reconstruction algorithm, two orthogonal current injection patterns ($N=2$) are used to inject 20 mA current.

First the measurements for B_z are taken for each element inside the object. Then $\nabla^2 B_z$ is calculated using the equation (3.39).

As seen from equation (3.38), the x-component and y-component of current density must be calculated. The components of current density are calculated for a uniform conductivity distribution. With these calculated values we can produce the coefficient matrix, which is defined in equation (3.40).

We use two different current injection patterns. The computed coefficient matrix for one current injection pattern will be ill-conditioned. For this reason, the coefficient matrices can be combined.

$$\mathbf{A}_{\text{combined}} = [\mathbf{A}_1 \quad \mathbf{A}_2]^\top \quad (3.43)$$

After computing the combined coefficient matrix, singular values of matrix $\mathbf{A}_{\text{combined}}$ are examined. The last two singular values are truncated. Because they are very close to zero.

After computing the coefficient matrix, equation (3.44) is used to find \mathbf{R} .

$$\mathbf{R}=(\mathbf{A}_{\text{combined}})^{\dagger}\mathbf{b}_{\text{combined}} \quad (3.44)$$

where, $\mathbf{b}_{\text{combined}}$ is the combined version of Laplacian of B_z for two current injection pattern.

The reconstructed conductivity distribution is calculated using equation (3.34) and equation (3.42).

The Algebraic reconstruction algorithm has the following steps:

Step-1: Measure B_z for all current injection pattern and calculate the Laplacian of B_z .

Step-2: Assume an initial conductivity distribution, σ_0 .

Step-3: Calculate J_x and J_y

Step-4: Compute the combined coefficient matrix, $\mathbf{A}_{\text{combined}}$.

Step-5: Find pseudo inverse of $\mathbf{A}_{\text{combined}}$.

Step-6: Solve $\mathbf{R}=(\mathbf{A}_{\text{combined}})^{\dagger}\mathbf{b}_{\text{combined}}$ and then calculate conductivity distribution.

Step-7: If the stopping criterion, $\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \epsilon$, for a predefined

tolerance value, is obtained the iterations will terminate. Otherwise go back to Step 3.

The flowchart for Algebraic reconstruction algorithm is shown in Figure 3.5.

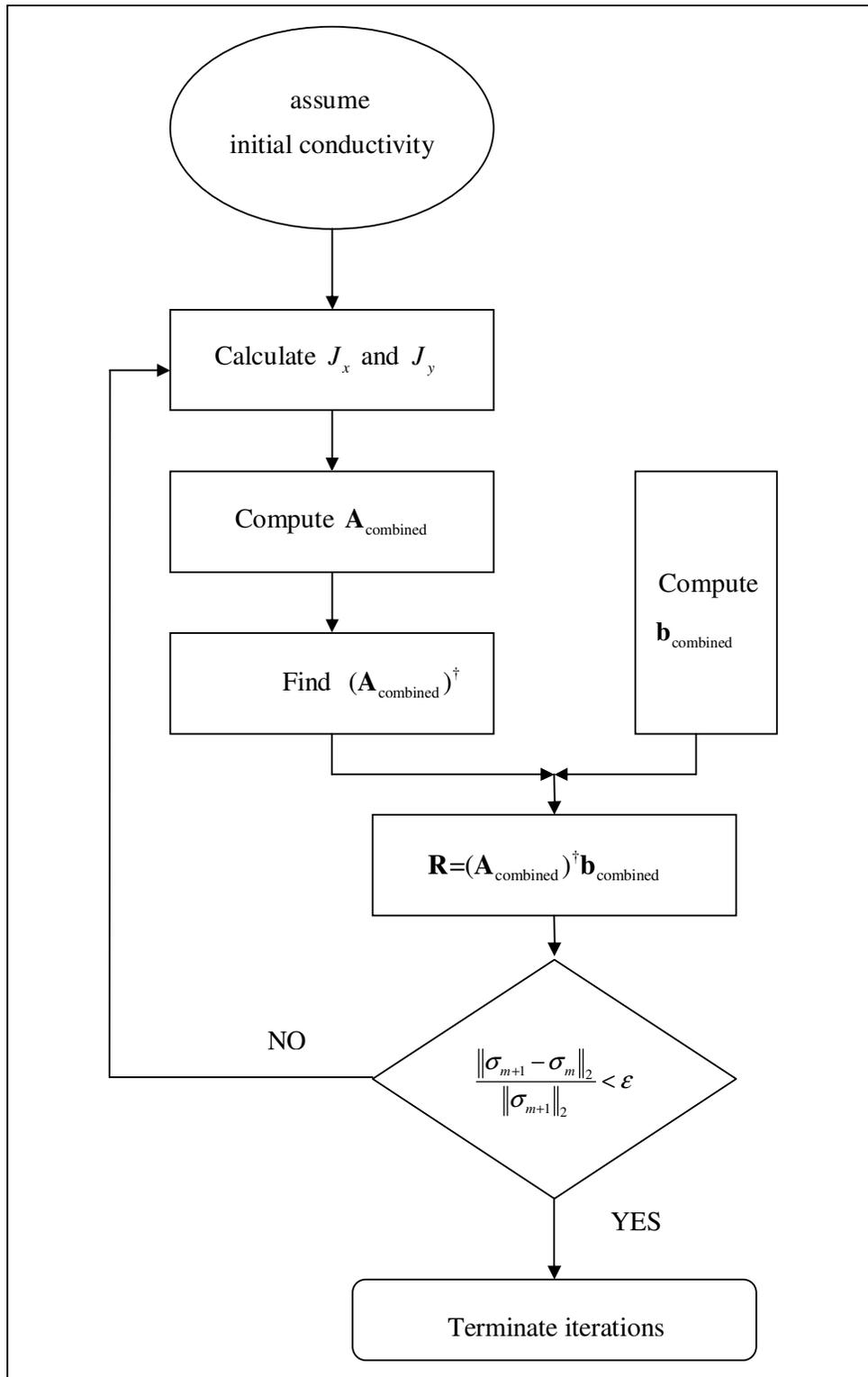


Figure 3.5: Flowchart for Algebraic reconstruction algorithm.

CHAPTER 4

SIMULATION AND COMPARISON

4.1 Introduction

In this chapter, computer models used in simulations and the reconstructed results which are taken from the simulations are explained. Different computer models and different conditions are used to test the performance of the reconstruction algorithms.

4.2 Computer models

Four different computer models are used to test the performance of four different reconstruction algorithms which are explained in previous chapters. The dimensions of the models are selected according to the experimental phantom used in experiments in METU 0.15 Tesla MRI system. In order to compare the reconstruction algorithm's performance, same dimensions are used for four different models.

The size of the imaging slice is chosen as 9 cm x 9 cm and the electrode size is selected 1/5 of the model's edge size. So the electrode size is 1.8 cm. The thickness of the model is 0.2 cm. For each model 20 mA current is used in each current injection.

4.2.1 Computer Model-1

Computer model-1 is shown in Figure 4.1. In this model, there are four different objects inside a background. The objects have isotropic conductivities. Two of the objects are more resistive than the background. The other two objects are more conductive than the background. The background conductivity has been chosen to be 0.2 S/m. The object conductivities are shown in Table 4.1.

Table 4.1: Conductivity values of computer model-1

Region	Object	Conductivity value (S/m)
1	Background	0.2
2	Big Square	0.1
3	Big Circle	0.4
4	Small Square	2
5	Small Circle	0.02

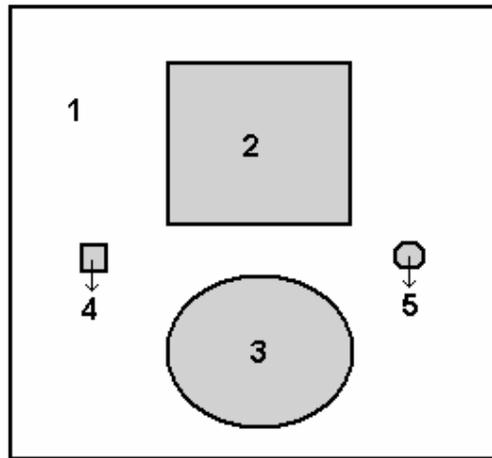


Figure 4.1: Computer model-1 with different conductivity regions.

Four electrodes are used in computer model-1. With these four electrodes, two different and orthogonal current injection patterns are used in model-1. These current injection patterns are shown in Figure 4.2.

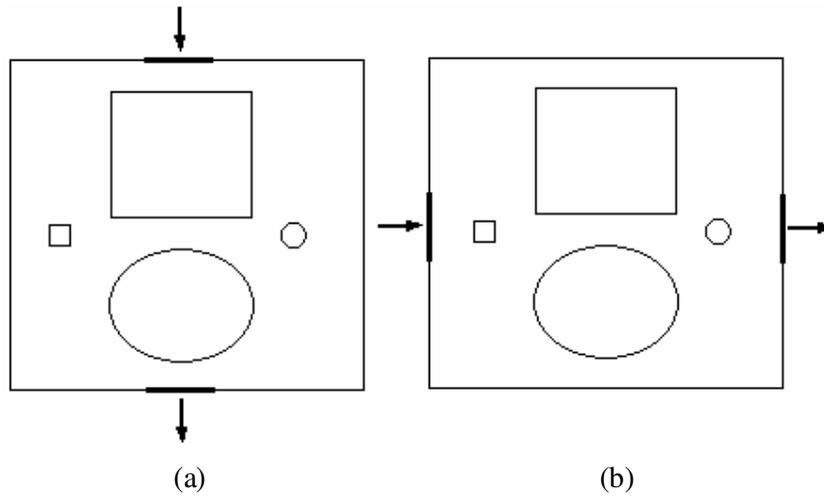


Figure 4.2: Current injection patterns. (a) Vertical current injection pattern. (b) Horizontal current injection pattern.

4.2.2 Computer Model-2

Computer model-2 is shown in Figure 4.3. There are two squares in the middle of the model. Both squares have $0.225 \text{ cm} \times 0.225 \text{ cm}$ which is a size of a pixel of the imaging slice. The background conductivity is equal to 0.2 S/m . There are two cases for this model. First one is conductive case. For conductive case, the conductivities of squares are 2 S/m . The other case is resistive case. For resistive case, the conductivities of squares are 0.02 S/m .

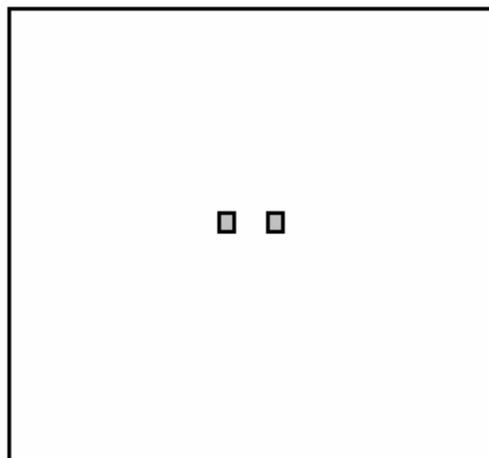


Figure 4.3: Computer model-2 with different conductivity regions.

This model is used to test the spatial resolution performance of four reconstruction algorithms by calculating the Full-Width-at-Half-Maximum (FWHM) values. Also the gap between two squares is changed. All positions for this model are shown in Table 4.2.

Table 4.2: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

Same as model-1, four electrodes are used to inject vertical current injection pattern and horizontal current injection pattern which are orthogonal.

4.2.3 Computer Model-3

Computer model-3 is shown in Figure 4.4. There are 9 squares on the diagonal of the imaging slice. The sizes of the squares are equal to the size of a pixel. For this model, there are two cases, which are conductive and resistive, to test the performance of all algorithms. Background conductivity is 0.2 S/m. Conductive square conductivity is 2 S/m and the resistive square conductivity is 0.02 S/m.

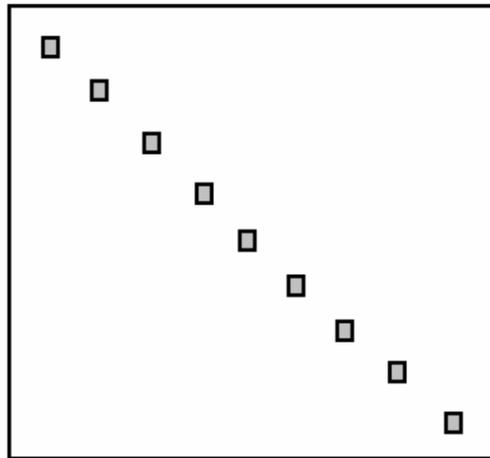


Figure 4.4: Computer model-3 with different conductivity regions.

In model-3, there are four electrodes. The position of electrodes are same as model-1. Two different and orthogonal current injection patterns are used. This model is used to test the spatial resolution performance of all algorithms for conductive and resistive cases.

4.2.4 Computer Model-4

Computer model-4 is shown in Figure 4.5. The background conductivity is equal to 0.2 S/m. There is a square at the center of the imaging slice. The size of the square is 2.25 cm x 2.25 cm. There are two different cases which are conductive case and resistive case. For conductive case, the square is more conductive than background. For resistive case, the square is more resistive than background.

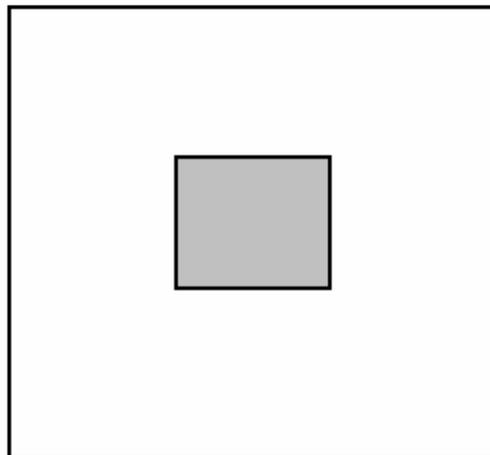


Figure 4.5: Computer model-4 with different conductivity regions.

For each case, there are 10 different contrast levels. This means, for each level square has different conductivity value. Different conductivity values of square for all cases of this model are shown in Table 4.3.

Table 4.3: Square conductivities for different cases and contrast levels

<i>Contrast Level</i>	1	2	3	4	5	6	7	8	9	10
Conductive case (Square conductivity S/m)	0.2	0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8	2.0
Resistive case (Square conductivity S/m)	0.2	0.1	0.066	0.05	0.04	0.033	0.028	0.025	0.022	0.02

4.2.5 Thorax Model

Thorax model is shown in Figure 4.6. This model simulates different tissues in thorax.

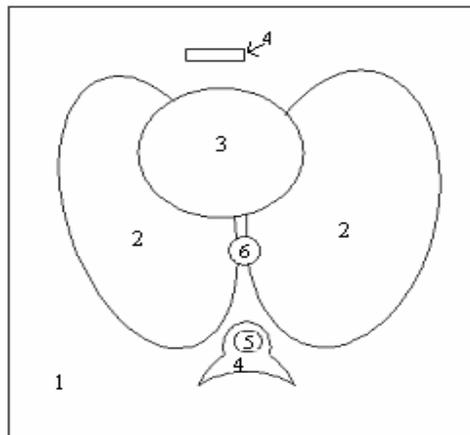


Figure 4.6: Thorax Model with different conductivity regions.

Each region has different conductivity value. The conductivity values of all regions are shown in Table 4.4.

Table 4.4: Conductivity values of Thorax Model

Region	Tissue Type	Conductivity value (S/m)
1	Skeletal muscle	0.2
2	Lung	0.0667
3	Heart	0.667
4	Bone	0.0006
5	Spinal cord (CSF)	1.5625
6	Aorta	0.667

4.3 Phantom used in experiments

The reconstruction algorithms tested with an experimental phantom. For experimental study, a 3D phantom is produced. The geometry of this phantom is shown in Figure 4.7(a). The inner size of this phantom is equal to $9 \times 9 \times 9$ cm. The middle part of phantom is used for taking the experimental data. The front view of this part is shown in Figure 4.7(b). Size of this part is $9 \times 9 \times 2$ cm. 1 cm slice thickness is used for taking the experimental data. The conductivities of the regions are shown in Table 4.5.

Table 4.5: Experimental model's conductivity values

Region	Object	Conductivity value (S/m)
1	Background	0.2
2	Square	0.1
3	Circle	0.4

To obtain these conductivity values, combination of different materials are used. The ratios of combination are shown in Table 4.6.

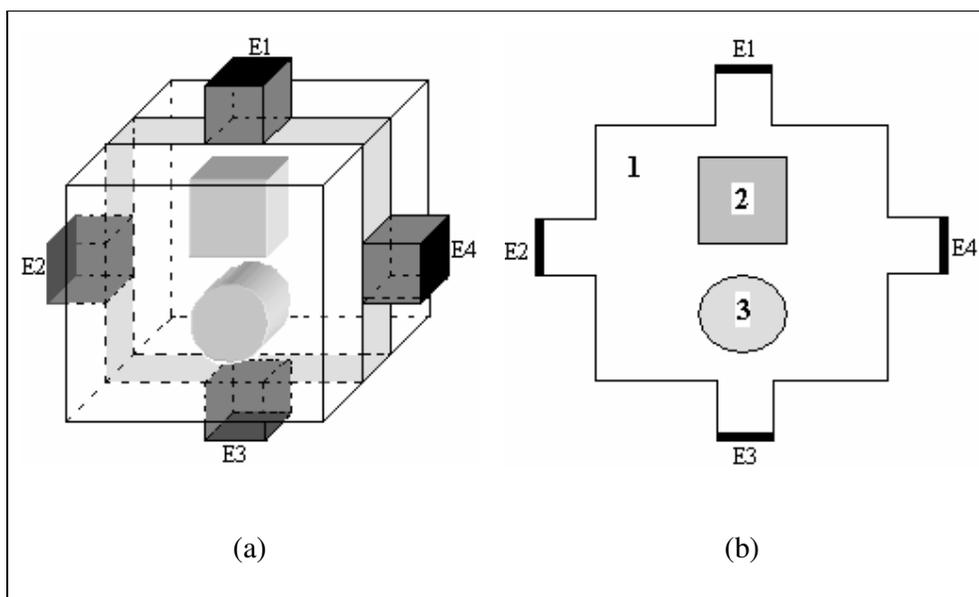


Figure 4.7: Experimental phantom. (a) 3D view of phantom. (b) Front view of imaging slice (with four recessed electrodes).

Table 4.6: Material ratios for different conductivity values.

Material	Conductivity value (0.1 S/m)	Conductivity value (0.2 S/m)	Conductivity value (0.4 S/m)
Water	100 ml	100 ml	100 ml
Salt	-	-	0.1 gr
CuSO ₄	0.1 gr	0.1 gr	0.1 gr
TX150 [26]	1 gr	0.2 gr	-
TX151 [26]	-	1.8 gr	1 gr
Agar [27]	0.5 gr	-	1 gr

For experiment, 20 mA current is injected to the phantom. Two different and orthogonal currents are used. One of them is vertical, the other is horizontal.

4.4 Noise used for computer models

To test the performance of reconstruction algorithms under noisy case, a random Gaussian noise which is explained by Scott *et al* [22], is used. The real Signal-to-

Noise Ratio (SNR) uses the ratio between the signal and the noise. But this has a disadvantage. This ratio can be changed by the applied current. If we increase the amount of the applied current, the amount of the magnetic flux density will be increased. The signal amount used in SNR calculation increases. So the SNR ratio decreases. Scott *et al* [22], uses a SNR which is independent from the direction and amount of the applied current. This SNR value is related only the MRI system which is used for experiments. They define SNR as:

$$SNR_{MR} = \frac{A}{\sigma} = \Psi_s \Delta x \Delta y \Delta z \sqrt{N \cdot T_s} M(x, y) \quad (4.1)$$

where, A is the magnitude of the noise-free pixel value of the corresponding MR image, Ψ_s is a system SNR, $\Delta x \Delta y \Delta z$ is the voxel volume, N is the total number of excitations (averages times phase encodes), T_s is the readout sampling time for one echo, and $M(x, y)$ is the magnetization. The phase error probability density function used in equation (4.1) is defined as:

$$f_{\Phi}(\phi) = \frac{1}{2} \exp\left(\frac{-a^2}{2}\right) + \frac{a \cos(\phi)}{2\sqrt{2\pi}} \exp\left(\frac{-a^2 \sin^2(\phi)}{2}\right) \operatorname{erfc}\left(\frac{-a \cos(\phi)}{\sqrt{2}}\right) \quad (4.2)$$

where, $a = \sqrt{2}SNR$, and Φ represents the phase error. Lorca [23] gives the details about system SNR calculation in his MSc thesis.

For this thesis, the values for SNR are chosen as 30, 20 and 13. Birgül *et al* [24] shows that the system SNR for 0.15 T METU MRI scanner is equal to 13. Also a 2 T MRI scanner has system SNR which is equal to 30 [22]. For these reasons, the SNR values are chosen to test the noise performance of reconstruction algorithms.

4.5 Error calculation and stopping criteria

To test the error performance of four reconstruction algorithms, the error values are calculated using the L_2 norm error. In experimental studies, the real conductivity values of target objects are not known. But in simulations, the real conductivity values of the objects are known. According to this information total error values can be calculated as:

$$\varepsilon_{\sigma, total} = \frac{\|\sigma_r - \sigma_i\|_2}{\|\sigma_r\|_2} \times 100\% \quad (4.3)$$

where, σ_r means real conductivity of the object, σ_i means reconstructed conductivity of the object and $\|\cdot\|_2$ is the L_2 norm. L_2 norm is defined in equation (4.4).

$$\|\vec{a}\| = \left(\sum_i a_i^2 \right)^{1/2} \quad (4.4)$$

where, a_i is the i^{th} element of \vec{a} .

If someone wants to calculate the error value in a specific region, equation (4.5) must be used.

$$\varepsilon_{\sigma, region} = \frac{\|\sigma_{r, region} - \sigma_{i, region}\|_2}{\|\sigma_{r, region}\|_2} \times 100\% \quad (4.5)$$

where, $\sigma_{r, region}$ and $\sigma_{i, region}$ are the real conductivity distribution and reconstructed conductivity distribution for the specific region.

For computer models, error calculation is easy. Since, real conductivity distributions are known. But in experimental studies, the real conductivity distributions are not known. For this reason, a stopping criterion must be defined

to stop the iterations for iterative algorithms. To prevent infinite iterations a tolerance value is defined to terminate iterations. Stopping criteria is shown as:

$$\frac{\|\sigma_{m+1} - \sigma_m\|_2}{\|\sigma_{m+1}\|_2} < \varepsilon \quad (4.6)$$

where, ε is the tolerance value, σ_m is the reconstructed conductivity after m^{th} iteration, σ_{m+1} is the reconstructed conductivity after $(m+1)^{th}$ iteration and $\|\cdot\|_2$ is the L_2 norm defined in equation (4.4).

4.6 Spatial Resolution

Spatial resolution of an imaging system is defined as the system ability to distinguish minimum size object. Also the spatial resolution gives the degree of deblurring in an image. For this purpose, the Point Spread Function (PSF) is examined. Since, it includes the information about the spatial resolution.

The spatial resolution performances of four reconstruction algorithms are tested by using the model-2. In this model, there are two small squares which have a size equal to one pixel size. The PSF at the middle row of the model is examined. Also the PSF according to the gap between two pixels is examined. To find the spatial resolution performance, the maximum and the minimum values on PSF are taken. According to these values the Full-Width-at-Half-Maximum (FWHM) value can be calculated easily. This is shown in Figure 4.8. FWHM value gives the information about how the gap between two squares effect the reconstructed conductivity and spatial resolution.

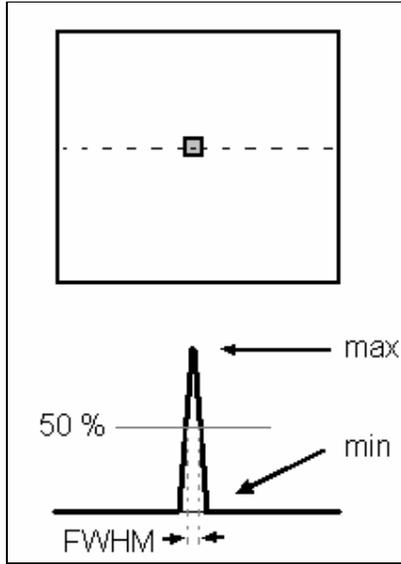


Figure 4.8: FWHM on the PSF of the middle row of the imaging slice.

4.7 Simulation Results for Magnetic Flux Density Based Reconstruction Algorithms

In the following sections, the results of four magnetic flux density based reconstruction algorithms (which are explained in chapter 3) are given and discussed.

4.7.1 Reconstruction by Harmonic B_z algorithm

In this section, the reconstructed conductivity distributions of four different models are examined and discussed by using the Harmonic B_z algorithm explained in chapter 3.

As explained in chapter 3, this iterative algorithm uses only the z component of the magnetic flux density (B_z), which is parallel to the main magnetic field. First, the B_z is taken and the Laplacian of B_z is calculated. Then, the x -gradient and y -gradient of the potential for uniform case are calculated. With these values, the x -

derivative and y-derivative of conductivity distribution can be found. At the end, the conductivity distribution can be calculated by using the line integral.

4.7.1.1 Results for Model-1

As explained in Section 4.2.1, there are five different conductivity regions in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

The real conductivity distribution for model-1 is shown in Figure 4.9(a). The reconstructed conductivity distributions after iteration 1 and after iteration 7 are shown in Figure 4.9(b) and Figure 4.9(c). These results are taken for noise free data.

The 7th iteration is chosen. Since, Harmonic B_z algorithm starts to be stable after iteration 7. This is easily seen from the total error figure which is shown in Figure 4.10(a). All regions' error characteristic can be seen in Figure 4.10. The reconstructed error values of all regions, for iteration 7 are shown in Table 4.7.

As seen from Figure 4.9 and Figure 4.10, Harmonic B_z algorithm reconstructs conductivity successfully without added noise.

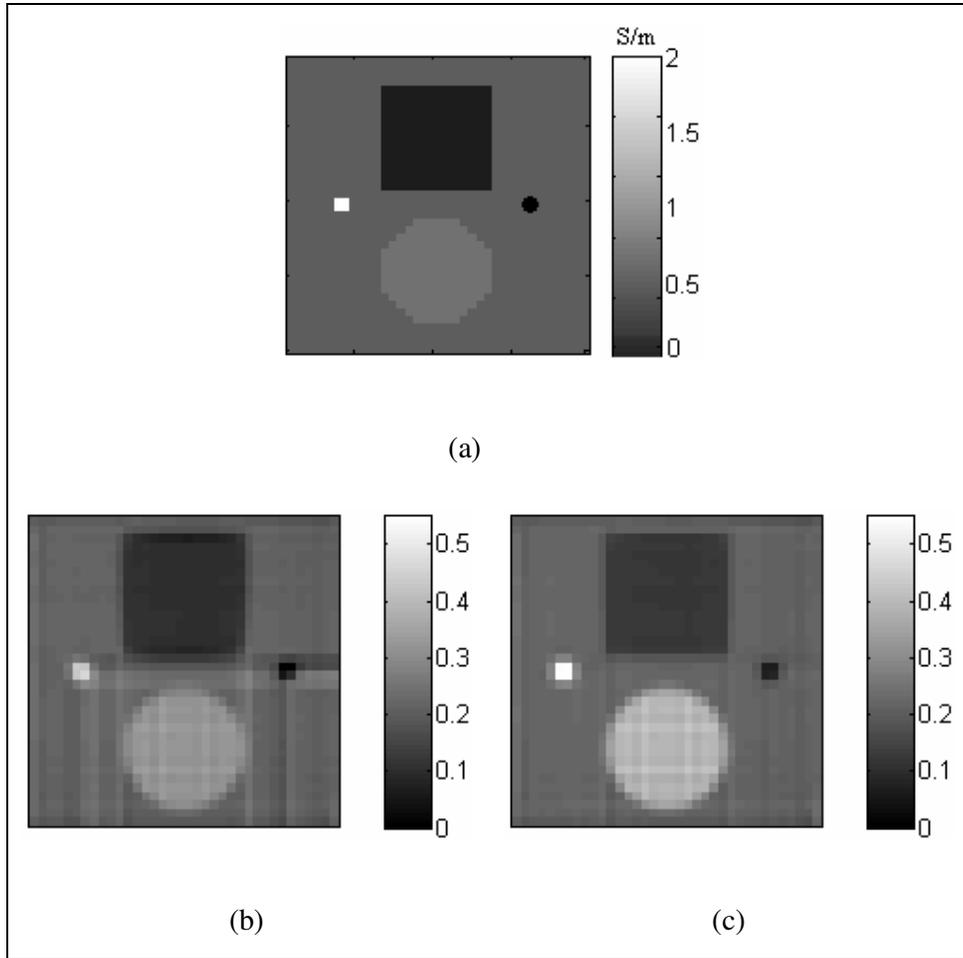


Figure 4.9: Reconstruction by Harmonic B_z algorithm for Model-1 (noise free data). (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution after iteration 1. (c) Reconstructed conductivity distribution after iteration 7.

Table 4.7: Reconstructed error values after iteration 7

$\mathcal{E}_{\sigma, total}$ (%)	$\mathcal{E}_{\sigma, background}$ (%)	$\mathcal{E}_{\sigma, bigsquare}$ (%)	$\mathcal{E}_{\sigma, bigcircle}$ (%)	$\mathcal{E}_{\sigma, smallsquare}$ (%)	$\mathcal{E}_{\sigma, smallcircle}$ (%)
33.83	11.17	23.05	8.08	72.67	264.30

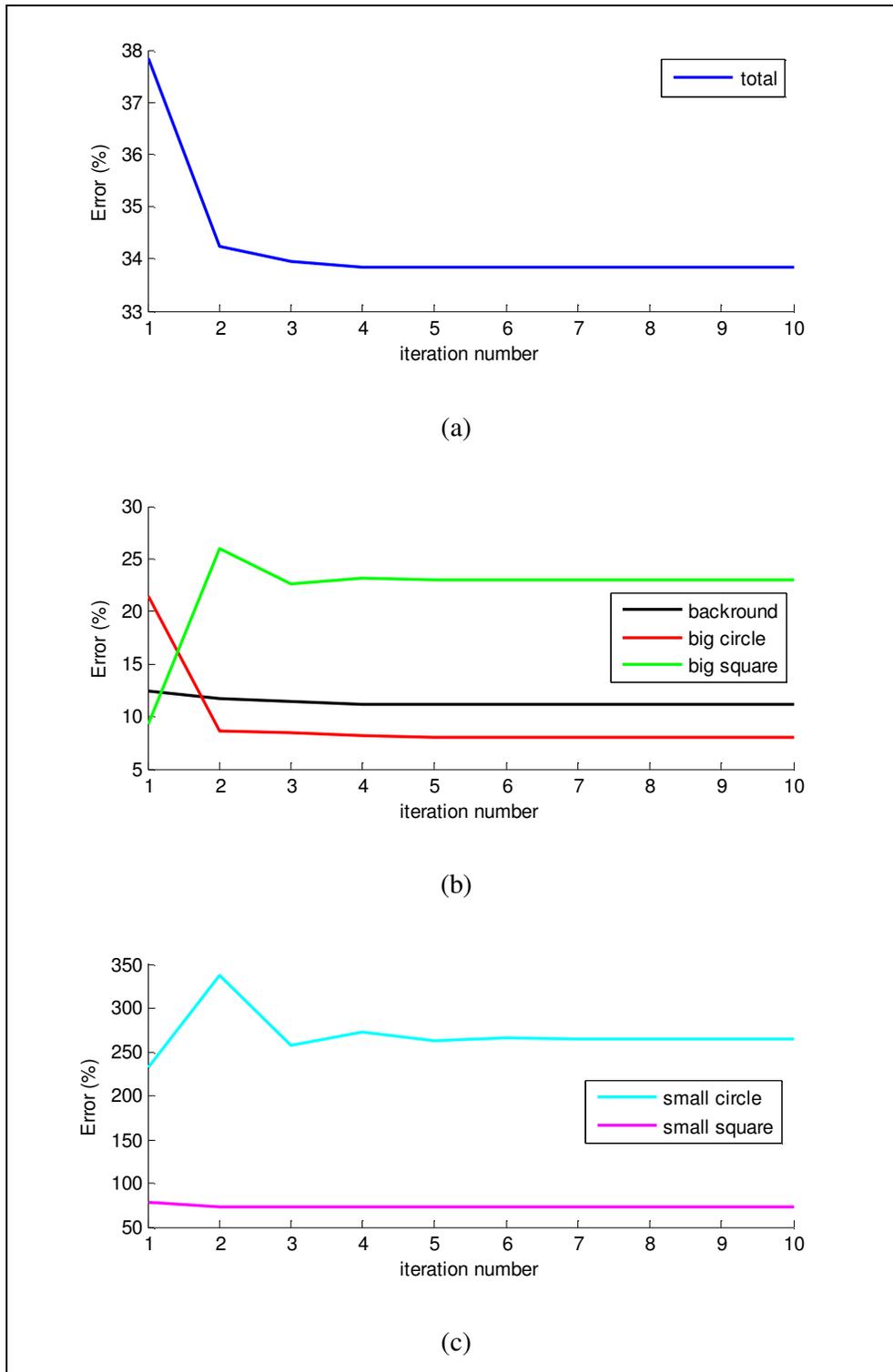


Figure 4.10: Reconstructed conductivity errors by Harmonic B_z algorithm for Model-1 (noise free data). (a) Total error. (b) Background error, Big Circle error, Big Square error. (c) Small Circle error, Small Square error.

Then this reconstruction algorithm tested with different noise levels, SNR 30, SNR 20 and SNR 13. The reconstructed conductivity distributions are shown in Figure 4.11.

As seen from Figure 4.11, the reconstructed conductivity distributions for noisy data are not good as noise free data. This result is expected. Because, Harmonic B_z algorithm uses Laplacian of B_z , which means twice differentiation of both B_z and noise.

Also this reconstruction algorithm uses line integral after finding the x-derivative of conductivity and y-derivative of conductivity. According to these reasons, it is seen that SNR must be high to take good results.

Increasing the number of current injection patterns and increasing the amount of current give good reconstructed conductivity results at low SNR values.

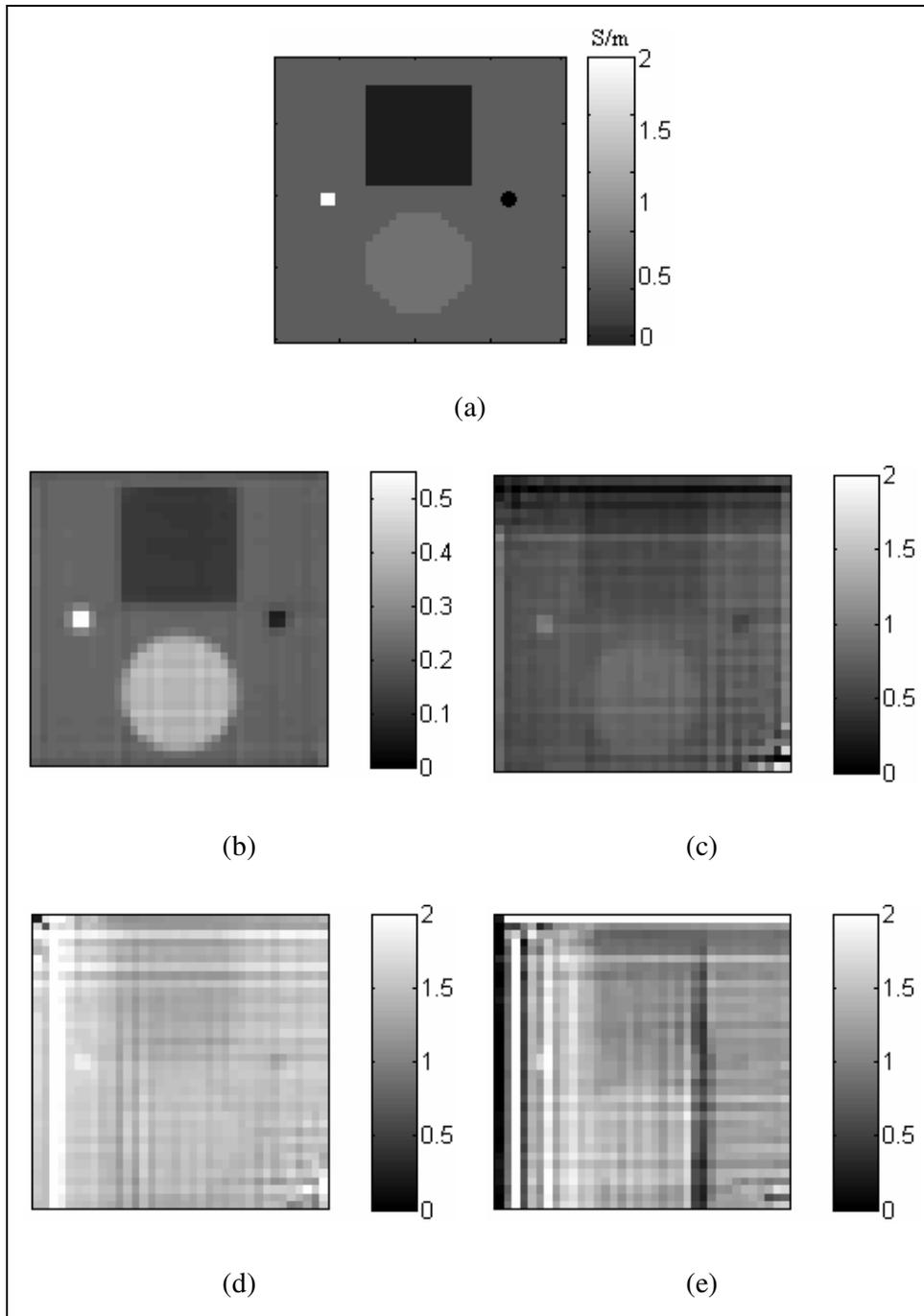


Figure 4.11: Reconstruction by Harmonic B_z algorithm for Model-1 (all images for iteration 7). (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution, noise free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

Total error plot comparison can be seen in Figure 4.12. As seen in figure, the minimum SNR value must be 30 to reconstruct good conductivity distributions. This algorithm is not stable for low SNR vales. But it reconstructs conductivity successfully without added noise.

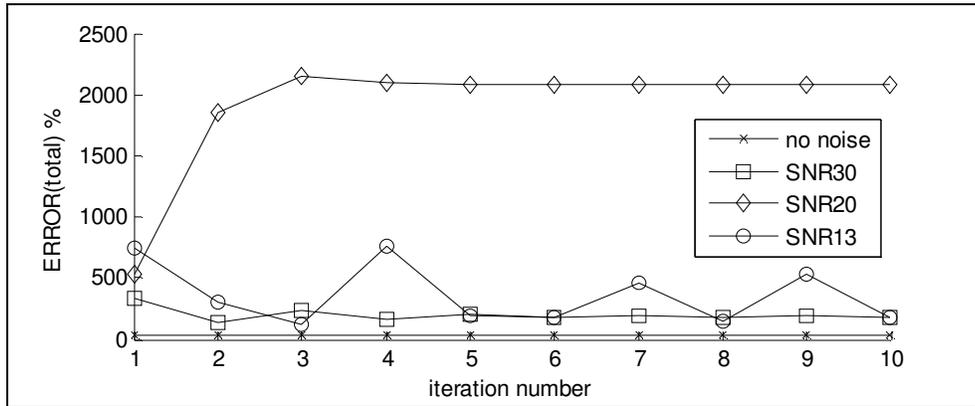


Figure 4.12: Comparison of reconstruction errors for Harmonic B_z algorithm, for different SNR values and no noise data, for Model-1.

4.7.1.2 Results for Model- 2

As explained in Section 4.2.2, there are two square objects with different conductivity values than background, in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns. There are two cases for this model. First one is conductive case. For conductive case, the conductivities of squares are 2 S/m. The other case is resistive case. For resistive case, the conductivities of squares are 0.02 S/m. Also there are 10 different positions, which are shown in Table 4.8, used to test the spatial resolution performance of Harmonic B_z reconstruction algorithm by calculating the Full-Width-at-Half-Maximum (FWHM) values.

Table 4.8: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

The reconstructed conductivity distributions for both cases with position 4 are shown in Figure 4.13. These results are taken without added noise. Also for each case and position, 4 iterations are used. Since, the algorithm starts to converge after 4th iteration.

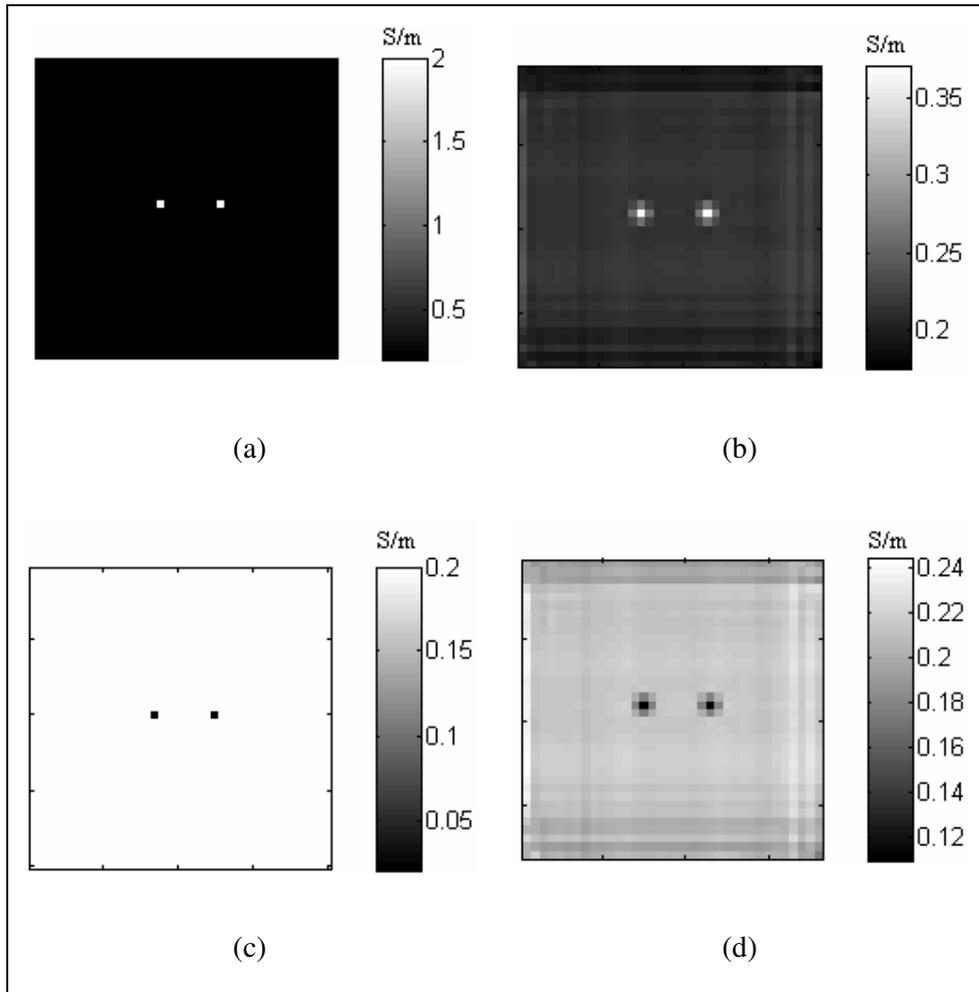


Figure 4.13: Reconstruction by Harmonic B_z algorithm for Model-2, position-4. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

Also, the Point Spread Functions for the middle row of the imaging slice and position-1 are shown in Figure 4.14. Calculated FWHM values according to change of gap between two squares for all positions are shown in Figure 4.15.

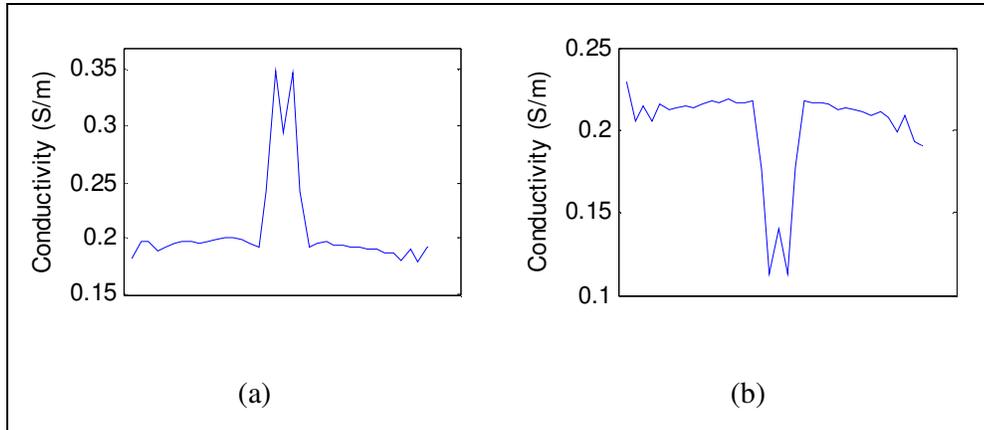


Figure 4.14: PSF for middle row of imaging slice by Harmonic B_z algorithm for Model-2, position-1. (a) For conductive case. (b) For resistive case.

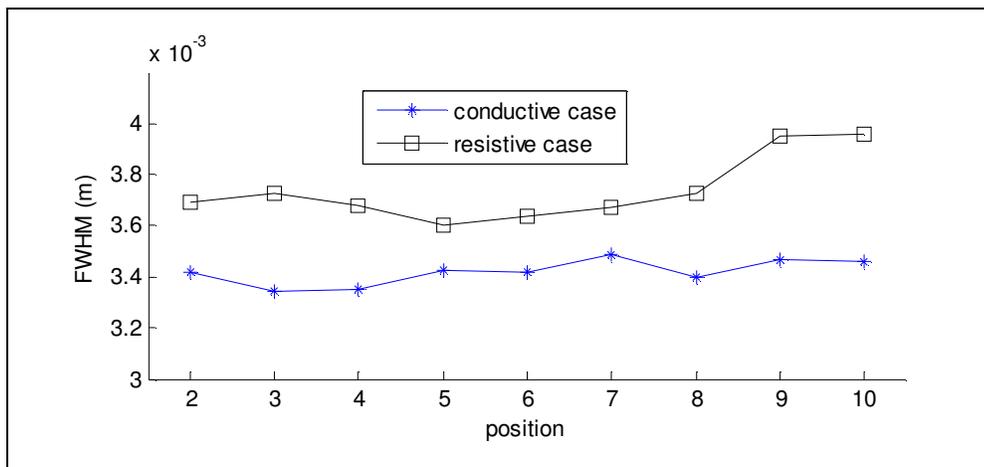


Figure 4.15: FWHM values according to change of gap between two squares for all positions.

As seen from Figure 4.15, the calculated FWHM values are close for conductive and resistive cases. For position-1, the gap between two squares is 1 pixel. Only for position-1, two reconstructed squares are merged. This means a pixel affects the other pixels which are placed around the target pixel. But for other positions, the calculated FWHM values are independent from the position of two squares.

These results show that Harmonic B_z algorithm can reconstruct an object which is equal to a pixel size. But target object affects the pixels which are placed around this target object.

4.7.1.3 Results for Model-3

In Model-3, there are 9 small squares on the diagonal of the imaging slice. The sizes of the squares are equal to the size of a pixel.

For this model, there are two cases which are conductive and resistive, to test the performance of algorithm. Background conductivity is 0.2 S/m. Conductive square conductivity is 2 S/m and the resistive square conductivity is 0.02 S/m.

Reconstructed conductivity distributions for conductive case and resistive case are shown in Figure 4.16.

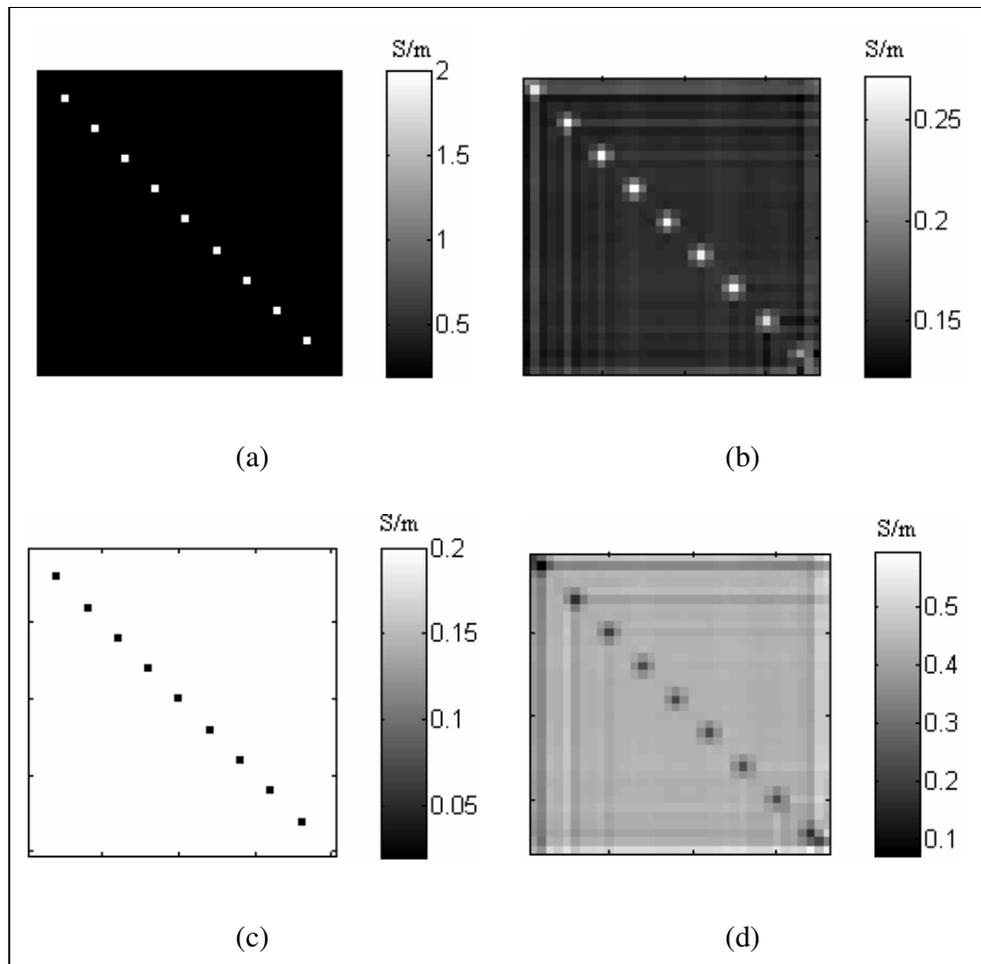


Figure 4.16: Reconstruction by Harmonic B_z algorithm for Model-3 (after iteration 8). (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

The effect of the line integral, which is used in reconstruction algorithm, can be seen easily. Especially on the top left corner, a high change in the conductivity value can affect the rest of the column or row of the imaging slice. This is a disadvantage for this reconstruction algorithm.

4.7.1.4 Results for Model- 4

In Model-4, there is a square at the center of the imaging slice. The size of the square is 2.25 cm x 2.25 cm. There are two different cases which are conductive case and resistive case. For conductive case, the square is more conductive than background. For resistive case, the square is more resistive than background. Also there are 10 different contrast levels. This means, for each contrast level square has different conductivity value. The used conductivity values of square for all cases are shown in Table 4.3 in the Section 4.2.4.

For conductive case and resistive case, 4 iterations are used for all contrast levels. The algorithm starts to converge at 4th iteration.

For conductive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.4$ S/m (contrast level-2) are shown in Figure 4.17.

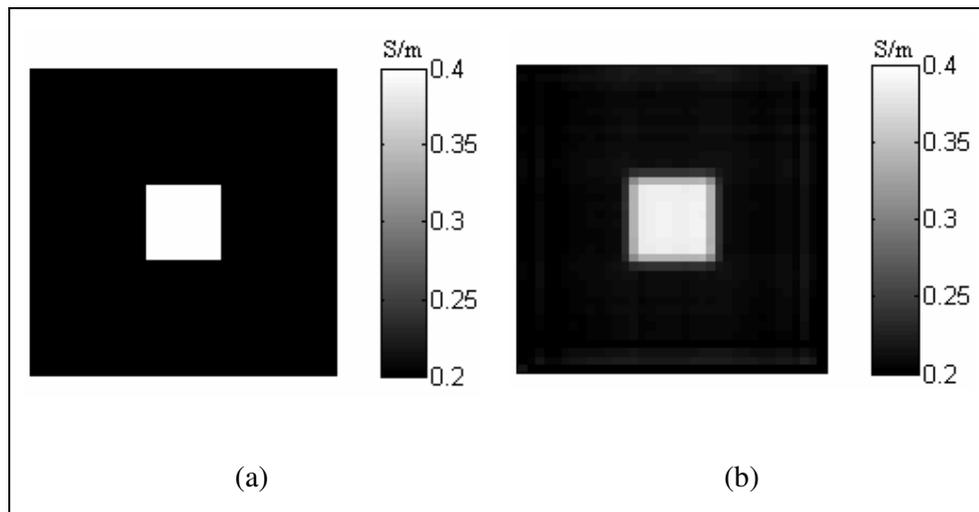


Figure 4.17: Reconstruction by Harmonic B_z algorithm for Model-4, contrast level-2. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case (after iteration 4).

For resistive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.1$ S/m (contrast level-2) are shown in Figure 4.18.

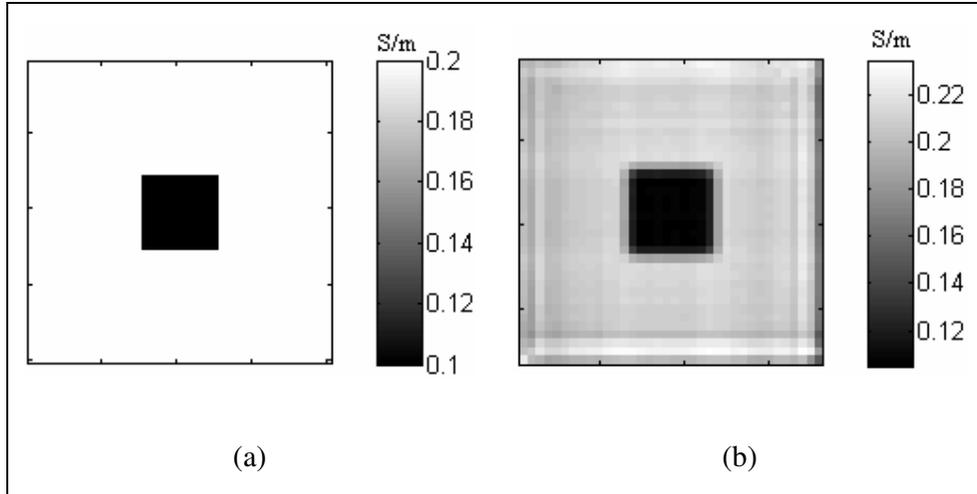


Figure 4.18: Reconstruction by Harmonic B_z algorithm for Model-4, contrast level-2. (a) Real conductivity distribution for resistive case. (b) Reconstructed conductivity distribution for resistive case (after iteration 4).

For conductive case, the square's conductivity increases when the contrast level increases. But for resistive case, the square's conductivity decreases when the contrast level increases. The square error plots according to the change of the contrast levels for both cases are shown in Figure 4.19.

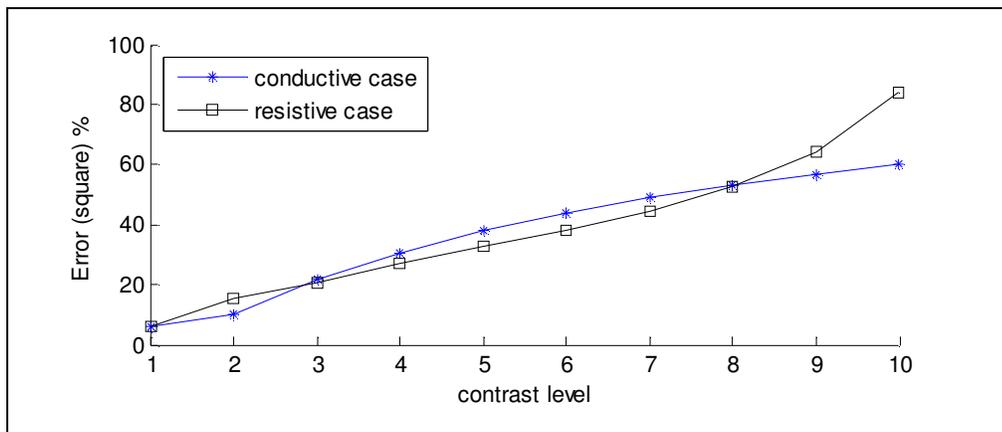


Figure 4.19: Error plots of the square for 10 different contrast levels.

As seen from Figure 4.19, if we increase the difference between square conductivity value and background conductivity value, the error value of the reconstructed square will increase. This means if the contrast is high, the reconstruction error will be high.

Also in Figure 4.20, the plot of the real conductivity values of square and mean of the reconstructed conductivity values of square for each contrast level can be seen.

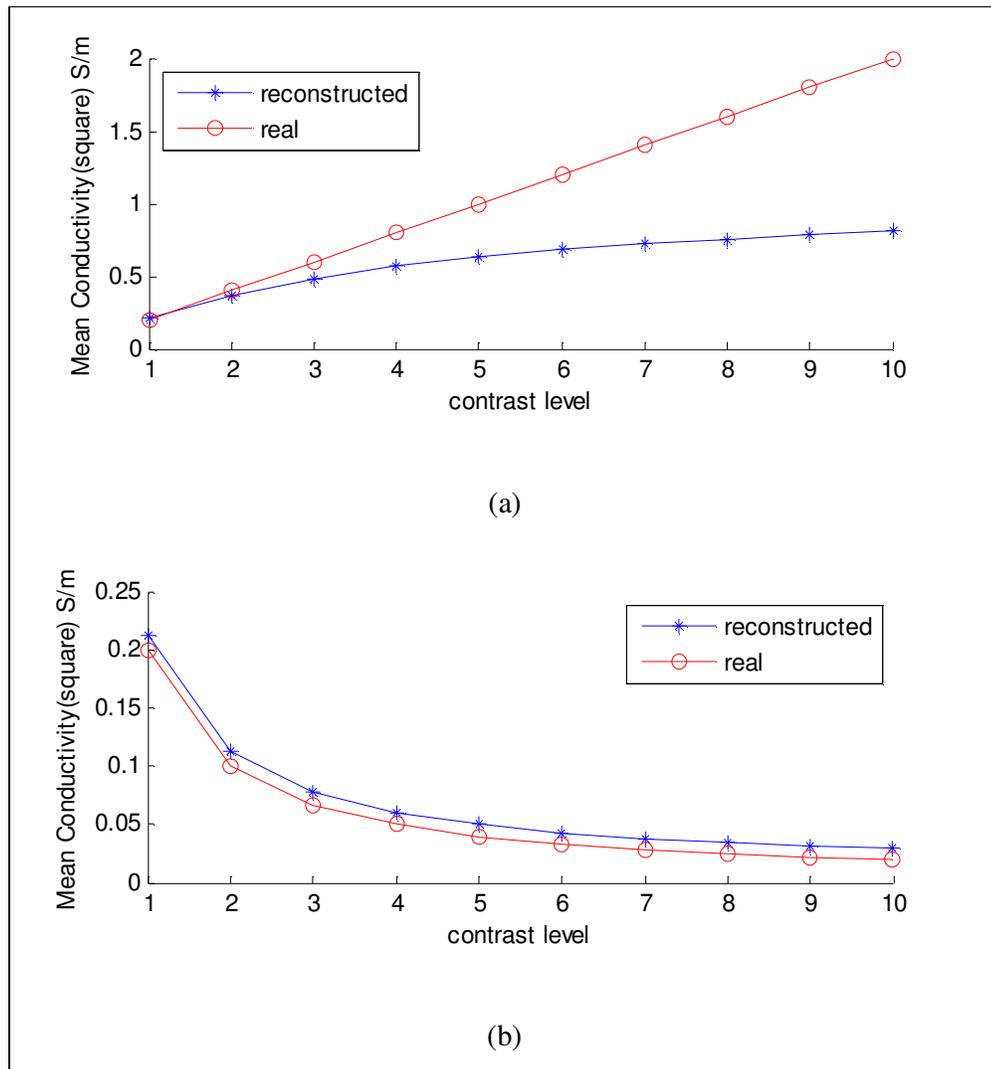


Figure 4.20: Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels. (a) For conductive case. (b) For resistive case.

4.7.1.5 Results for Thorax Model

The model shown in Section 4.2.5 is used to test the performance of Harmonic B_z for noise-free data and noisy data. The reconstructed conductivity distributions are shown in Figure 4.21.

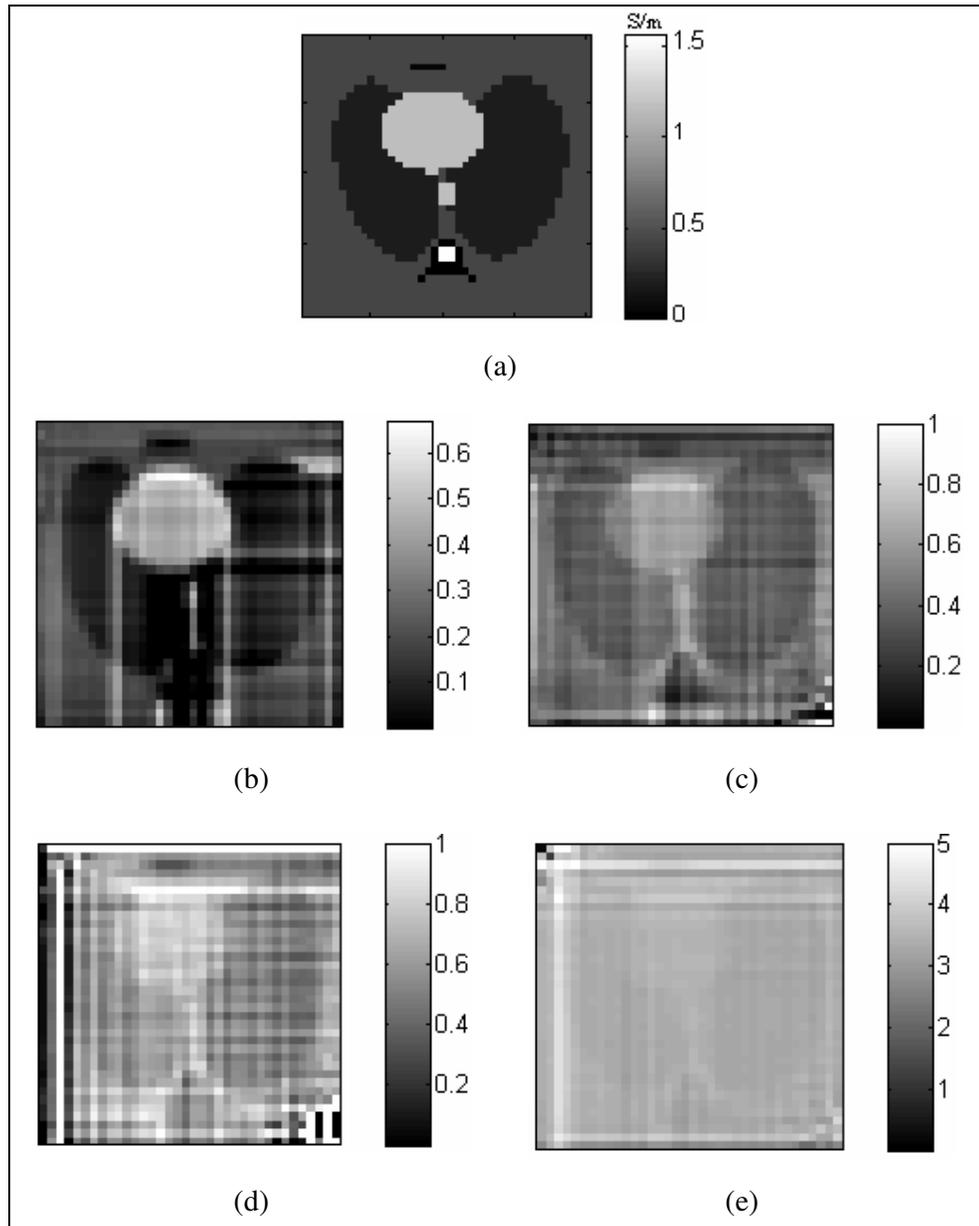


Figure 4.21: Reconstruction by Harmonic B_z algorithm for Thorax Model (all images for iteration 2). (a) Real conductivity distribution. (b) Reconstructed conductivity distribution, noise-free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Figure 4.21, the reconstructed conductivity distributions for noisy data are not good as noise free data. The total reconstruction errors for noisy case and noise free case are shown in Table 4.9.

Table 4.9: Reconstructed error values for thorax model.

	<i>Noise-free case</i>	<i>SNR30</i>	<i>SNR20</i>	<i>SNR13</i>
$\mathcal{E}_{\sigma, total}$ (%)	52.65	105.67	357.9	501.6

4.7.2 Reconstruction by Variational Gradient B_z Algorithm.

In this section, the reconstructed conductivity distributions of four different models are examined and discussed by using the Variational Gradient B_z (VGB $_z$) algorithm explained in chapter 3.

As explained in chapter 3, this iterative algorithm uses only the z component of the magnetic flux density (B_z), which is parallel to the main magnetic field. Different from the Harmonic B_z algorithm, VGB $_z$ algorithm differentiate B_z only once. Firstly the B_z is taken and the x-gradient and y-gradient of B_z is calculated. Then the x-gradient and y-gradient of the potential for uniform case are calculated. At the end the conductivity distribution can be calculated by using these gradient values.

4.7.2.1 Results for Model- 1

As explained in Section 4.2.1, there are five different conductivity regions in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

The real conductivity distribution for model-1 is shown in Figure 4.22(a). The reconstructed conductivity distributions after iteration 1 and after iteration 9 are shown in Figure 4.22(b) and Figure 4.22(c). These results are taken for noise free data.

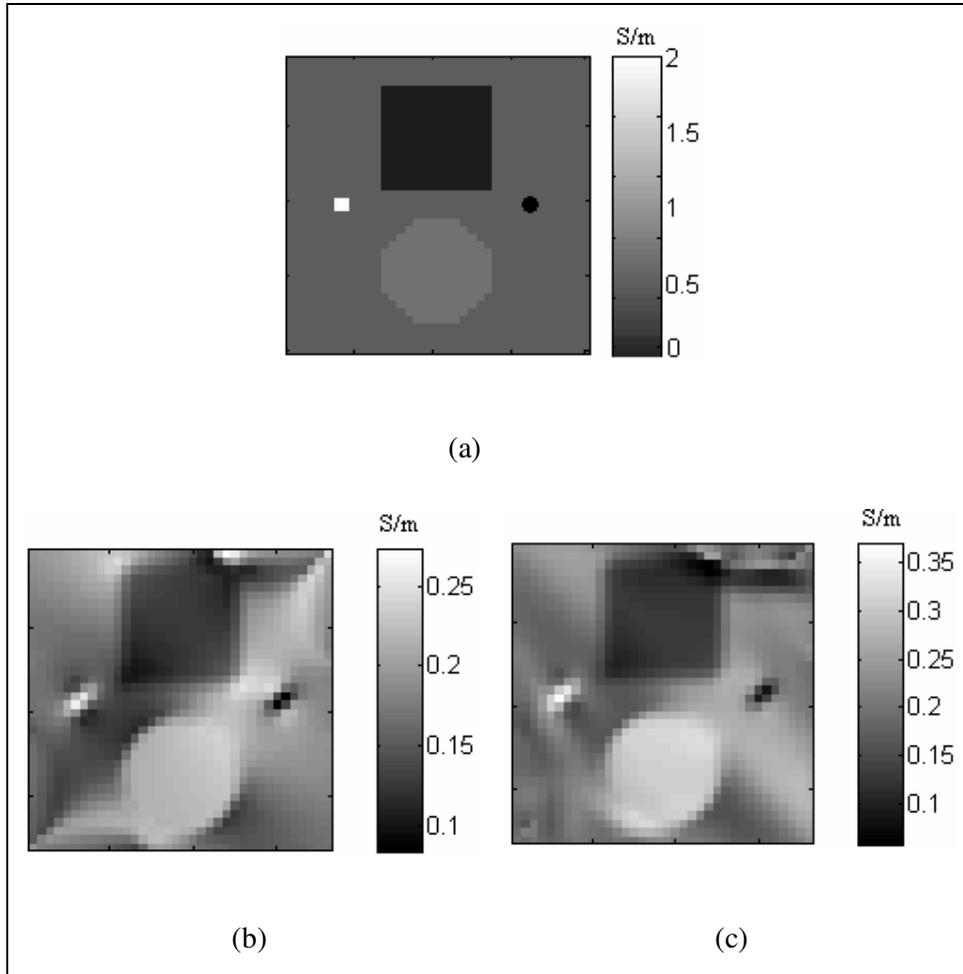


Figure 4.22: Reconstruction by VGB_z algorithm for Model-1 (noise free data). (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution after iteration 1. (c) Reconstructed conductivity distribution after iteration 9.

As seen from Figure 4.22, the reconstructed conductivity distribution at the corners of the imaging slice is not good as the reconstructed conductivity distribution at the other areas. This is a result of the differentiation of B_z along x and y directions. In VGB_z algorithms x -gradient and y -gradient of B_z are calculated and summed. Especially at the edges of electrodes, this summation can affect the reconstructed conductivity distribution. This effect starts from the edges of electrodes and spread to the corners of the image. In Model-1, the ratio of electrode size to model edge is equal to $1/5$. This means electrode size is equal to 1.8 cm for a 9 cm model edge. Tests on this algorithm show that, if electrode-

edge ratio is equal or greater than $3/5$, VGB_z algorithm will successfully reconstruct the conductivity distributions at the corner of the imaging slice.

The effect of the Electrode-Edge Ratio (EER) is shown in Figure 4.23.

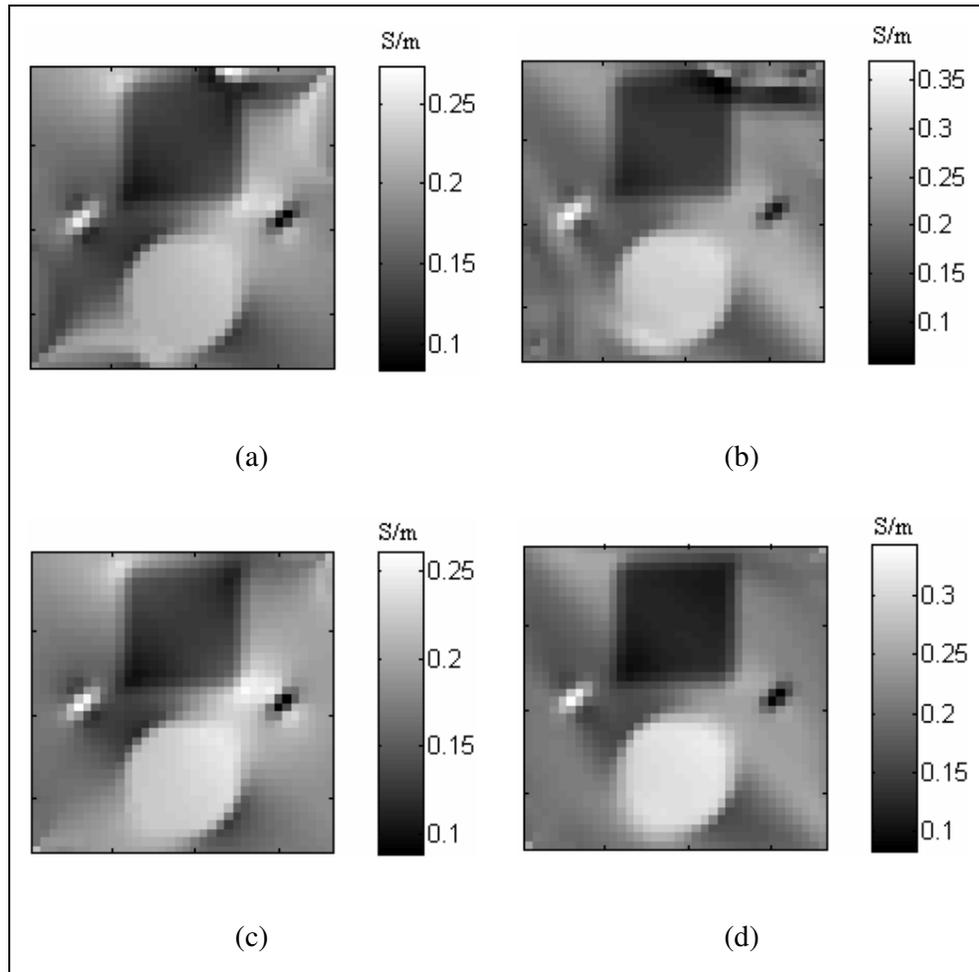


Figure 4.23: Effect of electrode-edge ratio (noise free data) (a) Reconstructed conductivity distribution, 1 iteration, $EER=1/5$. (b) Reconstructed conductivity distribution, 9 iterations, $EER=1/5$. (c) Reconstructed conductivity distribution, 1 iteration, $EER=3/5$. (d) Reconstructed conductivity distribution, 9 iterations, $EER=3/5$.

In Figure 4.23(a) and Figure 4.23(b), there are artifacts at the corners of the reconstructed conductivity distribution for small EER. If we increase the electrode size, these artifacts will disappear. This is shown in Figure 4.23(c) and Figure 4.23(d).

The 9th iteration is chosen. Since, VGB_z algorithm starts to be stable after iteration 9. This is easily seen from the total error figure which is shown in Figure 4.24(a). All regions' error characteristics are shown in Figure 4.24. The reconstructed error values of all regions, for iteration 1 and iteration 9 are shown in Table 4.10.

Table 4.10: Reconstructed error values of all regions, for iteration-1, iteration-9

	<i>After iteration 1</i>	<i>After iteration 9</i>
$\mathcal{E}_{\sigma, total} (\%)$	48.72	42.44
$\mathcal{E}_{\sigma, background} (\%)$	18.58	20.11
$\mathcal{E}_{\sigma, bigsquare} (\%)$	32.96	30.82
$\mathcal{E}_{\sigma, bigcircle} (\%)$	45.21	24.73
$\mathcal{E}_{\sigma, smallsquare} (\%)$	88.14	83.70
$\mathcal{E}_{\sigma, smallcircle} (\%)$	477.01	501.88

As seen from Table 4.10, the small circle, which has very small conductivity value than the other regions, is reconstructed with a high reconstruction error.

As seen from Figure 4.22 and Figure 4.24, VGB_z algorithm reconstructs conductivity successfully without added noise. Only there are artifacts at the corners of reconstructed conductivity distribution for small electrode size.

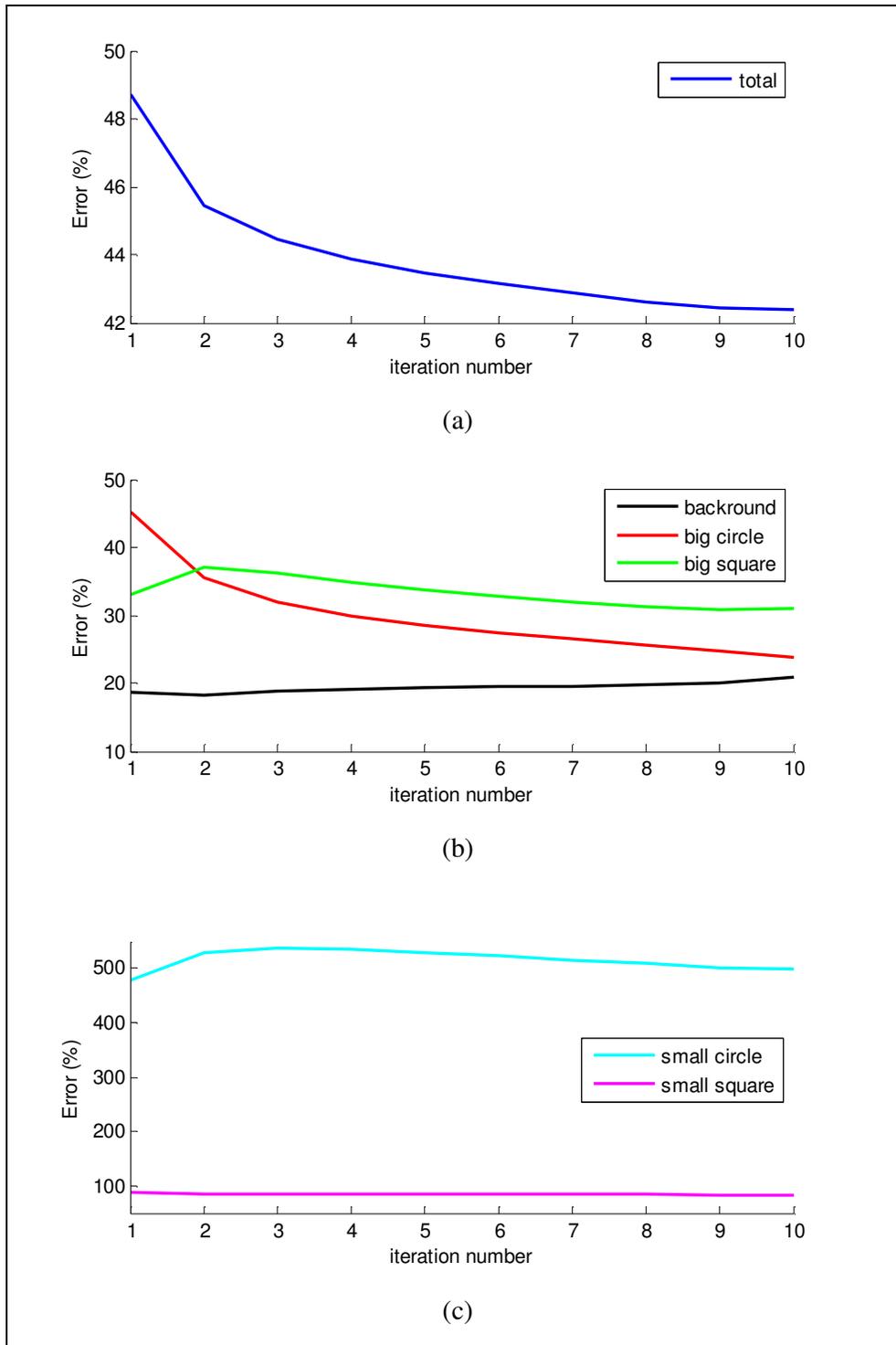


Figure 4.24: Reconstructed conductivity errors by VGB_z algorithm for Model-1 (noise free data). (a) Total error. (b) Background error, Big Circle error, Big Square error. (c) Small Circle error, Small Square error.

This reconstruction algorithm tested with different noise levels: SNR 30, 20 and 13. The reconstructed conductivity distributions are shown in Figure 4.25.

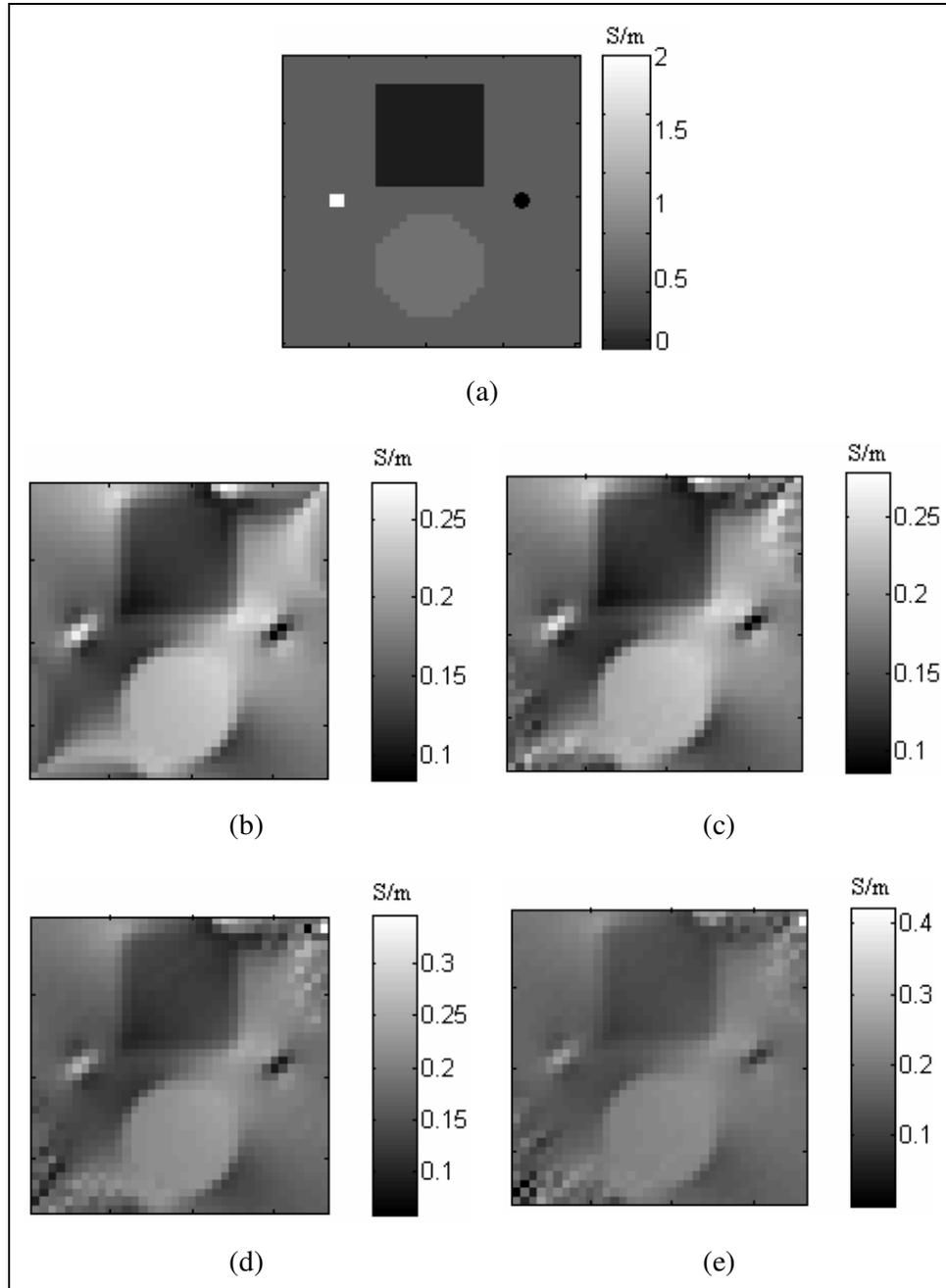


Figure 4.25: Reconstruction by VGB_z algorithm for Model-1. (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution noise free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Figure 4.25, the reconstructed conductivity distributions for noisy data are good as noise free data. VGB_z algorithm has a better noise immunity. This result is expected. Because, VGB_z algorithm uses only one differentiation of B_z , which means one differentiation of both B_z and noise. So this algorithm gives good results for both noisy and noise free data.

If we increase the number of current injection patterns, amount of currents, and size of electrodes, this algorithm will give good reconstructed conductivity distributions.

Total error plot comparison is shown in Figure 4.26. As seen from the figure, this algorithm reconstructs conductivity successfully with low SNR values. But reconstruction errors for low SNR value will increase with the increasing number of iterations.

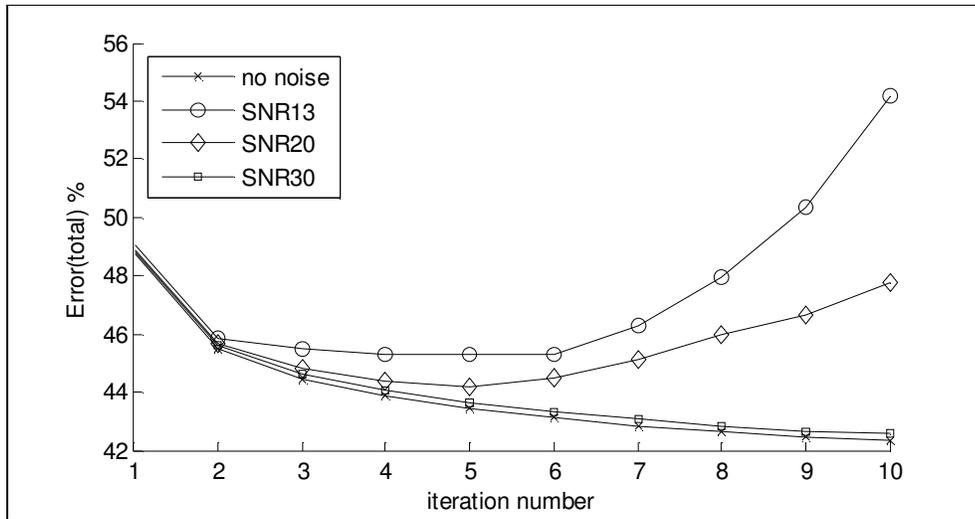


Figure 4.26: Comparison of reconstruction errors for VGB_z algorithm, for different SNR values and noise free data, for Model-1.

4.7.2.2 Results for Model- 2

As explained in Section 4.2.2, there are two square objects with different conductivity values than background, in the imaging slice and there are two

orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

There are two cases for this model. First one is conductive case. For conductive case, the conductivities of squares are 2 S/m. The other case is resistive case. For resistive case, the conductivities of squares are 0.02 S/m. Also there are 10 different positions, which are shown in Table 4.11, used to test the spatial resolution performance of VGB_z reconstruction algorithm by calculating the Full-Width-at-Half-Maximum (FWHM) values.

Table 4.11: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

In Figure 4.27 and Figure 4.28, reconstructed conductivity distributions for both cases with position-4 can be seen. These results are taken without added noise. Also for each position, 4 iterations are used. Since, the algorithm starts to converge at iteration 4.

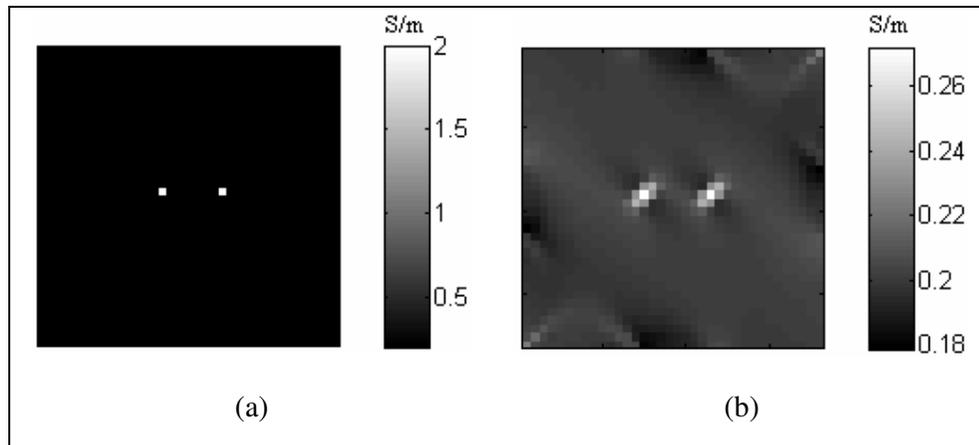


Figure 4.27: Reconstruction by VGB_z algorithm for Model-2, position-4, for conductive case. (a) Real conductivity distribution. (b) Reconstructed conductivity distribution.

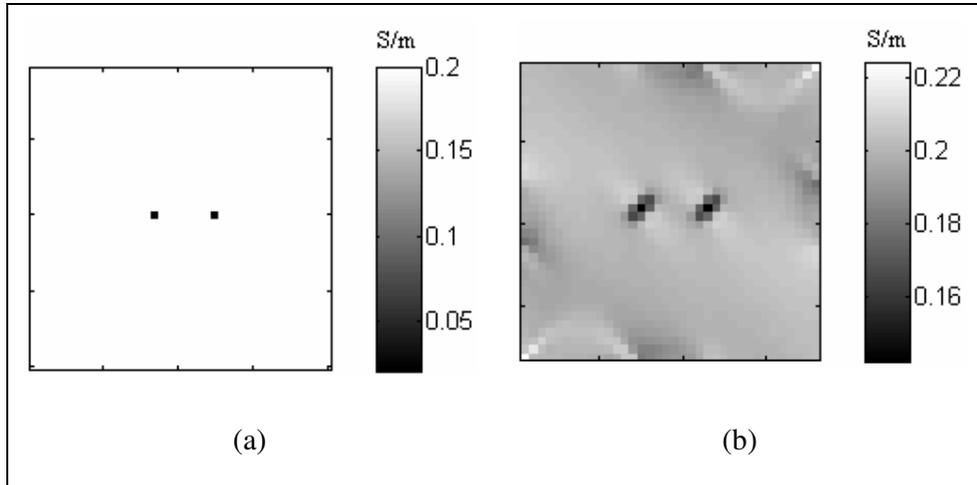


Figure 4.28: Reconstruction by VGB_z algorithm for Model-2, position-4, for resistive case. (a) Real conductivity distribution. (b) Reconstructed conductivity distribution.

The Point Spread Function for the middle row of the imaging slice and position-1 is shown in Figure 4.29.

Calculated FWHM values according to change of gap between two squares for all positions are shown in Figure 4.30.

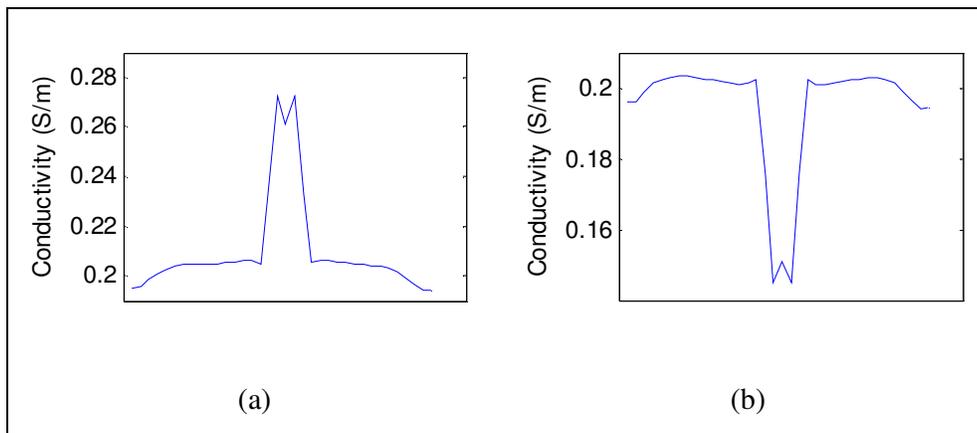


Figure 4.29: PSF for middle row of imaging slice by VGB_z algorithm for Model-2, position-1. (a) For conductive case. (b) For resistive case.

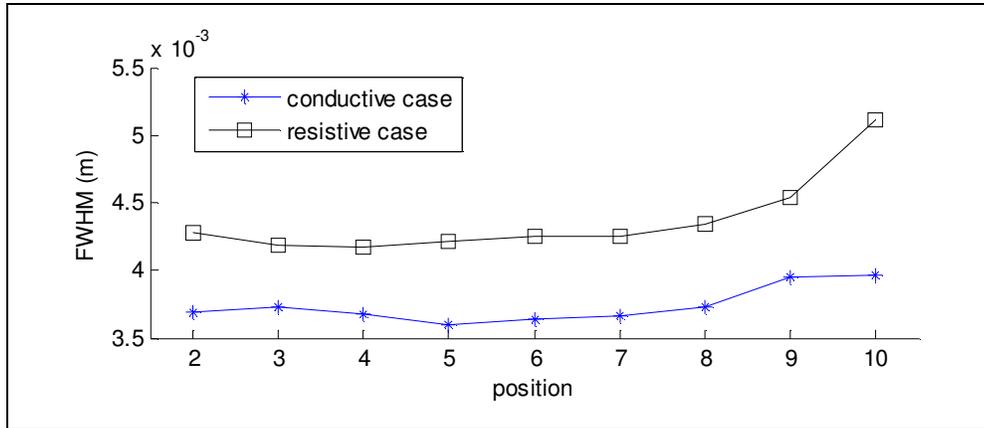


Figure 4.30: FWHM values according to change of gap between two squares for all positions.

As seen from Figure 4.27, the calculated FWHM values for conductive case are smaller than FWHM values for resistive case. For position-1, the gap between two squares is 1 pixel. Only for position-1, two reconstructed squares are merged. The two square objects affect each other. But the other positions the FWHM values are nearly twice of the size of a pixel. This means a pixel affects the other pixels which placed around the target pixel. These results show that VGB_z algorithm's reconstruction performance is not good to reconstruct an object, whose size is equal to one pixel size.

4.7.2.3 Results for Model-3

In Model-3, there are 9 small squares on the diagonal of the imaging slice. The sizes of the squares are equal to the size of a pixel. For this model, there are two cases, which are conductive and resistive, to test the performance of this algorithm. Background conductivity is 0.2 S/m. Conductive square conductivity is 2 S/m and the resistive square conductivity is 0.02 S/m.

Reconstructed conductivity distributions for conductive case and resistive case are shown in Figure 4.31.

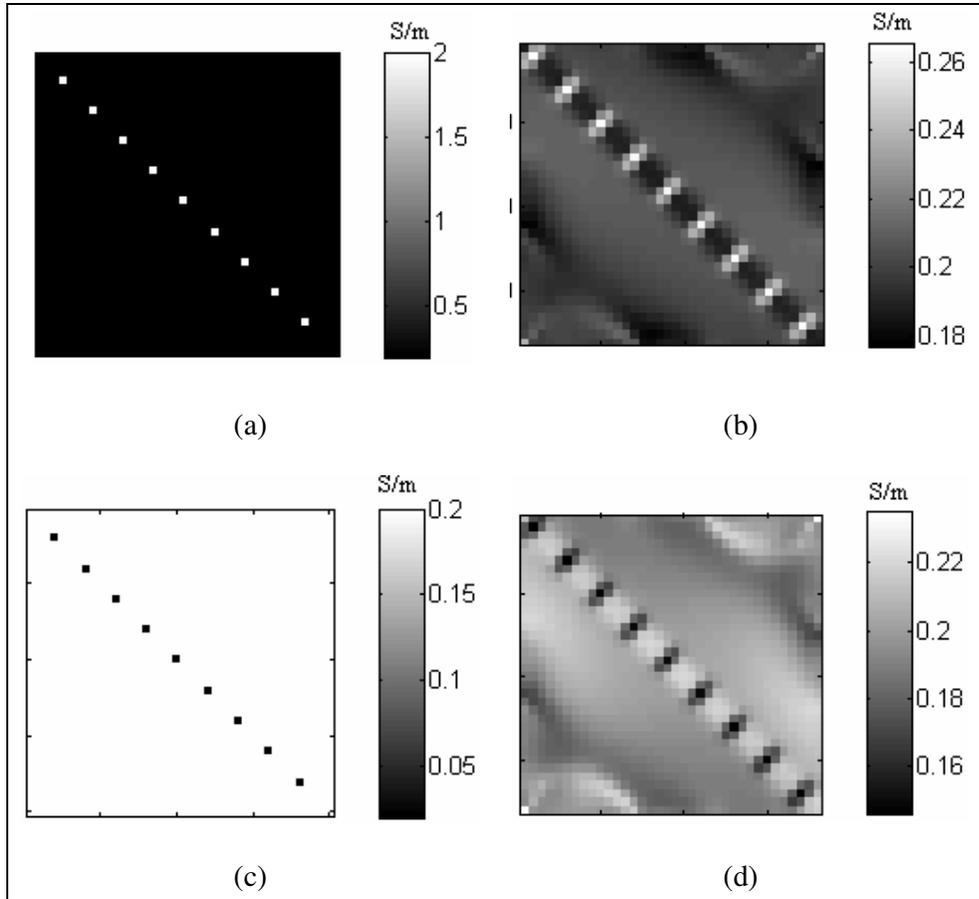


Figure 4.31: Reconstruction by VGB_z algorithm for Model-3, (after iteration 8). (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

The effect of the differentiation at the corner of the imaging slice can be seen. As described in previous section, this effect starts from the electrode edges. Also Figure 4.31 shows that an object, which has a size equal to a pixel size, affects the other pixels which placed around this object.

4.7.2.4 Results for Model-4

In Model-4, there is a square at the center of the imaging slice. The size of the square is 2.25 cm x 2.25 cm. There are two different cases which are conductive case and resistive case. For conductive case, the square is more conductive than background. For resistive case, the square is more resistive than background.

Also there are 10 different contrast levels. This means, for each contrast level square has different conductivity value. These are shown in Table 4.3 in the Section 4.2.4. For conductive case and resistive case, 5 iterations are used for all contrast levels. The algorithm starts to converge at 5th iteration.

For conductive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.4$ S/m (contrast level-2) are shown in Figure 4.32(a) and Figure 4.32(b). For resistive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.1$ S/m (contrast level-2) are shown in Figure 4.32(c) and Figure 4.32(d).

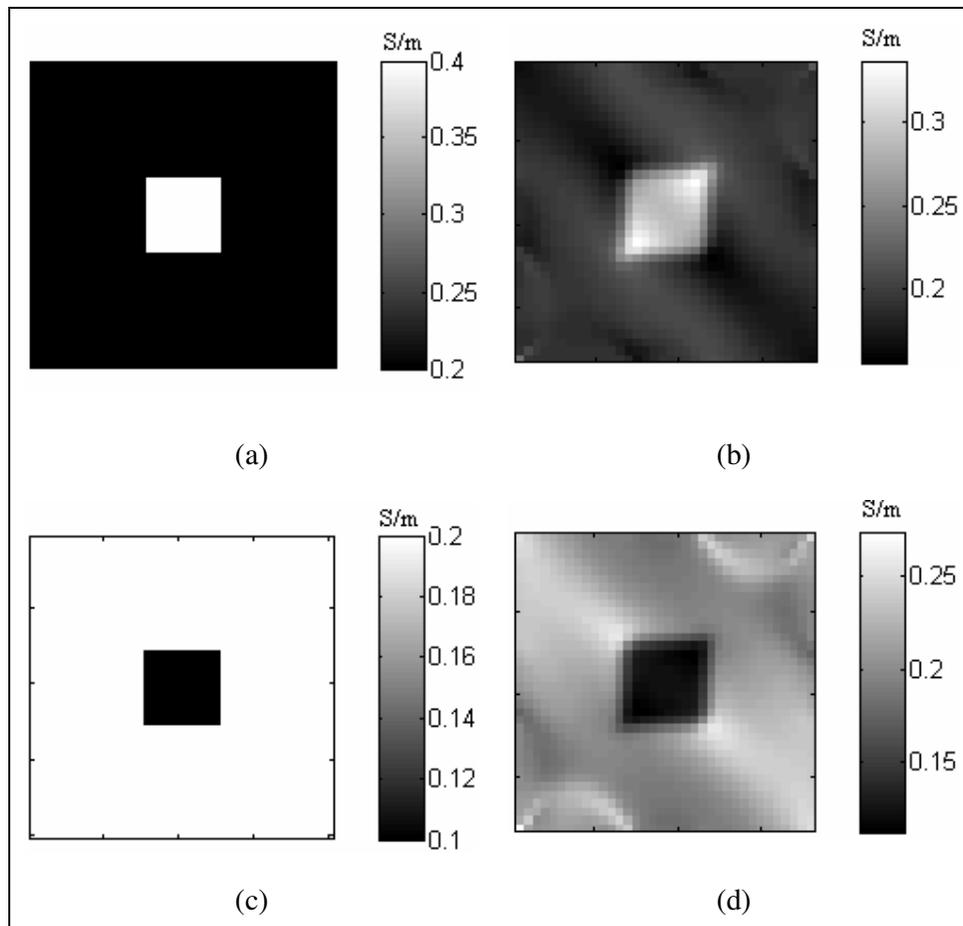


Figure 4.32: Reconstruction by VGB_z algorithm for Model-4, contrast level-2. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case (after iteration 5). (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case (after iteration 5).

For conductive case, the square's conductivity increases when the contrast level increases. But for resistive case the square's conductivity decreases when the contrast level increases. The square error plots according to the change of the contrast level for both cases are shown in Figure 4.33.

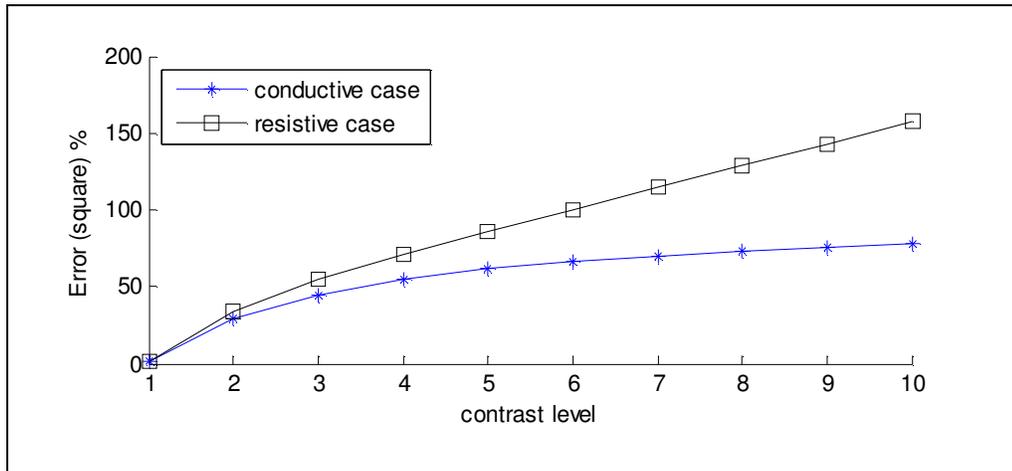


Figure 4.33: Error plots of the square for 10 different contrast levels.

As seen from Figure 4.33, if the difference between square conductivity value and background conductivity value are increased, the error value of the reconstructed square will increase. This means if the contrast is increased, the error will be high.

In Figure 4.34, the plot of the real conductivity values of square and mean of the reconstructed conductivity values of square for each level are shown. As seen from figure that for resistive case, the performance of algorithm is better than the conductive case.

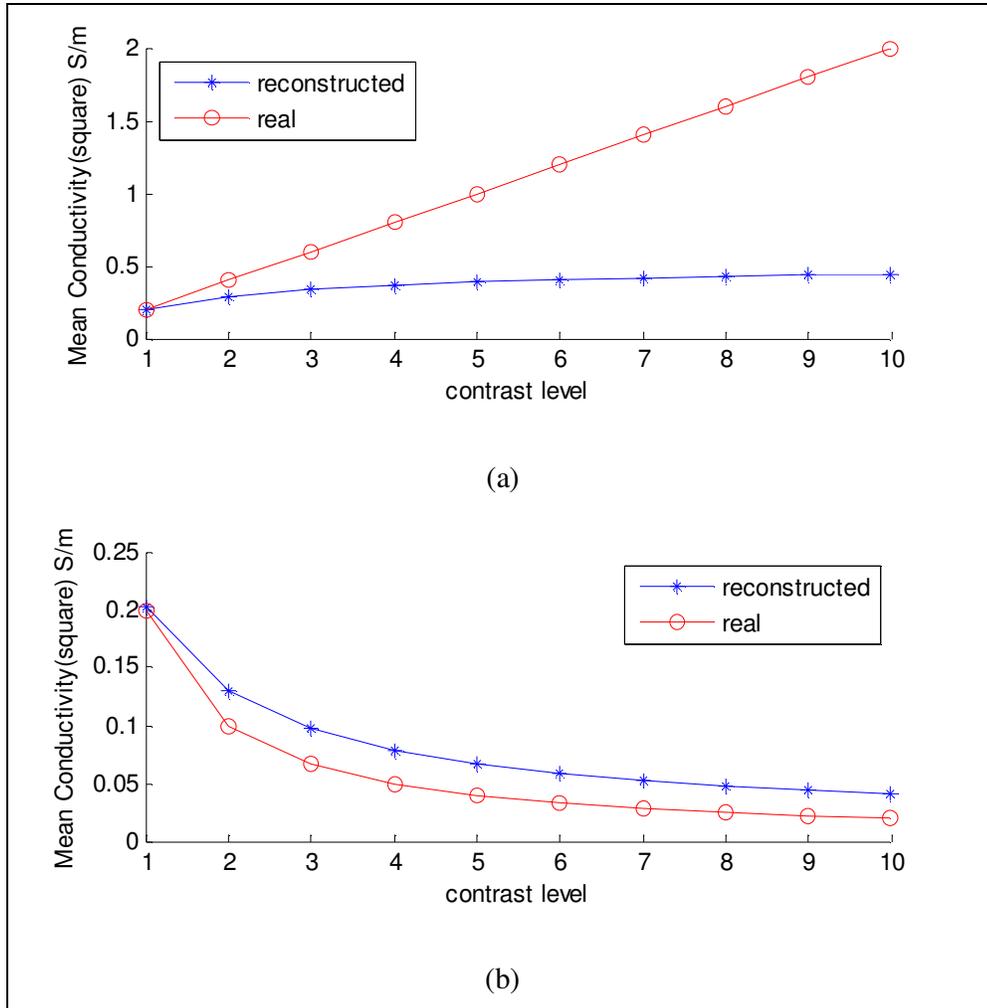


Figure 4.34: Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels. (a) For conductive case. (b) For resistive case.

4.7.2.5 Results for Thorax Model

The model shown in Section 4.2.5 is used to test the performance of VGB_z for noise-free and noisy data. The reconstructed conductivity distributions are shown in Figure 4.35. The total reconstruction errors for noisy and noise free case are shown in Table 4.12.

Table 4.12: Reconstructed error values for thorax model

	<i>Noise-free case</i>	<i>SNR30</i>	<i>SNR20</i>	<i>SNR13</i>
$\epsilon_{\sigma, total}$ (%)	65.52	65.76	65.93	66.67

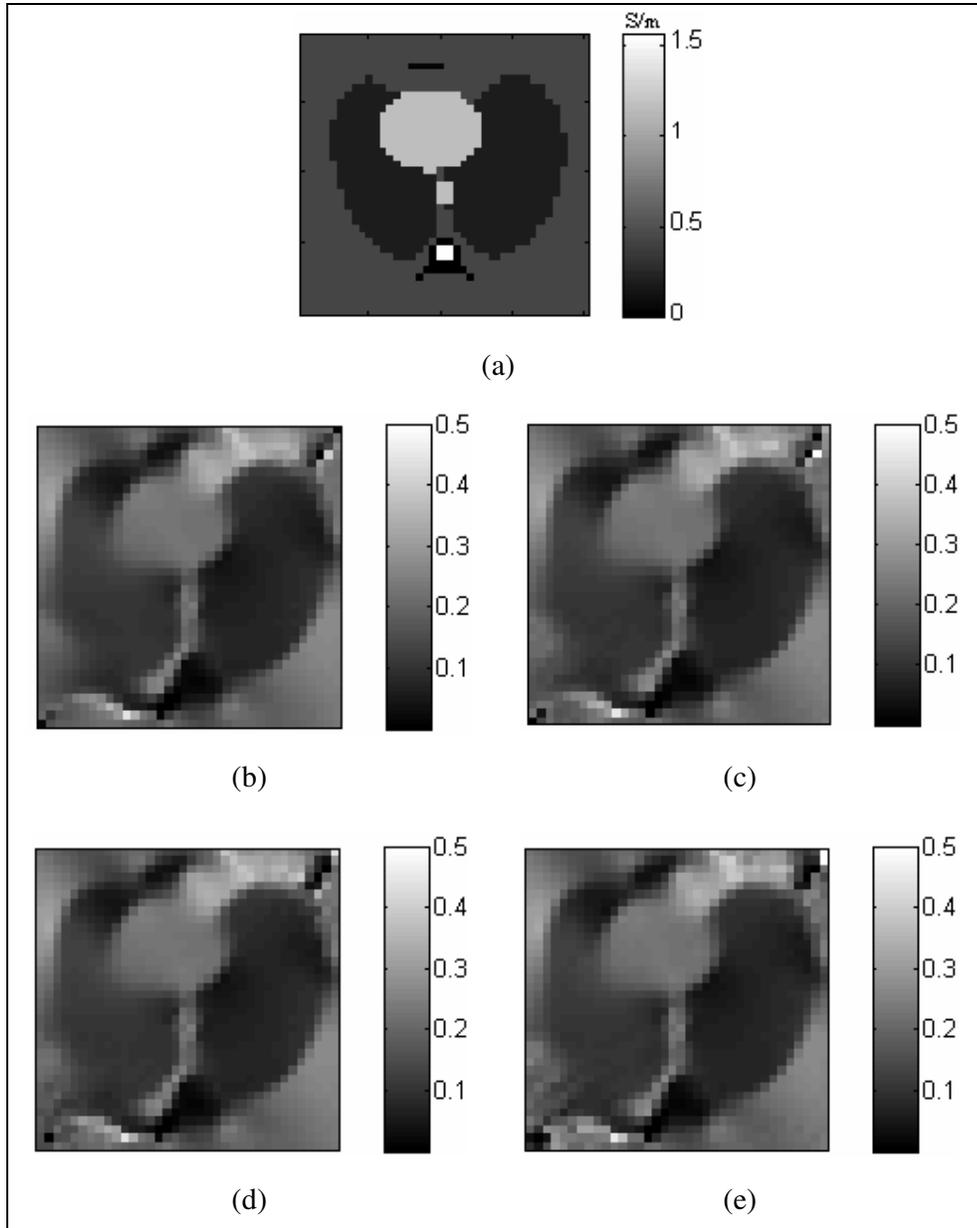


Figure 4.35: Reconstruction by VGB_z algorithm for Thorax Model (all images for iteration 3). (a) Real conductivity distribution. (b) Reconstructed conductivity distribution, noise-free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Figure 4.35, the reconstructed conductivity distributions for noisy data are good as noise free data. But there are artifacts at the corners of image because of the small electrode size.

4.7.3 Reconstruction by Sensitivity Matrix

In this section, the reconstructed conductivity distributions of four different models are examined and discussed by using the Sensitivity Matrix algorithm explained in chapter 3.

As explained in chapter 3, this algorithm uses only the z component of the magnetic flux density (B_z), which is parallel to the main magnetic field. A sensitivity matrix is produced according to the change of conductivity values of each pixel in the imaging slice. It is produced for only the initial conductivity distribution. The pseudo inverse of this matrix is calculated using Singular Value Decomposition (SVD). For models explained in this thesis, the singular values of sensitivity matrix, which are very close to zero, are truncated. Because, their values are very close to zero and gives wrong reconstructed conductivity values. Then, the perturbations in magnetic field measurements are calculated. At the end the conductivity distribution can be calculated by using the perturbation in element conductivity values which are calculated.

4.7.3.1 Results for Model- 1

As explained in Section 4.2.1, there are five different conductivity regions in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

The real conductivity distribution for model-1 is shown in Figure 4.36(a). The reconstructed conductivity distributions for noise free data are shown in Figure 4.36(b).

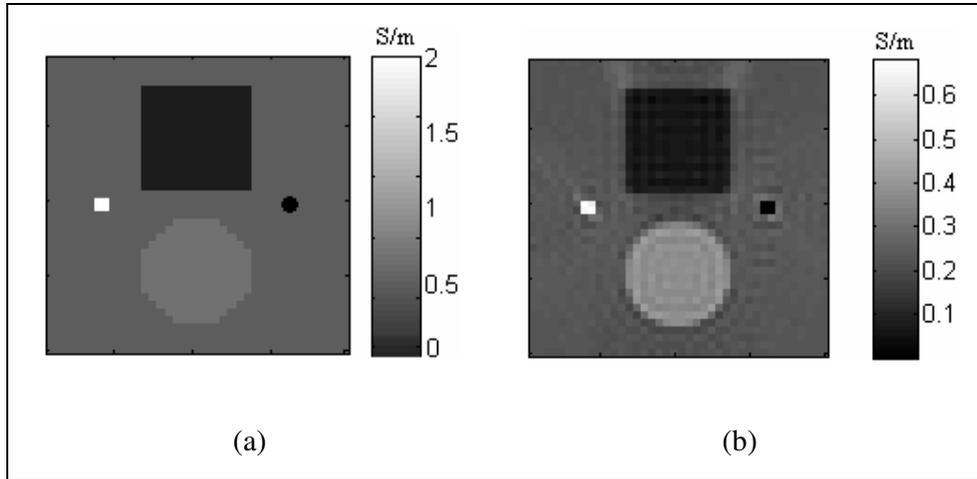


Figure 4.36: Reconstruction by Sensitivity Matrix for Model-1. (a) Real conductivity distribution. (b) Reconstructed conductivity distribution for noise free data.

For noise free data, the reconstructed conductivity error values of all regions are shown in Table 4.13. As seen from Table 4.13, the small circle, which has very small conductivity value than the other regions, is reconstructed with a high reconstruction error.

Table 4.13: Reconstructed error values of all regions, noise free data

	Noise free case
$\mathcal{E}_{\sigma, total} (\%)$	31.39
$\mathcal{E}_{\sigma, background} (\%)$	17.76
$\mathcal{E}_{\sigma, bigsquare} (\%)$	33.73
$\mathcal{E}_{\sigma, bigcircle} (\%)$	8.59
$\mathcal{E}_{\sigma, smallsquare} (\%)$	66.81
$\mathcal{E}_{\sigma, smallcircle} (\%)$	94.35

This reconstruction algorithm reconstructs a conductivity distribution with low total error.

Then this reconstruction algorithm tested with different noise levels: SNR 30, SNR 20 and SNR 13. The reconstructed conductivity distributions are shown in Figure 4.37.

The reconstructed conductivity errors of all regions are shown in Table 4.14.

Table 4.14: Reconstructed error values of all regions for noise free and noisy data

	<i>Noise free case</i>	<i>SNR 30</i>	<i>SNR 20</i>	<i>SNR 13</i>
$\mathcal{E}_{\sigma, total} (\%)$	31.39	32.41	33.43	35.93
$\mathcal{E}_{\sigma, background} (\%)$	17.76	20.57	22.93	29.43
$\mathcal{E}_{\sigma, bigsquare} (\%)$	33.73	37.92	42.53	51.71
$\mathcal{E}_{\sigma, bigcircle} (\%)$	8.59	9.78	9.88	11.59
$\mathcal{E}_{\sigma, smallsquare} (\%)$	66.81	66.62	66.77	65.12
$\mathcal{E}_{\sigma, smallcircle} (\%)$	94.35	94.30	94.24	94.16

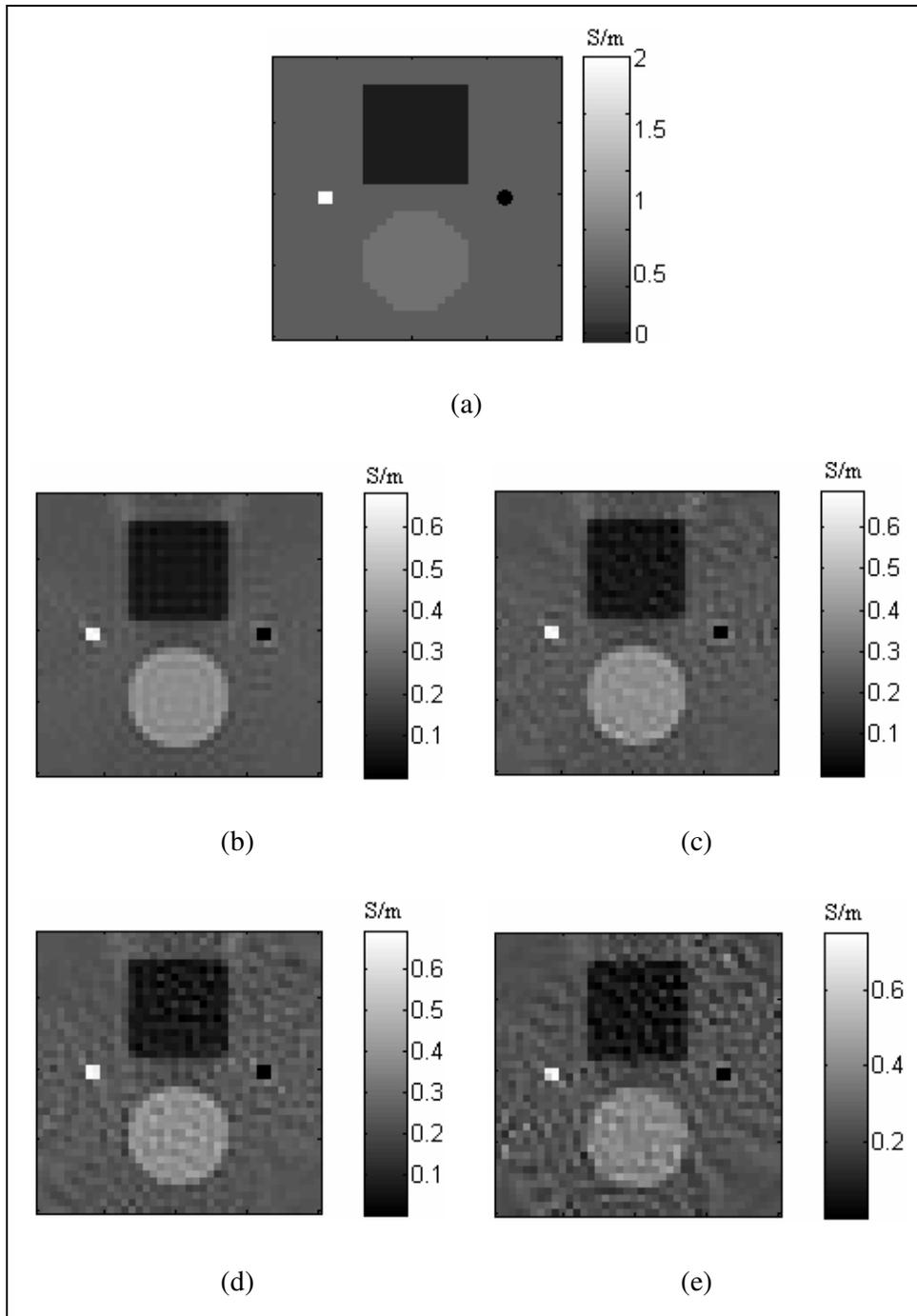


Figure 4.37: Reconstruction by Sensitivity Matrix for Model-1. (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution, noise free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Table 4.14, if we decrease the SNR value, the total error value will increase. In Figure 4.37, the effect of SNR can be seen. This algorithm reconstructs conductivity distributions for noisy case good as noise free case. If we increase the number of current injection patterns and amount of current, this algorithm will give good reconstructed conductivity results for lowest SNR values.

The mean reconstructed conductivity values and the real conductivity values of regions are shown in Table 4.15.

Table 4.15: Mean reconstructed conductivity values of all regions for noise free and noisy data

	Real Conductivity	Mean Reconstructed Conductivity			
		<i>Noise free</i>	<i>SNR 30</i>	<i>SNR 20</i>	<i>SNR 13</i>
$\sigma_{background}$ (S/m)	0.2	0.2312	0.2334	0.2357	0.2390
$\sigma_{bigsquare}$ (S/m)	0.1	0.0702	0.0709	0.0720	0.0758
$\sigma_{bigcircle}$ (S/m)	0.4	0.3768	0.3799	0.3851	0.3890
$\sigma_{smallsquare}$ (S/m)	2	0.6638	0.6677	0.6646	0.6987
$\sigma_{smallcircle}$ (S/m)	0.02	0.0011	0.0011	0.0012	0.0012

Sensitivity matrix is produced for an initial conductivity distribution which is uniform. The Sensitivity Matrix algorithm reconstructs conductivity distribution of the imaging slice. This reconstructed conductivity distribution can be used as initial conductivity distribution to produce a new sensitivity matrix. This produced sensitivity matrix can be used during reconstruction.

In Figure 4.38, the reconstructed conductivity distributions for two different cases are shown.

For case-1, the conductivity distribution is reconstructed by using sensitivity matrix which is produced from a uniform conductivity distribution.

For case-2, the conductivity distribution is reconstructed by using sensitivity matrix which is produced from a reconstructed conductivity distribution.

The mean reconstructed conductivity values of all regions for two cases are shown in Table 4.16.

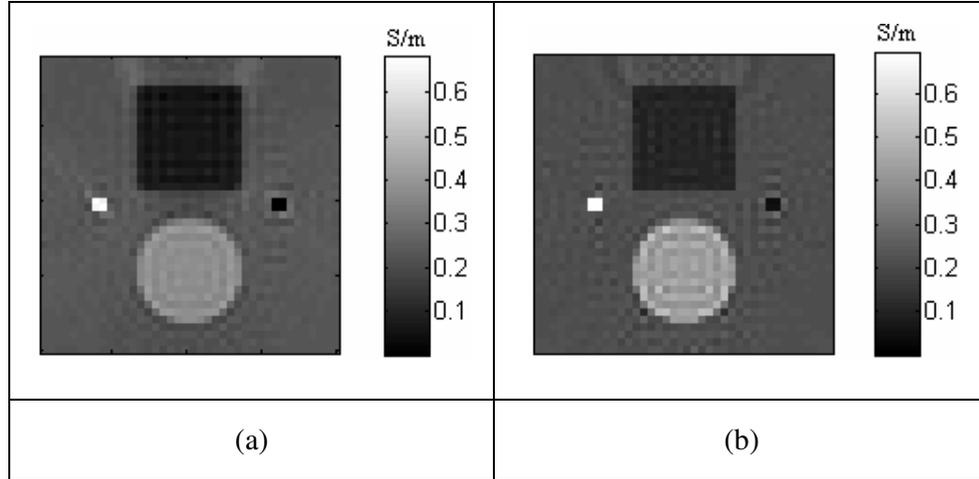


Figure 4.38: Reconstructed conductivity distributions for two cases (for noise-free data). (a) Case-1, reconstruction by sensitivity matrix produced from uniform conductivity distribution. (b) Case-2, reconstruction by sensitivity matrix produced from reconstructed conductivity distribution.

Table 4.16: Mean reconstructed conductivity values of all regions for two cases (for noise free data).

	Real Conductivity	Mean Reconstructed Conductivity	
		<i>Case-1</i>	<i>Case-2</i>
$\sigma_{background}$ (S/m)	0.2	0.2312	0.2289
$\sigma_{bigsquare}$ (S/m)	0.1	0.0702	0.0798
$\sigma_{bigcircle}$ (S/m)	0.4	0.3768	0.4289
$\sigma_{smallsquare}$ (S/m)	2	0.6638	0.6824
$\sigma_{smallcircle}$ (S/m)	0.02	0.0011	0.0022

As seen from Figure 4.38 and Table 4.16, reconstructed conductivity values for two cases are nearly same. For each case, the algorithm needs high computation time during sensitivity matrix production. If we choose initial conductivity

distribution sufficiently near to the true conductivity values, new sensitivity matrix production will be eliminated.

In this study, the sensitivity matrix is calculated for only the initial conductivity distribution. Initial conductivity distribution is assumed $\sigma_0=0.2$ S/m. Then this matrix is used during conductivity distribution reconstruction.

4.7.3.2 Results for Model-2

As explained in Section 4.2.2, there are two square objects with different conductivity values than background, in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns. There are two cases for this model. First one is conductive case. For conductive case, the conductivities of squares are 2 S/m. The other case is resistive case. For resistive case, the conductivities of squares are 0.02 S/m. Also there are 10 different positions, which are shown in Table 4.17, used to test the spatial resolution performance of Sensitivity Matrix reconstruction algorithm by calculating the Full-Width-at-Half-Maximum (FWHM) values.

Table 4.17: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

In Figure 4.39, you can see reconstructed conductivity distribution for conductive case and resistive case. These results are taken for noise free data.

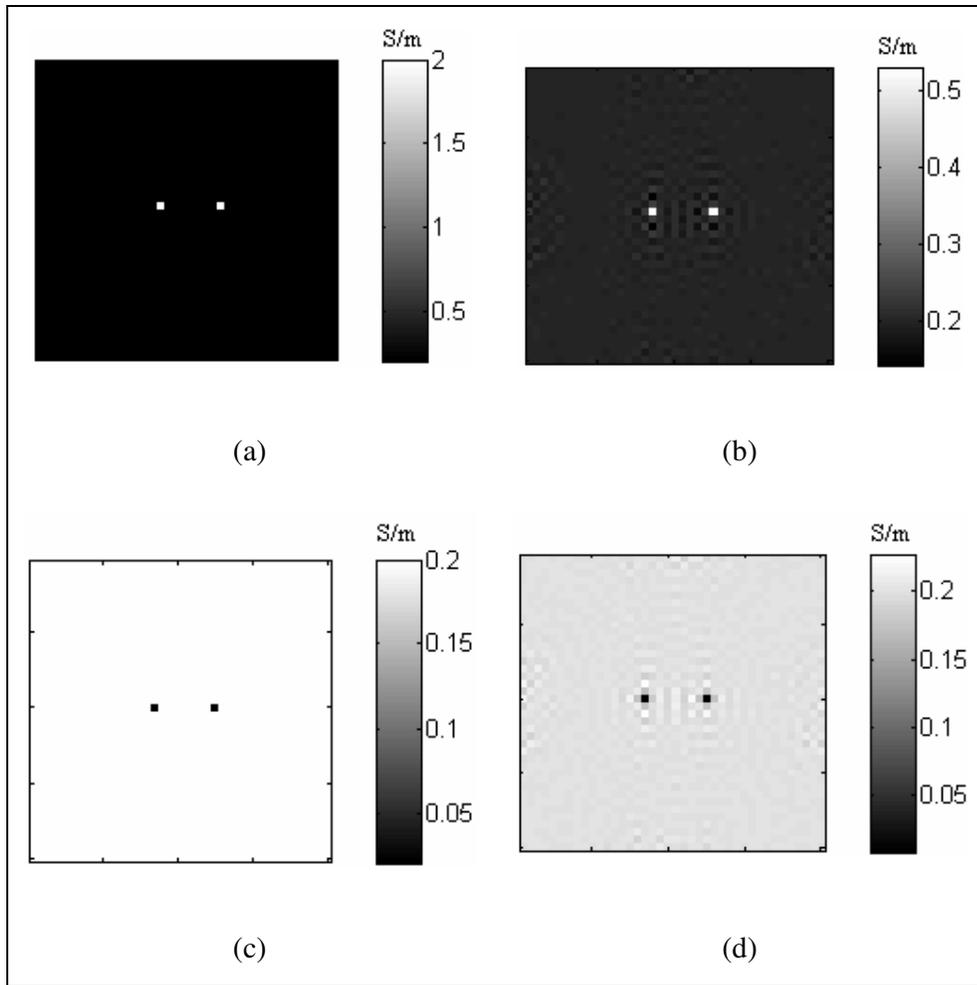


Figure 4.39: Reconstruction by Sensitivity Matrix for Model-2, position-4. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

For conductive and resistive case, the Point Spread Function for the middle row of the imaging slice for position-1 is shown in Figure 4.40 and calculated FWHM values according to change of gap between two squares for all positions are shown in Figure 4.41.

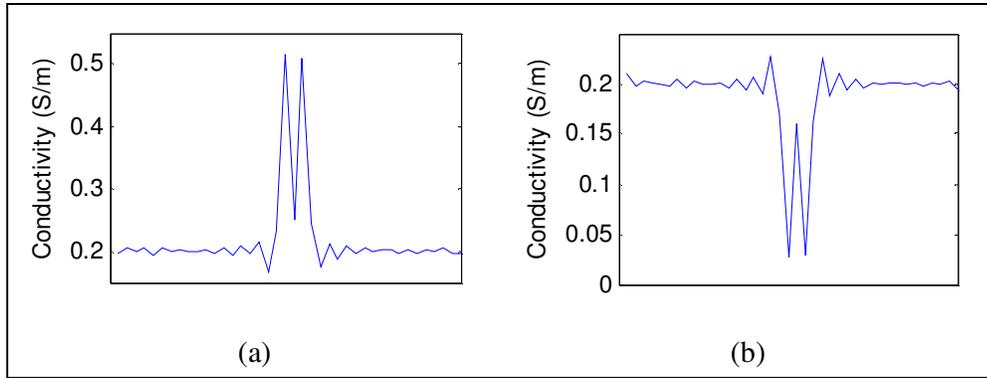


Figure 4.40: PSF for middle row of imaging slice by Sensitivity Matrix algorithm for Model-2, position-1. (a) Conductive case. (b) Resistive case.

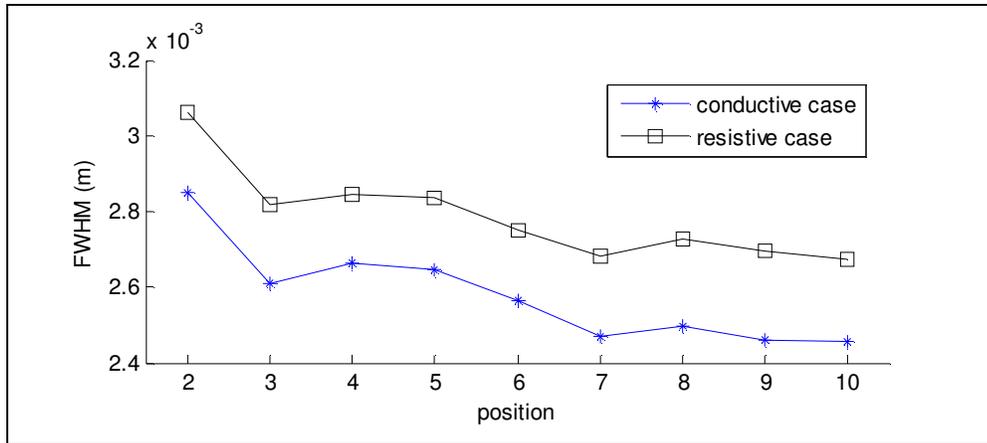


Figure 4.41: FWHM values according to change of gap between two squares for all positions.

As seen from Figure 4.41, the reconstructed FWHM values are very close to the pixel size especially for conductive case. For position-1, the gap between two squares is 1 pixel. Only for position-1, two reconstructed squares are merged. The two square objects affect each other. But the other positions the FWHM values are very close to the size of a pixel. These results show that Sensitivity Matrix algorithm successfully reconstruct an object which is equal to a pixel size. It has a good spatial resolution performance for conductive and resistive cases.

4.7.3.3 Results for Model-3

In Model-3, there are 9 small squares on the diagonal of the imaging slice. The sizes of the squares are equal to the size of a pixel. For this model, there are two cases which are conductive and resistive. Background conductivity is 0.2 S/m. Conductive square conductivity is 2 S/m and the resistive square conductivity is 0.02 S/m.

Reconstructed conductivity distributions for conductive case and resistive case are shown in Figure 4.42. These results are taken for noise free data.

As seen in Figure 4.42, the Sensitivity Matrix algorithm successfully reconstructs conductivity distribution both for resistive case and conductive case. The effect of the one pixel sized square objects to the around neighboring pixels are very small. This is an advantage for this algorithm.

The mean reconstructed conductivity values and the real conductivity values of squares are shown in Table 4.18.

Table 4.18: Mean Conductivity values of squares

Mean Conductivity(Conductive case)		Mean Conductivity(Resistive case)	
$\sigma_{real,square}$	$\sigma_{reconstructed,square}$	$\sigma_{real,square}$	$\sigma_{reconstructed,square}$
2	0.5146	0.02	0.0136

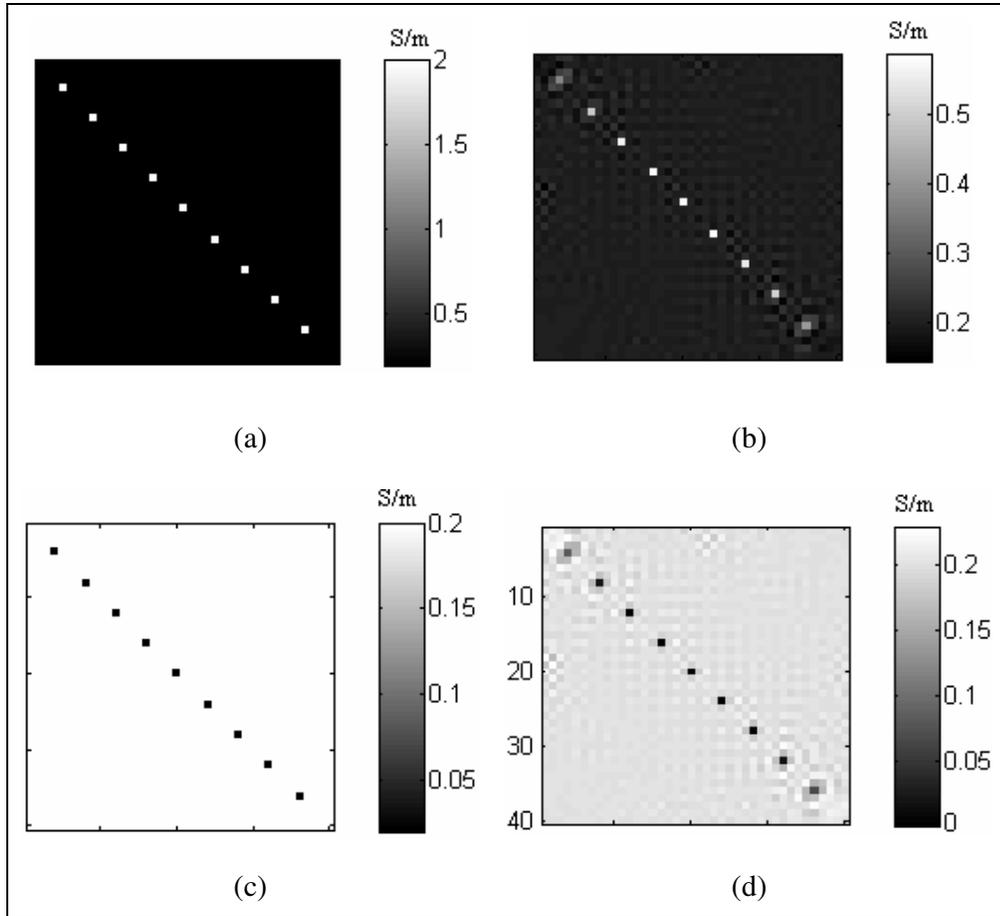


Figure 4.42: Reconstruction by Sensitivity Matrix for Model-3. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

4.7.3.4 Results for Model-4

In Model-4, there is a square at the center of the imaging slice. The size of the square is 2.25 cm x 2.25 cm. There are two different cases which are conductive case and resistive case. For conductive case, the square is more conductive than background. For resistive case, the square is more resistive than background. Also there are 10 different contrast levels. This means, for each level square has different conductivity value. These are shown in Table 4.3 in the Section 4.2.4.

For conductive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.4$ S/m (contrast level-2) are shown in Figure 4.43(a) and Figure 4.43(b).

For resistive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.1$ S/m (contrast level-2) are shown in Figure 4.43(c) and Figure 4.43(d).

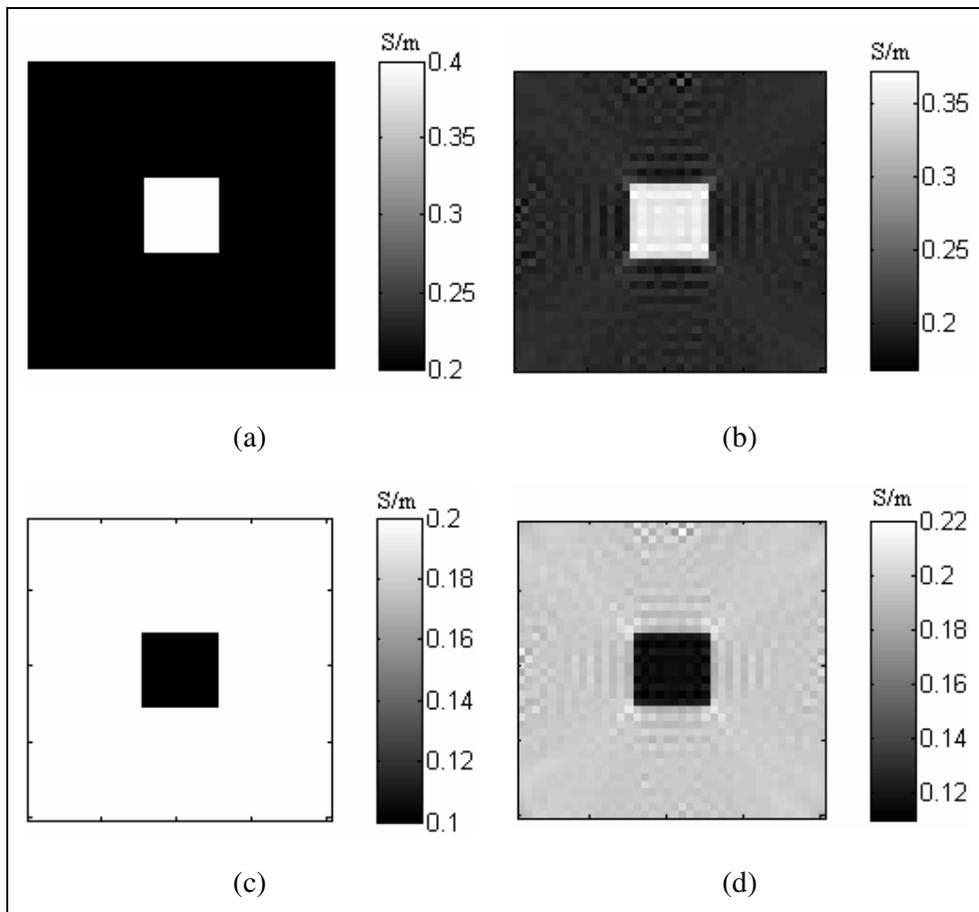


Figure 4.43: Reconstruction by Sensitivity Matrix for Model-4, contrast level-2. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

This algorithm successfully reconstructs conductivity distribution for Model-4 and for all contrast levels.

For the conductive case, the square's conductivity increases when the contrast level increases. But for resistive case the square's conductivity decreases when the contrast level increases. The square reconstruction error values according to the change of the contrast level for both cases are shown in Figure 4.44.

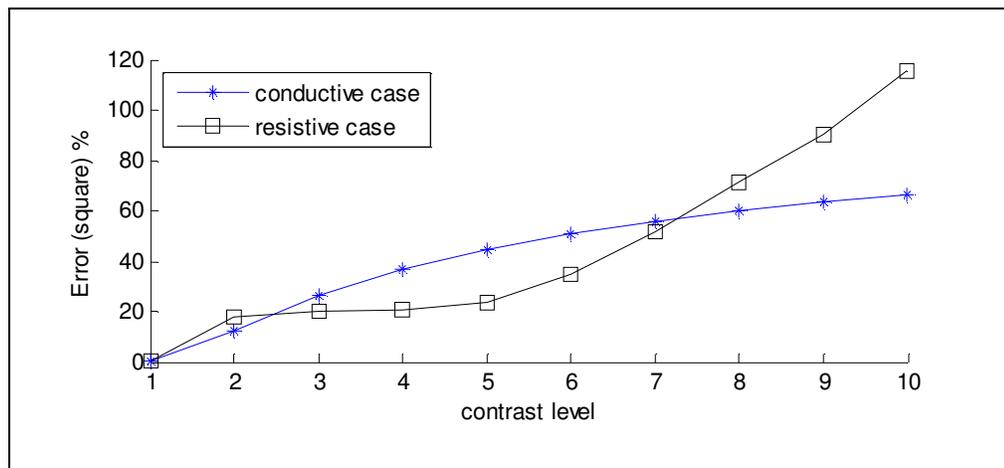


Figure 4.44: Error plots of the square for 10 different contrast levels.

As seen from Figure 4.44, if we increase the difference between square conductivity value and background conductivity value, the error value of the reconstructed square will increase. Especially for resistive case the error values are very high for higher contrast levels. This means if the contrast is high, the error will be high. For conductive case, this algorithm has better reconstruction performance than resistive case for higher contrast levels.

In Figure 4.45, the plot of the real conductivity values and mean of the reconstructed conductivity values of square for each contrast level can be seen.

These results show that Sensitivity Matrix algorithm successfully reconstruct conductivity distributions for both cases. But the error values are high for high conductivity value differences between square and background.

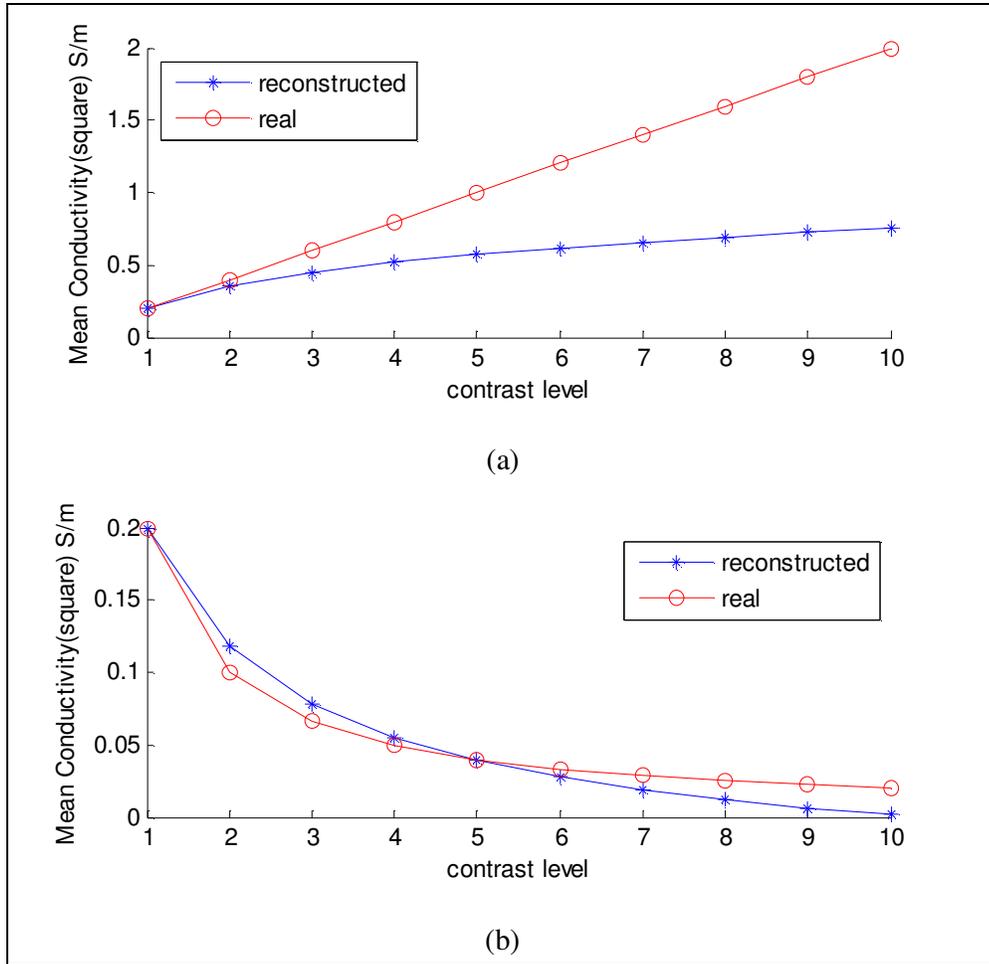


Figure 4.45: Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels. (a) For conductive case. (b) For resistive case.

4.7.3.5 Results for Thorax Model

The model shown in Section 4.2.5 is used to test the performance of Sensitivity matrix algorithm for noise-free and noisy data. The reconstructed conductivity distributions are shown in Figure 4.46. The total reconstruction errors for noisy and noise free case are shown in Table 4.19.

Table 4.19: Reconstructed error values for thorax model.

	Noise-free case	SNR30	SNR20	SNR13
$\epsilon_{\sigma, total}$ (%)	57.82	57.90	57.98	58.20

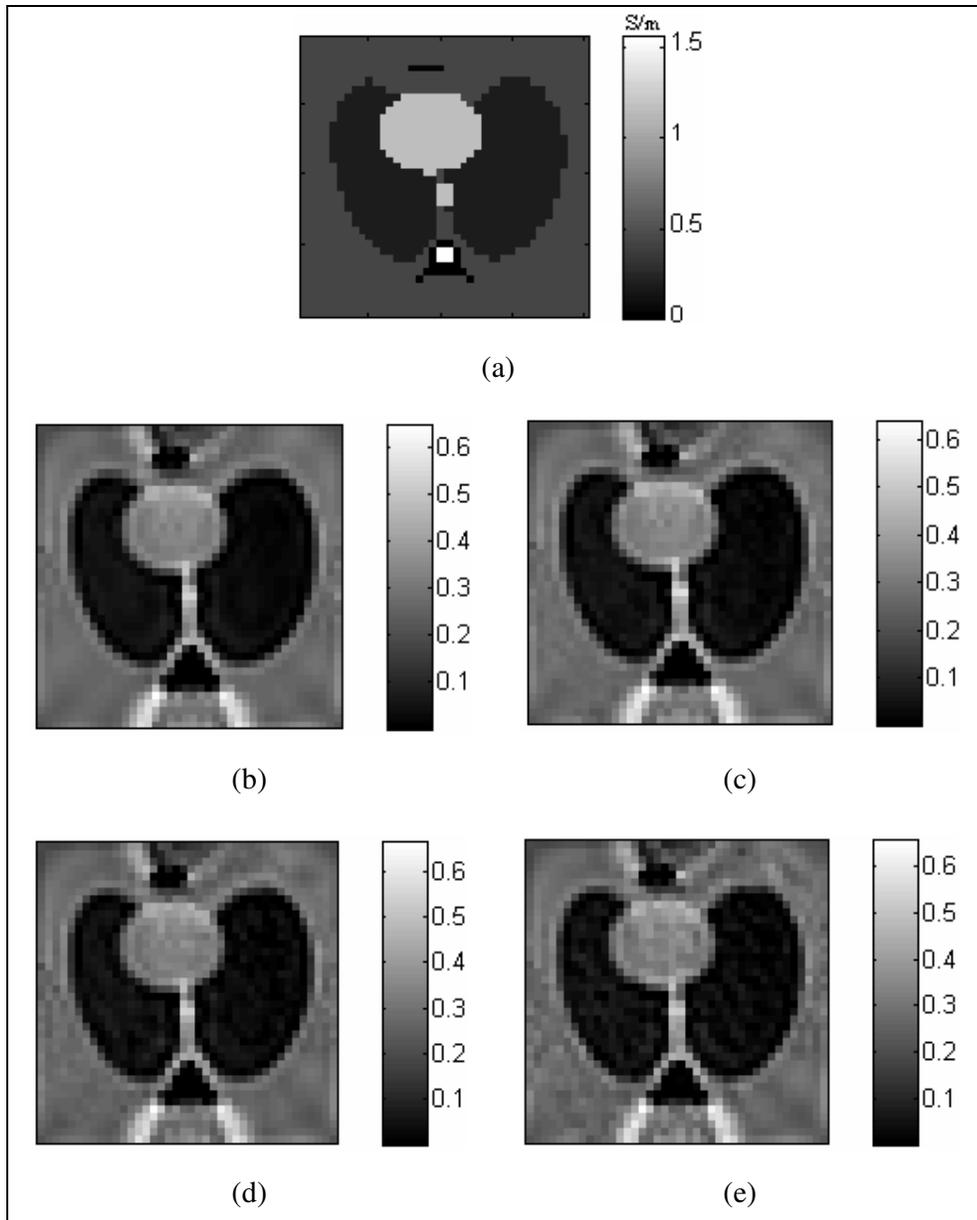


Figure 4.46: Reconstruction by Sensitivity Matrix algorithm for Thorax Model. (a) Real conductivity distribution. (b) Reconstructed conductivity distribution, noise-free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Figure 4.46, the reconstructed conductivity distributions for noisy data are good as noise free data. The effect of noise to this algorithm is very small.

4.7.4 Reconstruction by Algebraic Reconstruction Algorithm

In this section, the reconstructed conductivity distributions of four different models is examined and discussed by using the Algebraic Reconstruction algorithm explained in chapter 3.

As explained in chapter 3, this iterative algorithm uses only the z component of the magnetic flux density (B_z), which is parallel to the main magnetic field. Same as Harmonic B_z algorithm, Algebraic Reconstruction algorithm uses Laplacian of B_z . Firstly, the B_z is taken and Laplacian of B_z is calculated. Then, the x-gradient and y-gradient of the current density for uniform case are calculated. At the end, the conductivity distribution can be calculated using the calculated logarithmic resistivity values.

4.7.4.1 Results for Model- 1

As explained in Section 4.2.1, there are five different conductivity regions in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

The real conductivity distribution for model-1 is shown in Figure 4.47(a). The reconstructed conductivity distributions after iteration 2 and after iteration 8 are shown in Figure 4.47(b) and Figure 4.47(c). These results are taken for noise free data.

After the iteration 2, the algorithm starts to be stable. This means after iteration 2 the change in the reconstructed conductivity distribution is very small. It is easily seen from the total error figure which is shown in Figure 4.48(a). All regions' error characteristics are shown in Figure 4.48. The reconstructed error values of all regions, for iteration 2 are shown in Table 4.20.

As seen from Figure 4.47 and Figure 4.48, Algebraic Reconstruction algorithm reconstructs conductivity successfully without added noise.

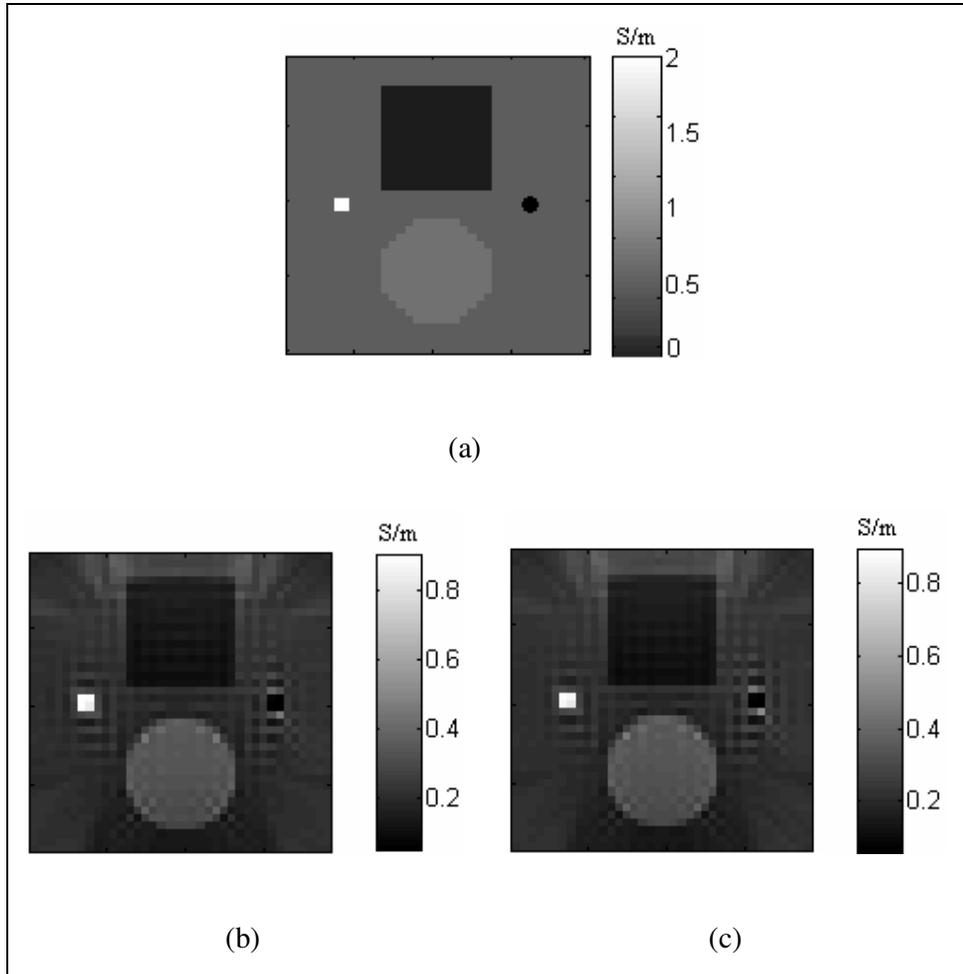


Figure 4.47: Reconstruction by Algebraic Reconstruction algorithm for Model-1 (noise free data). (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution after iteration 2. (c) Reconstructed conductivity distribution after iteration 8.

Table 4.20: Reconstructed error values after 2 iterations

$\mathcal{E}_{\sigma, total}$ (%)	$\mathcal{E}_{\sigma, background}$ (%)	$\mathcal{E}_{\sigma, bigsquare}$ (%)	$\mathcal{E}_{\sigma, bigcircle}$ (%)	$\mathcal{E}_{\sigma, smallsquare}$ (%)	$\mathcal{E}_{\sigma, smallcircle}$ (%)
27.05	22.72	43.66	18.27	55.96	208.23

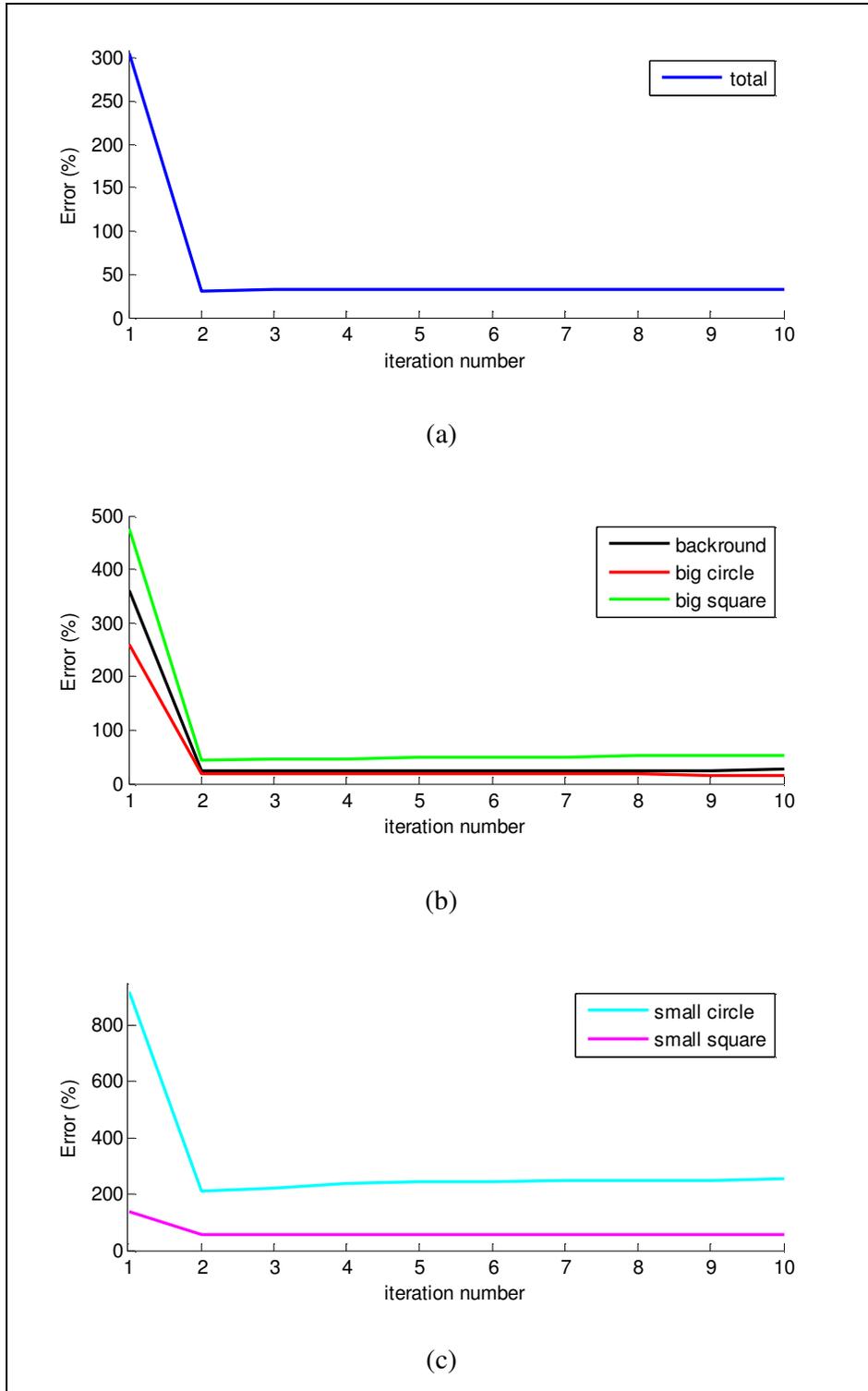


Figure 4.48: Reconstructed conductivity errors by Algebraic Reconstruction algorithm for Model-1 (noise free data). (a) Total error. (b) Background error, Big Circle error, Big Square error. (c) Small Circle error, Small Square error.

As seen from Figure 4.48, the error values are very high after the first iteration. But after iteration 2, the error values become stable near an error value.

Then this reconstruction algorithm tested with different noise levels, SNR 30, SNR 20 and SNR 13. The total error plot comparison is shown in Figure 4.49.

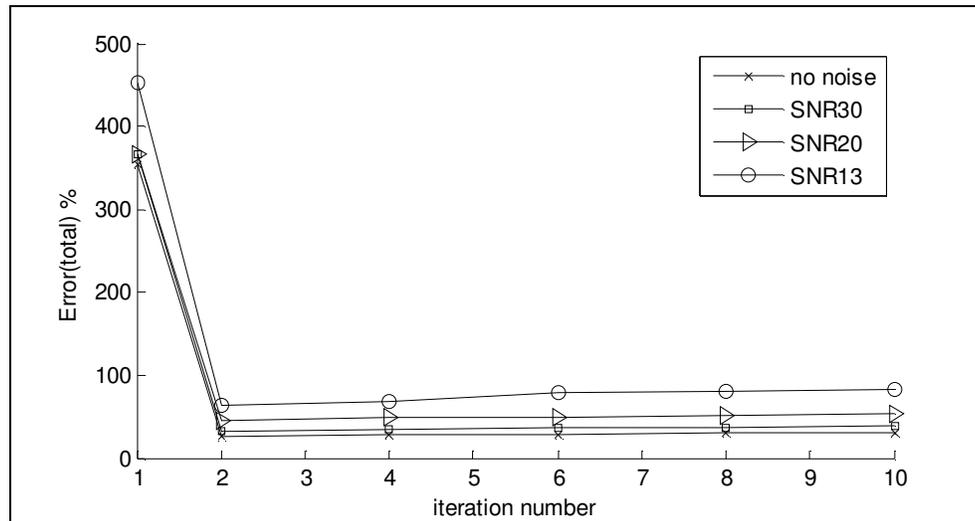


Figure 4.49: Comparison of reconstruction errors for Algebraic Reconstruction algorithm, for different SNR values and no noise case, for Model-1.

In Figure 4.49 it is seen that if SNR equal to 30, the reconstruction error is very near to error for noise free data.

The reconstructed conductivity distributions for noisy data and noise free data are shown in Figure 4.50. As seen from Figure 4.50 the reconstructed conductivity distribution are not good as noise free data. For lowest SNR values, this result is expected. Because Algebraic Reconstruction algorithm uses Laplacian of B_z , which means twice differentiation of both B_z and noise. For low SNR values, there are some dominant conductive pixels in the reconstructed conductivity distribution. According to these reasons it is seen that SNR must be high to take good results. Also increasing the number of current injection patterns and amount of current will give good reconstructed conductivity results with low SNR values.

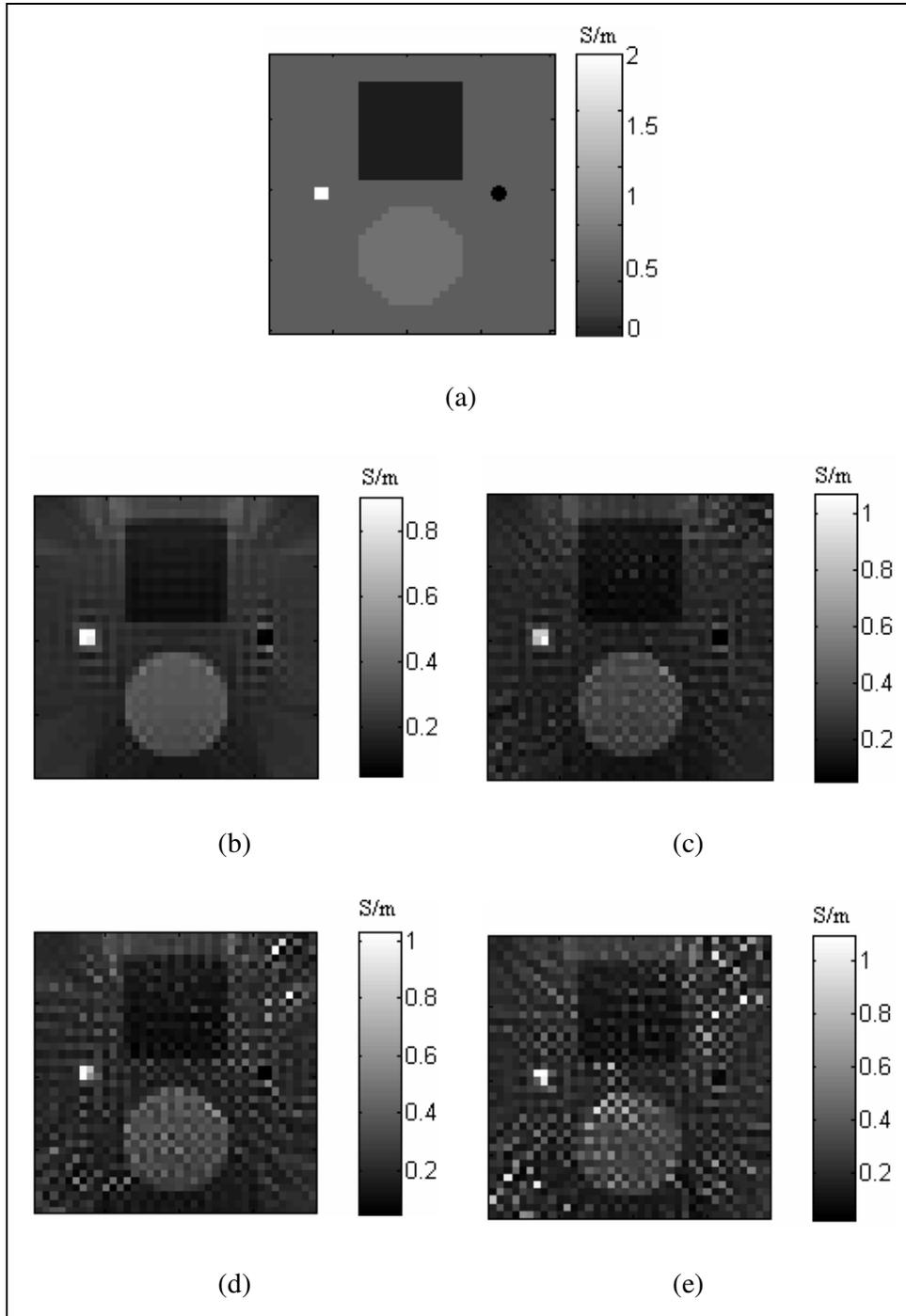


Figure 4.50: Reconstruction by Algebraic Reconstruction algorithm for Model-1 (all images for iteration 2). (a) Real conductivity distribution for Model-1. (b) Reconstructed conductivity distribution noise free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

4.7.4.2 Results for Model-2

As explained in Section 4.2.2, there are two square objects with different conductivity values than background, in the imaging slice and there are two orthogonal current injection patterns. 20 mA current is injected for both injection patterns. There are two cases for this model. First one is conductive case. For conductive case, the conductivities of squares are 2 S/m. The other case is resistive case. For resistive case, the conductivities of squares are 0.02 S/m. Also there are 10 different positions, which are shown in Table 4.21, used to test the spatial resolution performance of Algebraic Reconstruction algorithm by calculating the Full-Width-at-Half-Maximum (FWHM) values.

Table 4.21: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

In Figure 4.51, the reconstructed conductivity distribution for both cases with position-4 can be seen. These results are taken without added noise. Also for each position, 2 iterations are used. Since, the algorithm becomes stable after 2nd iteration.

Also, the Point Spread Function for the middle row of the imaging slice for position-1 is shown in Figure 4.52.

Calculated FWHM values according to change of gap between two squares for all positions are shown in Figure 4.53.

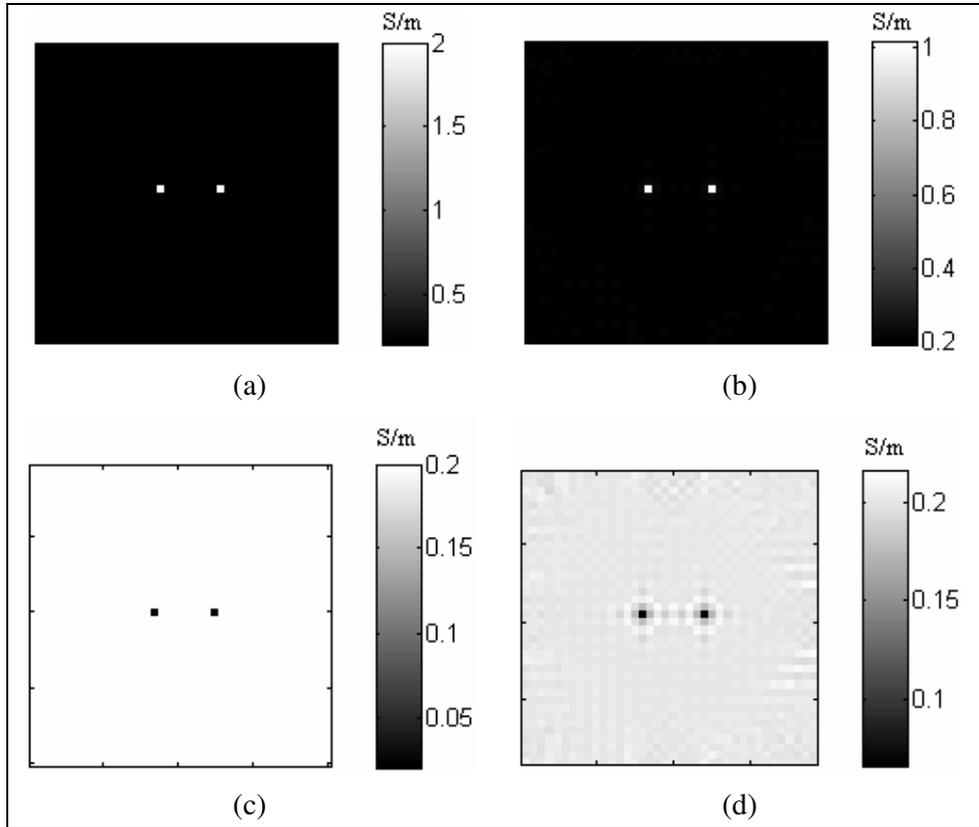


Figure 4.51: Reconstruction by Algebraic Reconstruction algorithm for Model-2, position-4. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

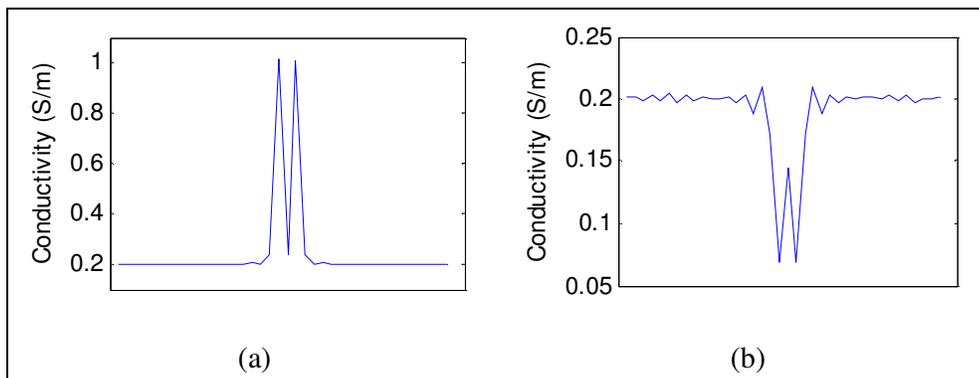


Figure 4.52: PSF for middle row of imaging slice by Algebraic Reconstruction algorithm for Model-2, position-1. (a) For conductive case. (b) For resistive case.

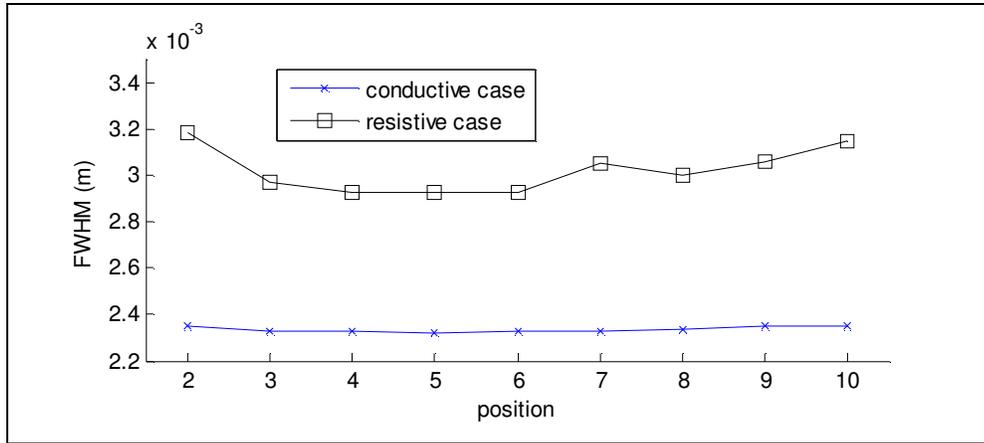


Figure 4.53: FWHM values according to change of gap between two squares for all positions.

As seen from Figure 4.53, the reconstructed FWHM values are very close to the pixel size especially for conductive case. As seen from Figure 4.52, for position-1, the gap between two squares is 1 pixel. Only for position-1, two reconstructed squares are merged for resistive case. The two square objects affect each other. These results show that Algebraic Reconstruction algorithm successfully reconstruct an object whose size is equal to one pixel size. It has a good spatial resolution performance.

4.7.4.3 Results for Model-3

In Model-3, there are 9 small squares on the diagonal of the imaging slice. The sizes of the squares are equal to the size of one pixel. To test the performance of this algorithm, conductive and resistive cases, are used. Background conductivity is 0.2 S/m. Conductive square conductivity is 2 S/m and the resistive square conductivity is 0.02 S/m.

Reconstructed conductivity distributions for conductive case and resistive case are shown in Figure 4.54.

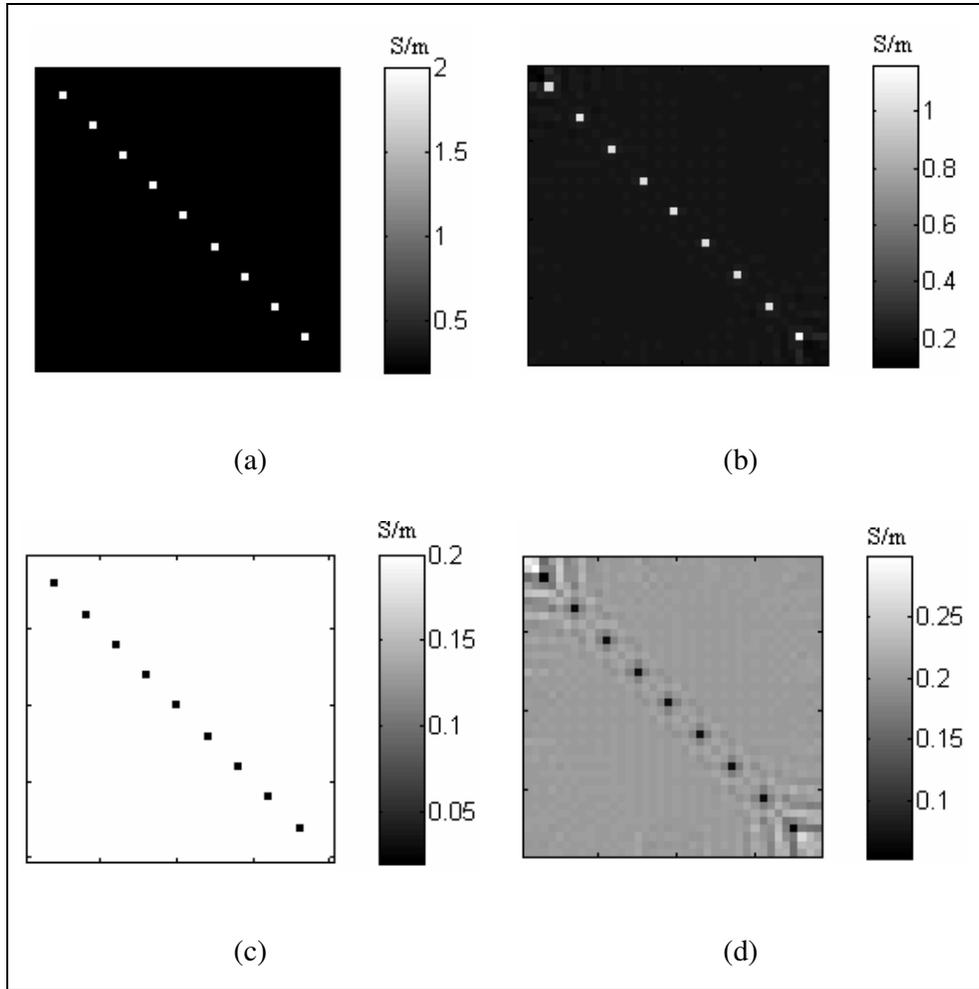


Figure 4.54: Reconstruction by Algebraic Reconstruction algorithm for Model-3, (after iteration 2) (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case. (c) Real conductivity distribution for resistive case. (d) Reconstructed conductivity distribution for resistive case.

As seen in Figure 4.54, the Algebraic Reconstruction algorithm successfully reconstructs conductivity distribution for resistive case and conductive case. This is an advantage for this algorithm. The effect of the one pixel sized square objects to the around neighboring pixels are very small for conductive case and resistive case. But the effect for resistive case is more than the effect for conductive case.

4.7.4.4 Results for Model-4

In Model-4, there is a square at the center of the imaging slice. The size of the square is 2.25 cm x 2.25 cm. There are two different cases which are conductive case and resistive case. For conductive case, the square is more conductive than background. For resistive case, the square is more resistive than background. Also there are 10 different contrast levels. This means, for each contrast level square has different conductivity value. These are shown in Table 4.3 in the Section 4.2.4.

For conductive case and resistive case, 2 iterations are used for all contrast levels. The algorithm starts to converge at 2nd iteration.

For conductive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.4$ S/m (contrast level-2) are shown in Figure 4.55.

For resistive case, the real conductivity distribution and the reconstructed conductivity distribution for $\sigma_{square} = 0.1$ S/m (contrast level-2) are shown in Figure 4.56.

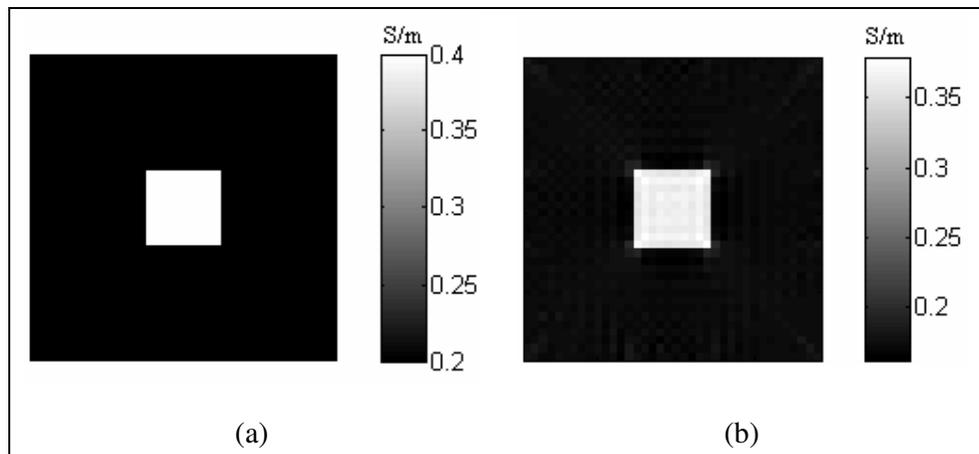


Figure 4.55: Reconstruction by Algebraic Reconstruction algorithm for Model-4, contrast level-2. (a) Real conductivity distribution for conductive case. (b) Reconstructed conductivity distribution for conductive case (after iteration 2).

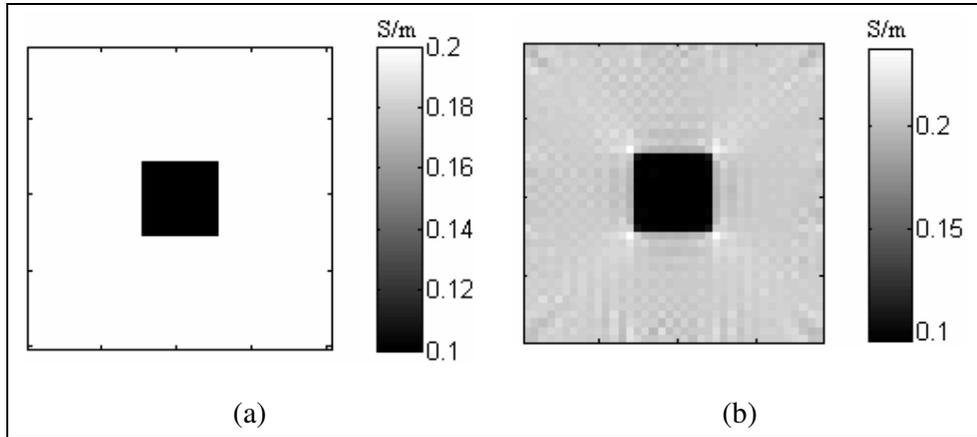


Figure 4.56: Reconstruction by Algebraic Reconstruction algorithm for Model-4, contrast level-2 (a) Real conductivity distribution for resistive case. (b) Reconstructed conductivity distribution for resistive case (after iteration 2).

For the conductive case, the conductivity of square increases as the contrast level increases. But for resistive case, the conductivity of square decreases when the contrast level increases. The square error plots according to the change of the contrast level for both cases are shown in Figure 4.57.

As seen from Figure 4.57, if we increase the difference between square conductivity value and background conductivity value, the error value of the reconstructed square will increase. This means if the contrast is increased, the error will increase.

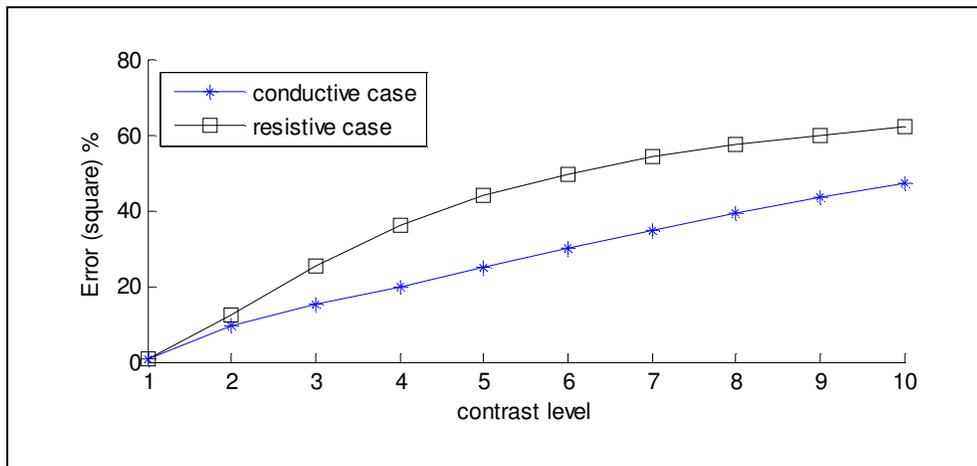


Figure 4.57: Error plots of the square for 10 different contrast levels

Also in Figure 4.58, the plot of the real conductivity values of square and mean of the reconstructed conductivity values of square for each level can be seen.

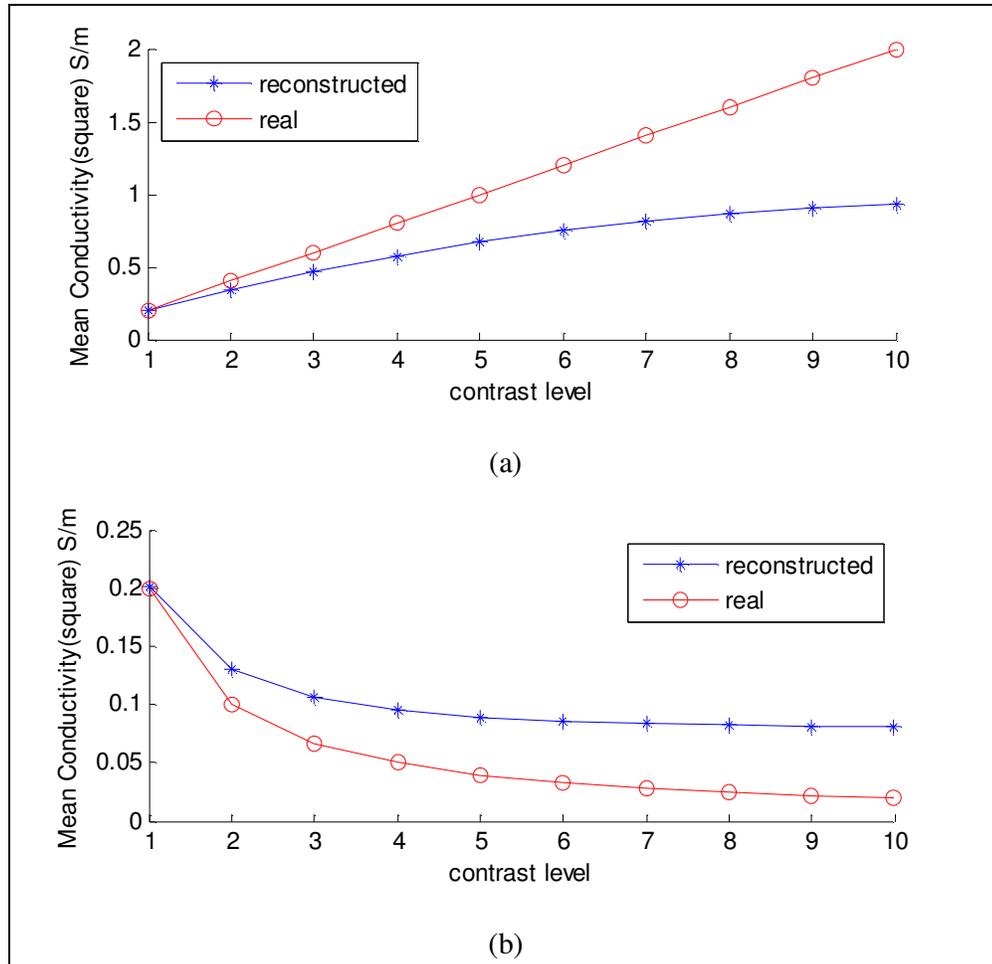


Figure 4.58: Mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels. (a) For conductive case. (b) For resistive case.

These results show that Algebraic reconstruction algorithm successfully reconstruct conductivity distributions for both cases. But the error values are high for high conductivity value differences between square and background.

4.7.4.5 Results for Thorax Model

The model shown in Section 4.2.5 is used to test the performance of Algebraic reconstruction matrix algorithm for noise-free and noisy data. The reconstructed conductivity distributions are shown in Figure 4.59.

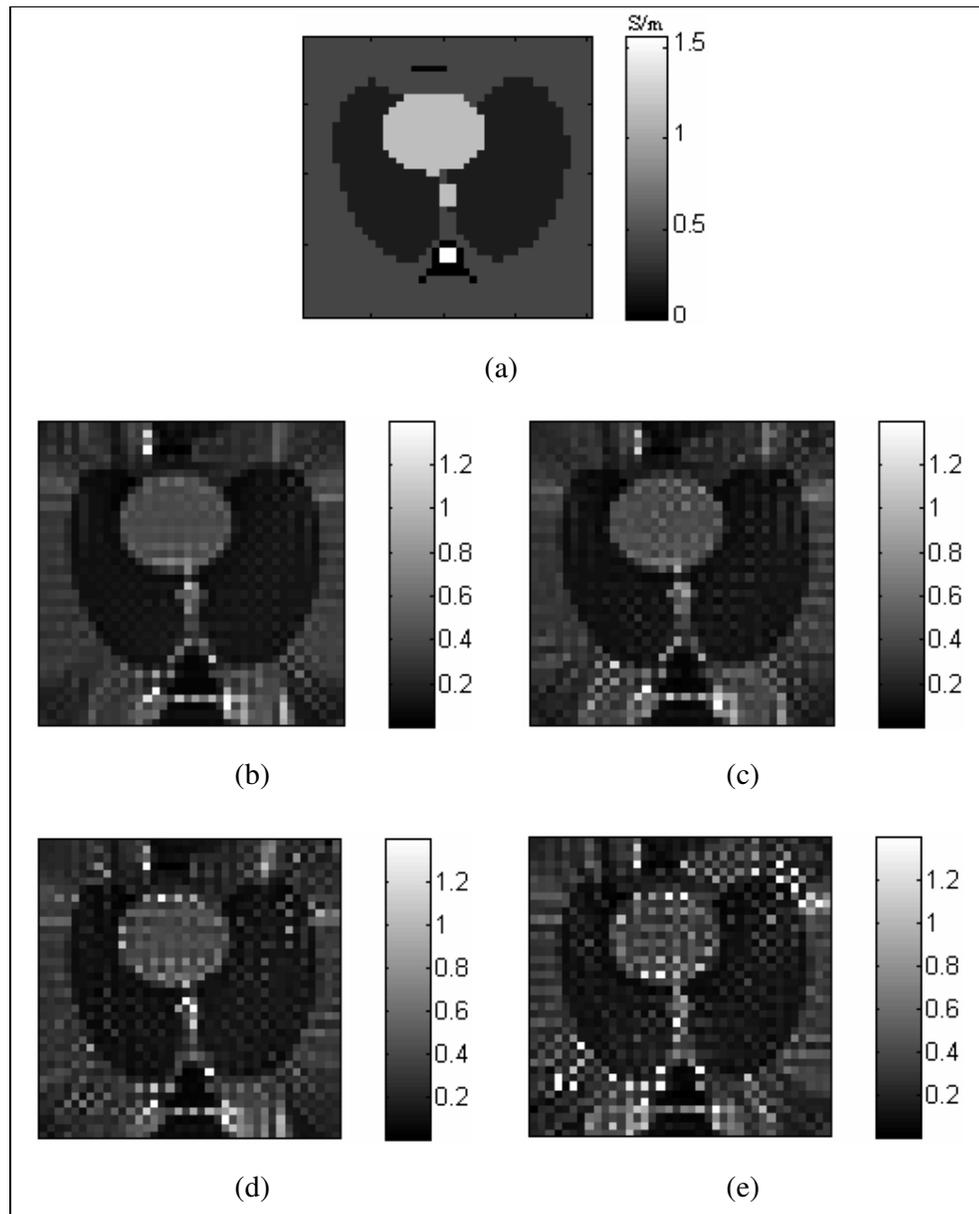


Figure 4.59: Reconstruction by Algebraic reconstruction algorithm for Thorax Model (all images are for iteration-4). (a) Real conductivity distribution. (b) Reconstructed conductivity distribution, noise-free data. (c) Reconstructed conductivity distribution SNR 30. (d) Reconstructed conductivity distribution SNR 20. (e) Reconstructed conductivity distribution SNR 13.

As seen from Figure 4.59, the reconstructed conductivity distributions for noisy data are not good as noise free data. The low SNR value affects Algebraic reconstruction algorithm. The total reconstruction errors for noisy cases and noise free case are shown in Table 4.22.

Table 4.22: Reconstructed error values for thorax model.

	<i>Noise-free case</i>	<i>SNR30</i>	<i>SNR20</i>	<i>SNR13</i>
$\mathcal{E}_{\sigma, total}$ (%)	74.31	74.66	88.76	167.58

4.8 Comparison of Four Magnetic Flux Density Based Algorithms

In this section, four magnetic flux density based reconstruction algorithms are compared under the same conditions for different computer models.

4.8.1 Noise-free data

For noise free data same conditions are used to test the performance of different algorithms. All models use two orthogonal current injection patterns. 20 mA current is injected for both injection patterns.

Three of the reconstruction algorithms are iterative algorithm. For this reason, the iteration number, which makes system stable, is chosen to compare results for each algorithm. For non-iterative reconstruction algorithm, only error values are taken for comparison.

First of all, the performance of each algorithm is compared according to the Model-1 explained in Section 4.2.1. The reconstructed conductivity distributions of Model-1 for each algorithm are shown in Figure 4.60.

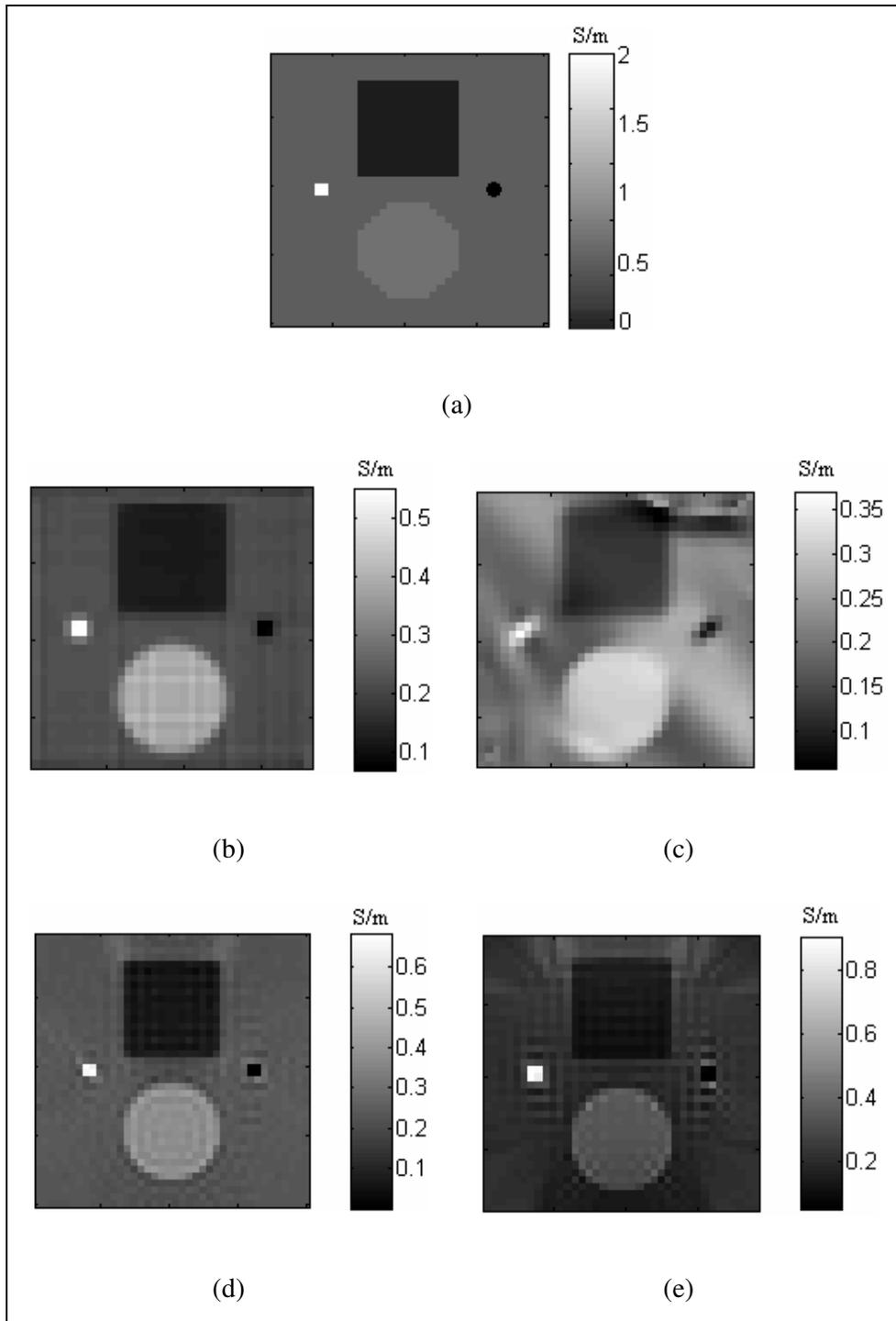


Figure 4.60: Reconstructed conductivity distributions for Model-1 (noise free data). (a) Real conductivity distribution for Model-1. (b) Reconstruction by Harmonic B_z after iteration 7. (c) Reconstruction by VGB_z after iteration 9. (d) Reconstruction by Sensitivity Matrix. (e) Reconstruction by Algebraic Reconstruction after iteration 2.

The reconstructed conductivity error for all regions of Model-1 is shown in Table 4.23.

Also the comparison of three iterative algorithms according to the total error values and iteration numbers are shown in Figure 4.61.

Table 4.23: Reconstructed error values of all regions for noise free data

	<i>Harmonic B_z</i> (7 iteration)	<i>VGB_z</i> (9 iteration)	<i>Sensitivity Matrix</i>	<i>Algebraic Reconstruction</i> (2 iteration)
$\mathcal{E}_{\sigma, total}$ (%)	33.83	42.44	31.39	27.05
$\mathcal{E}_{\sigma, background}$ (%)	11.17	20.11	17.76	22.72
$\mathcal{E}_{\sigma, bigsquare}$ (%)	23.05	30.82	33.73	43.66
$\mathcal{E}_{\sigma, bigcircle}$ (%)	8.08	24.73	8.59	18.27
$\mathcal{E}_{\sigma, smallsquare}$ (%)	72.67	83.70	66.81	55.96
$\mathcal{E}_{\sigma, smallcircle}$ (%)	264.30	501.88	94.35	208.23

Table 4.22 shows that the Algebraic reconstruction algorithm gives the lowest total error. But the total error value for Sensitivity Matrix algorithm is very close to the Algebraic Reconstruction algorithm's error value. All of the iterative algorithms have high reconstruction errors for small object, which has small conductivity value.

Also in Figure 4.60, it can be seen that, Sensitivity Matrix algorithm gives good results. Harmonic B_z algorithm and VGB_z algorithm have blurring effects. The Harmonic B_z result seems good. If the conductivity image for Harmonic B_z algorithm is compared with the Sensitivity Matrix algorithm or Algebraic Reconstruction algorithm, it can be seen that the edges of objects for Sensitivity

Matrix algorithm and Algebraic reconstruction algorithm are sharper than Harmonic B_z image.

Figure 4.60 shows that there are artifacts at the corners of the reconstructed conductivity distribution for VGB_z algorithm. If we increase the size of the electrode, we will reduce these artifacts on corners of image.

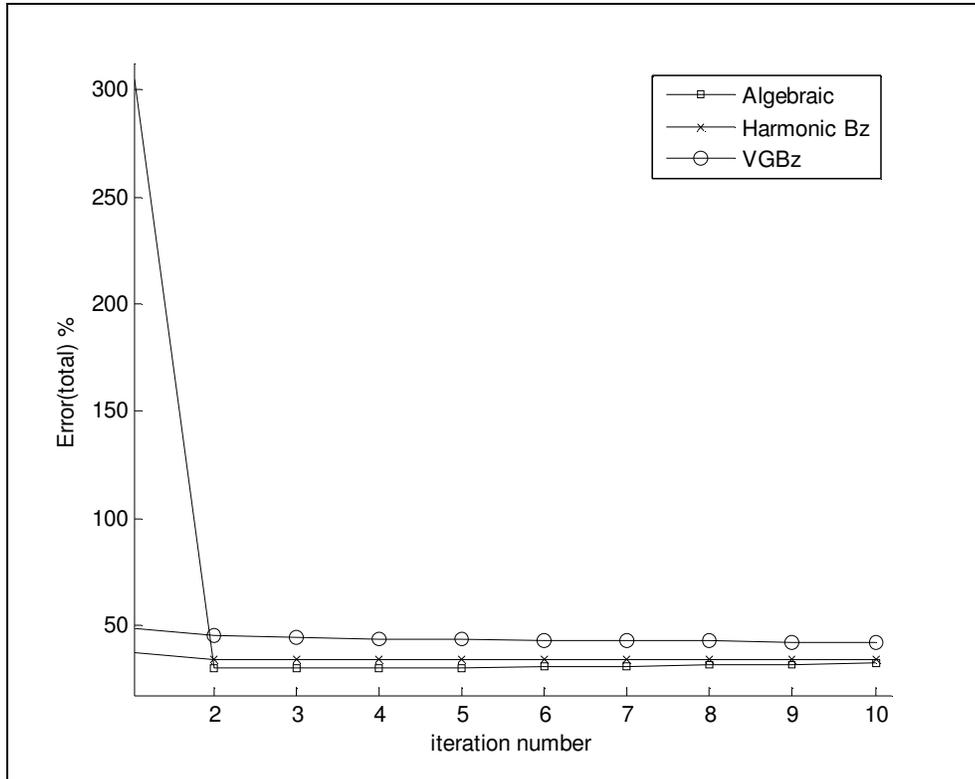


Figure 4.61: Comparison of three iterative algorithms according to the total error values and iteration numbers.

Figure 4.61 show that Algebraic Reconstruction algorithm has the lowest total error values in three iterative reconstruction algorithms. If the number of iteration increases, the reconstructed error values will decrease. Because this kind of algorithms correct themselves at each iteration until they converge a point. According to this, algorithms give good results for larger iteration numbers. But iteration is the main disadvantage for this kind of algorithms. Iterative algorithms need large computation times. For example, a non-iterative Sensitivity Matrix

algorithm needs 822 seconds to reconstruct a conductivity distribution. An iterative Harmonic B_z algorithm needs 314 seconds for one iteration; this means for 7 iterations it needs $7*314=2198$ seconds to reconstruct a good conductivity distribution.

The spatial resolution performances of all algorithms are tested with different positions shown in Table 4.24. The spatial resolution performances of all algorithms are tested by calculating the Full-Width-at-Half-Maximum (FWHM) values.

Table 4.24: Number of pixels in the gap, for different positions

Position	1	2	3	4	5	6	7	8	9	10
Number of pixels in the gap	1	3	5	7	9	11	15	19	23	27

Models used in simulation have a pixel size equal to 0.225 cm x 0.225 cm. Model-2, which is explained in Section 4.2.2, is used to measure the FWHM values. The calculated FWHM values for all reconstruction algorithms are shown in Figure 4.62, for 10 different positions.

Figure 4.62 shows that FWHM values for Sensitivity Matrix algorithm and Algebraic Reconstruction algorithm are very close to the pixel size. This means spatial resolution performances of these two algorithms are better than the other two reconstruction algorithms. The gap between two objects in Model-2 is equal to one for position-1. Only for position-1, two square elements are merged. This means two object affects each other for one pixel gap. But for other positions, these effect is disappeared.

For resistive case and conductive case, the calculated FWHM values are independent from the position of two square elements.

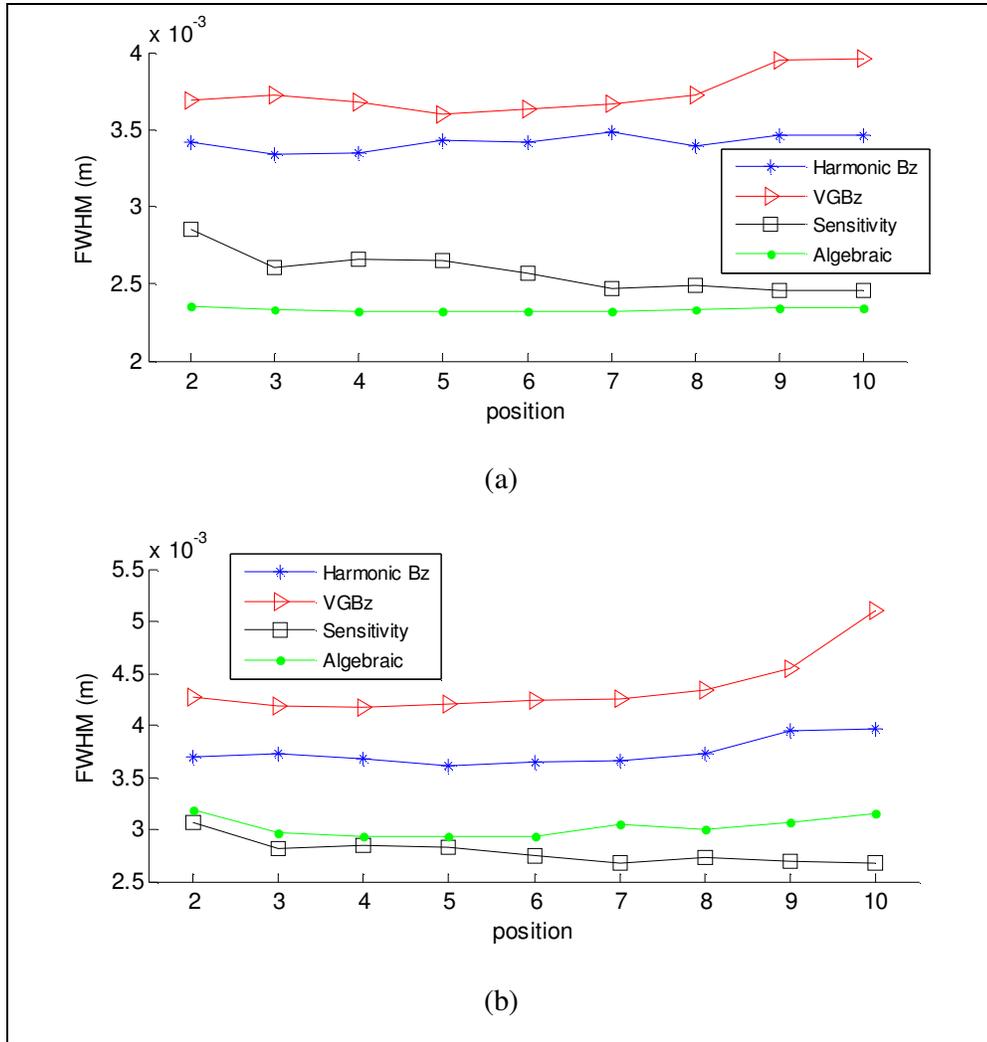


Figure 4.62: Calculated FWHM according to the positions of two square elements. (a) Conductive case. (b) Resistive case.

Model-4, explained in Section 4.2.4, is used to test the reconstruction performance according to the change of the difference between background conductivity and object conductivity.

In Figure 4.63, the comparison plot of the real conductivity values and mean of the reconstructed conductivity values of square for each contrast level can be seen.

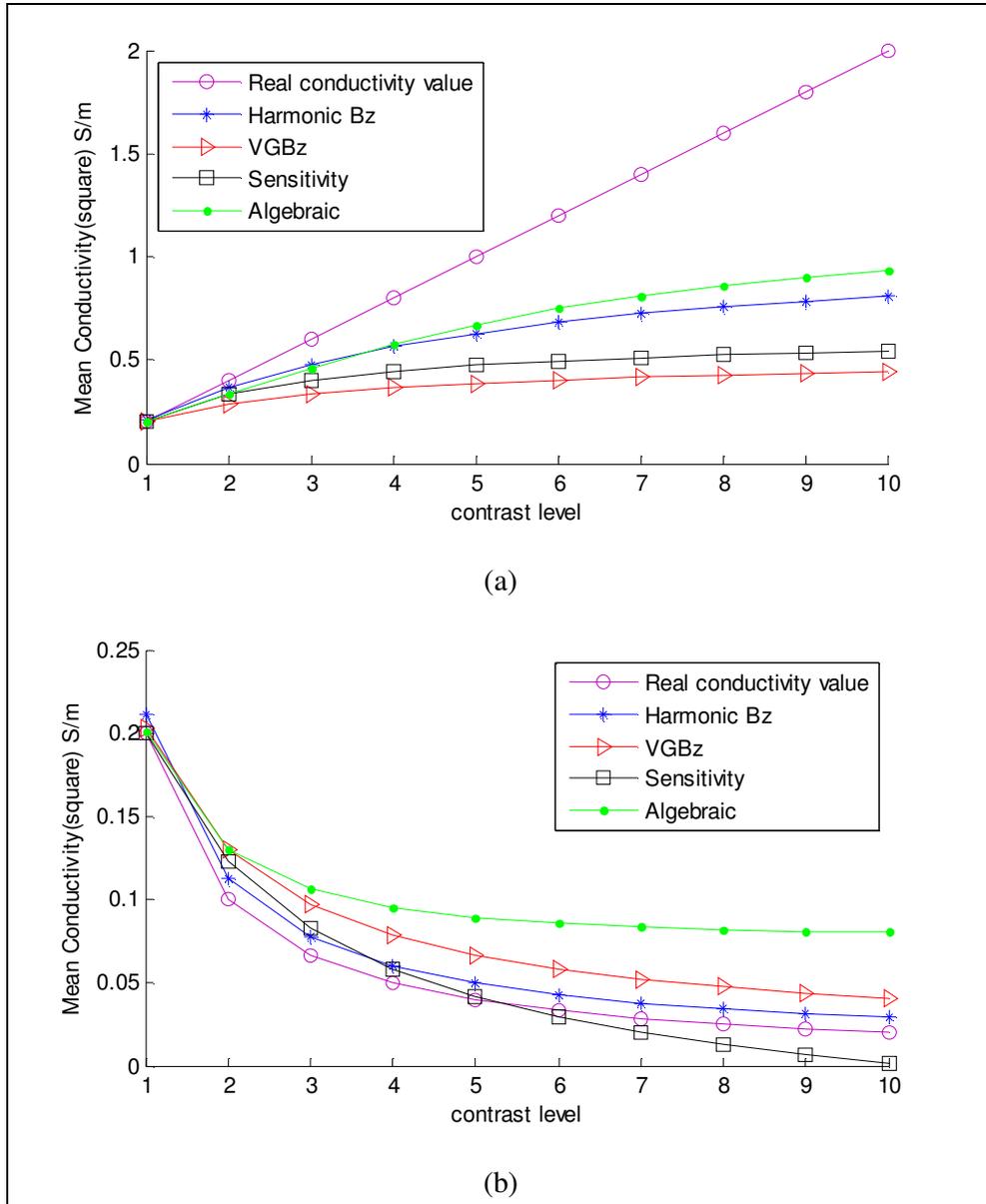


Figure 4.63: Comparison of mean real conductivity and mean reconstructed conductivity versus 10 different contrast levels, for all reconstruction algorithms. (a) For conductive case. (b) For resistive case.

For conductive case, Algebraic Reconstruction algorithm reconstructs near conductivity values to the real conductivity values. For this reason, performance of Algebraic Reconstruction algorithm for conductive case is better than other reconstruction algorithms. But for resistive case, Harmonic B_z algorithm reconstructs similar conductivity values to the real conductivity values. For this

reason, performance of Harmonic B_z algorithm for resistive case is better than other reconstruction algorithms. Also in Figure 4.63(b) it is seen that, performance of Sensitivity Matrix is very near to the Harmonic B_z performance. If we increase the difference between square conductivity value and background conductivity value, the error value of the reconstructed square will increase. This means if the contrast is high, the error of the reconstructed conductivity values will be high.

The reconstructed conductivity distributions of Thorax Model for each algorithm are shown in Figure 4.64.

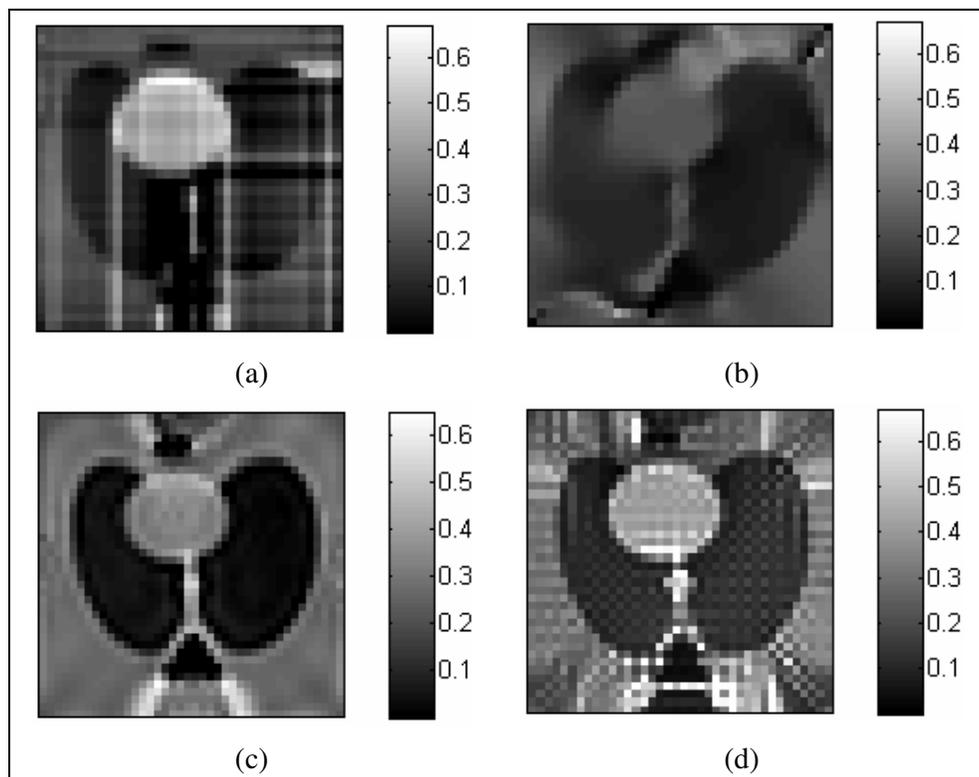


Figure 4.64: Reconstructed conductivity distributions for Thorax Model (noise free data). (a) By Harmonic B_z after iteration 2. (b) By VGB_z after iteration 3. (c) By Sensitivity Matrix. (d) By Algebraic Reconstruction after iteration 4.

For Thorax model, Sensitivity matrix algorithm has the best performance. Reconstructed conductivity image by VGB_z has blurring effect. Also the effect of

line integral can be seen from Figure 4.64(a). All reconstruction algorithms could not reconstruct spinal cord which has high conductivity value. There is bone around the spinal cord. The conductivity of bone is very small. So the current can not flow inside the spinal cord. And the reconstruction algorithms can not reconstruct this region.

4.8.2 Noisy data

For noisy data, same conditions are used to test the performance of different algorithms. All models use two orthogonal current injection patterns. 20 mA current is injected for both injection patterns. SNR is chosen as 30 to test the performance of four reconstruction algorithms.

Three of the reconstruction algorithms are iterative algorithm. For this reason, the iteration number, which makes system stable, is chosen to compare results for each algorithm. For non-iterative reconstruction algorithm only error values are taken for comparison.

First of all, the performance of each algorithm is compared according to the Model-1 explained in Section 4.2.1. The reconstructed conductivity distributions of Model-1 for each algorithm are shown in Figure 4.65.

The reconstructed conductivity error for all regions of Model-1 is shown in Table 4.25.

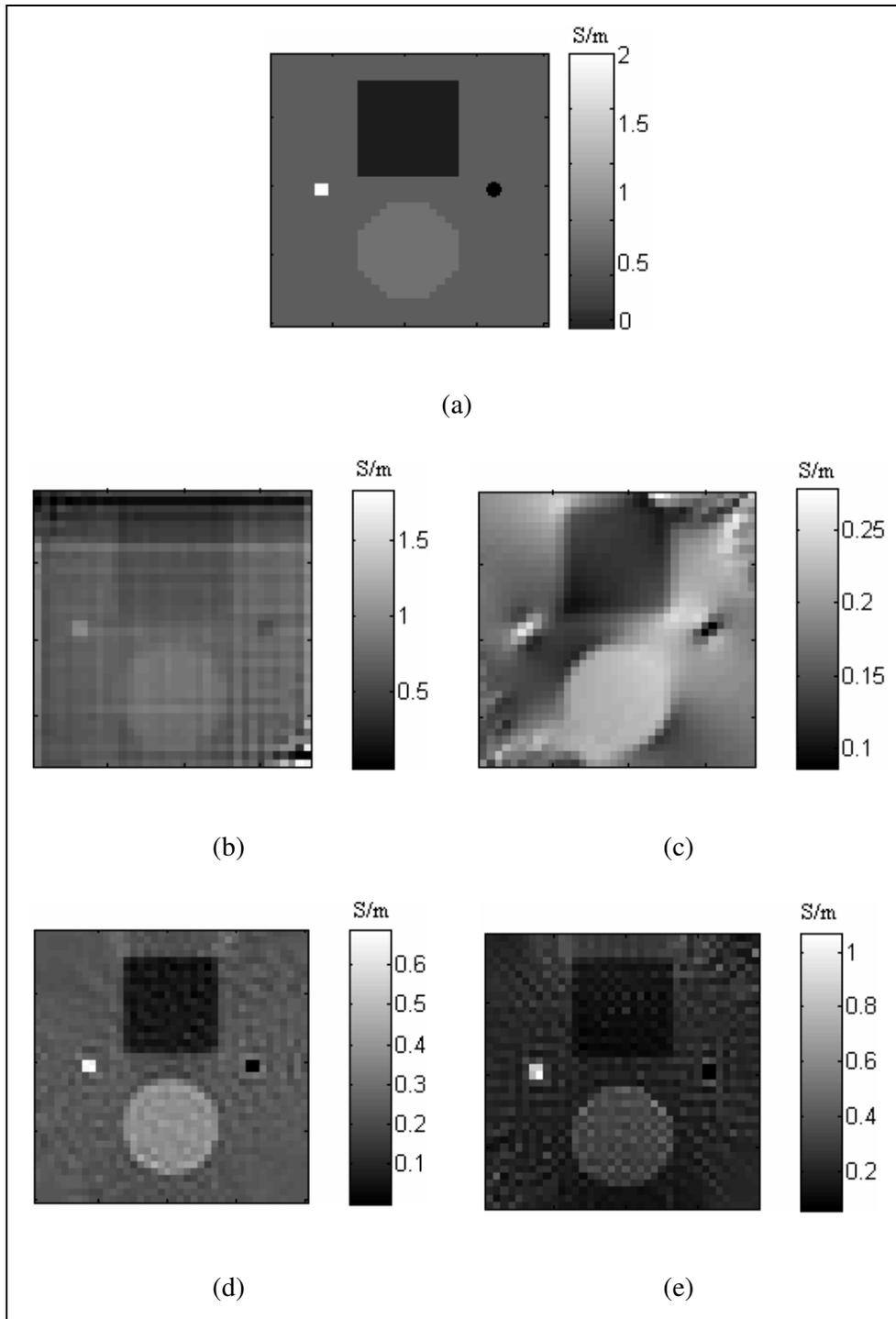


Figure 4.65: Reconstructed conductivity distributions for Model-1 (SNR 30). (a) Real conductivity distribution for Model-1. (b) Reconstruction by Harmonic B_z after iteration 7. (c) Reconstruction by VGB_z after iteration 9. (d) Reconstruction by Sensitivity Matrix. (e) Reconstruction by Algebraic Reconstruction after iteration 2.

Table 4.25: Reconstructed error values of all regions for noisy data

	<i>Harmonic B_z</i> (7 iteration)	<i>VGB$_z$</i> (9 iteration)	<i>Sensitivity Matrix</i>	<i>Algebraic Reconstruction</i> (2 iteration)
$\mathcal{E}_{\sigma, total}$ (%)	192.19	42.67	32.41	33.14
$\mathcal{E}_{\sigma, background}$ (%)	245.58	20.99	20.57	29.59
$\mathcal{E}_{\sigma, bigsquare}$ (%)	445.13	31.09	37.92	52.86
$\mathcal{E}_{\sigma, bigcircle}$ (%)	103.56	24.73	9.78	21.44
$\mathcal{E}_{\sigma, smallsquare}$ (%)	51.44	83.75	66.62	55.19
$\mathcal{E}_{\sigma, smallcircle}$ (%)	2744.00	504.20	94.30	210.32

As shown in Figure 4.65 and Table 4.25, the reconstructed conductivity distribution are not good as noise free data. Especially Harmonic B_z algorithm is very sensitive to the noise. For noisy data, the performance of Harmonic B_z algorithm is poor because of the twice differentiation of noisy B_z data. Signal-to-noise ratio must be high to take good results.

The reconstruction algorithm which gives third lowest error is Variational Gradient B_z algorithm. Because VGB $_z$ algorithm uses only one differentiation of B_z . So it works successfully for noisy data. But there is a blurring effect on VGB $_z$ reconstruction image.

According to the Table 4.25, Sensitivity Matrix algorithm's total reconstruction error value is the lowest error. The effect of noise is very low for this algorithm.

Also, Algebraic reconstruction algorithms' error value is very near to Sensitivity Matrix error value. These two algorithms are not sensitive as Harmonic B_z algorithm. As seen from Figure 4.65, the edges of objects for Sensitivity Matrix and Algebraic Reconstruction algorithms are sharper than the other two algorithms' reconstructed images.

The reconstructed conductivity distributions of Thorax Model for each algorithm are shown in Figure 4.66.

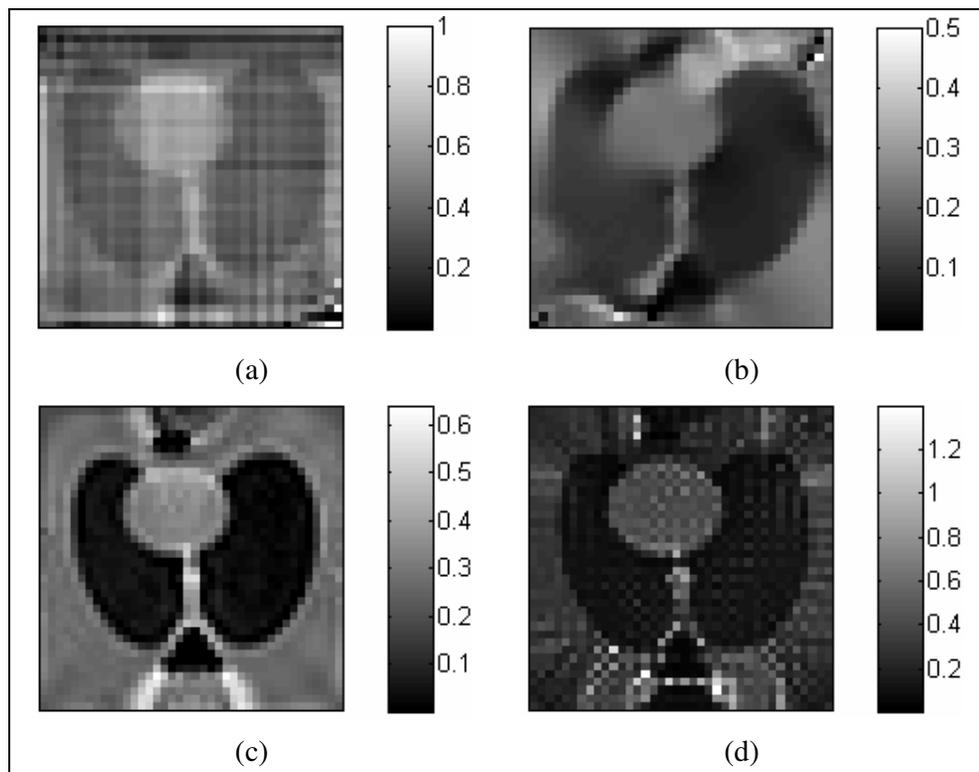


Figure 4.66: Reconstructed conductivity distributions for Thorax Model (SNR30). (a) By Harmonic B_z after iteration 2. (b) By VGB_z after iteration 3. (c) By Sensitivity Matrix. (d) By Algebraic Reconstruction after iteration 4.

Sensitivity matrix algorithm has the best reconstruction performance for noisy data. The number of current patterns or the size of the electrode can be increased to improve results for noisy data.

4.9 Experimental Results for Magnetic Flux Density Based Reconstruction Algorithms

In this section, the measured B_z data taken from 0.15 Tesla METU MRI system are shown. Four magnetic flux density based reconstruction algorithm are tested with the measured experimental data.

For experimental data, the phantom, which is explained in Section 4.3, was used. Only z component of \vec{B} was measured. Because, the reconstruction algorithms explained in this study use only one component of magnetic flux density.

The following steps were used to extract magnetic flux density from an MRI image.

1. First, for vertical current injection pattern, the phase image of slice was taken with positive current.
2. For vertical current injection pattern, the phase image of slice was taken with negative current.
3. For horizontal current injection pattern, the phase image of slice was taken with positive current.
4. For horizontal current injection pattern, the phase image of slice was taken with negative current.
5. To produce B_z data for vertical current, the phase image of vertical current injection pattern with negative current was subtracted from the phase image of positive current case.
6. To produce B_z data for horizontal case, the phase image of horizontal current injection pattern with negative current was subtracted from the phase image of positive current case.
7. The target regions of these images were taken by masking the undesired regions.

8. Each subtracted images were divided by gyromagnetic ratio (γ), current applied time per excitation (T_C). Finally we produce z component of magnetic flux density.

The measured B_z data for vertical current injection pattern and horizontal current injection pattern are shown in Figure 4.67.

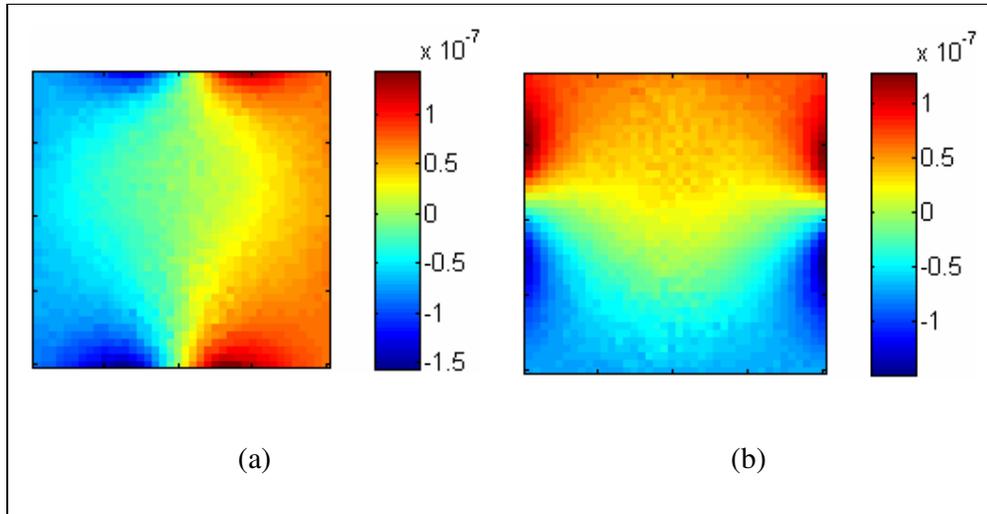


Figure 4.67: Experimental B_z data. (a) For vertical current injection pattern. (b) For horizontal current injection pattern.

As explained in Section 4.3, the experimental phantom has three regions which have different conductivity values. There is a square object which is more resistive than background and there is a circle object which is more conductive than background.

For reconstruction of conductivity distribution from the experimental data, two methods were used during the matrix inversion in Sensitivity Matrix algorithm. The first one is Singular Value Decomposition (SVD), the other one is Least Square Method (LSM) [25]. The true conductivity distribution and reconstructed conductivity distributions by Sensitivity Matrix algorithm are shown in Figure 4.68.

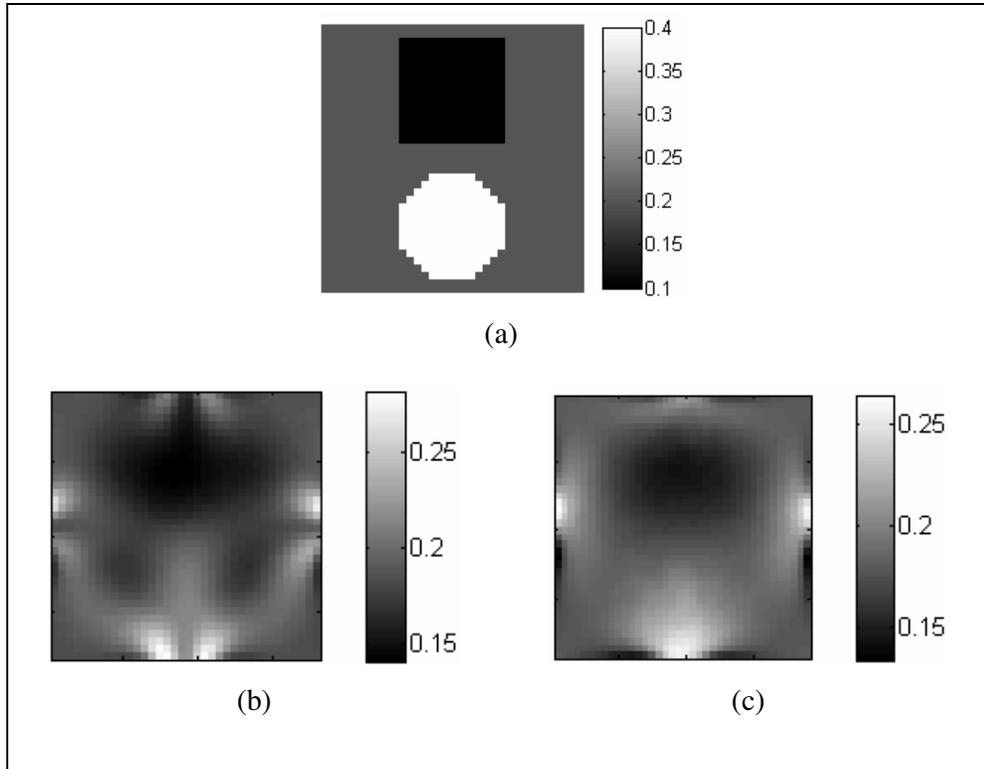


Figure 4.68: Reconstructed conductivity distributions for experimental phantom. (a) True conductivity distribution. (b) Reconstruction by Sensitivity Matrix algorithm with SVD. (c) Reconstruction by Sensitivity Matrix algorithm with LSM. (The unit of conductivity is $S.m^{-1}$)

The Sensitivity Matrix reconstruction algorithm successfully reconstructs conductivity distribution for experimental data. The error values for reconstructed conductivity distribution for Sensitivity Matrix reconstruction algorithm are shown in Table 4.26.

Table 4.26: Error values of regions for Sensitivity Matrix algorithm

	<i>Sensitivity Matrix (by SVD)</i>	<i>Sensitivity Matrix (by LSM)</i>
$\mathcal{E}_{\sigma, total}$ (%)	32.94	33.57
$\mathcal{E}_{\sigma, background}$ (%)	14.83	15.67
$\mathcal{E}_{\sigma, square}$ (%)	65.27	67.19
$\mathcal{E}_{\sigma, circle}$ (%)	47.29	47.78

As seen from Table 4.26, the total error value of experimental data with SVD is very close to the total error value of experimental data with LSM. The background conductivity distribution can be reconstructed with less error value than the other regions.

The reconstructed conductivity distributions with Harmonic B_z reconstruction algorithm, VGBz reconstruction algorithm and Algebraic reconstruction algorithm are shown in Figure 4.69.

The reconstructed conductivity distributions after iteration 1 and after iteration 4 are shown in Figure 4.69.

As seen in Figure 4.69, the Harmonic B_z reconstruction algorithm could not reconstruct conductivity distribution for experimental data. Harmonic B_z reconstruction algorithm is very sensitive to noise. The SNR is equal to 13 [24] for 0.15 Tesla METU MRI system. Also this algorithm use Laplacian of B_z and line integral during reconstruction. For this reason, the reconstruction by Harmonic B_z algorithm is not successful for this system.

The phantom used in experiments uses electrodes with small sizes. VGBz reconstruction algorithm could not reconstruct the conductivity distribution for experimental data. Because of the poor performance of VGBz algorithm for small sized electrodes. Also Algebraic reconstruction algorithm could not reconstruct conductivity distribution for experimental data. Since, the SNR value of the system is low.

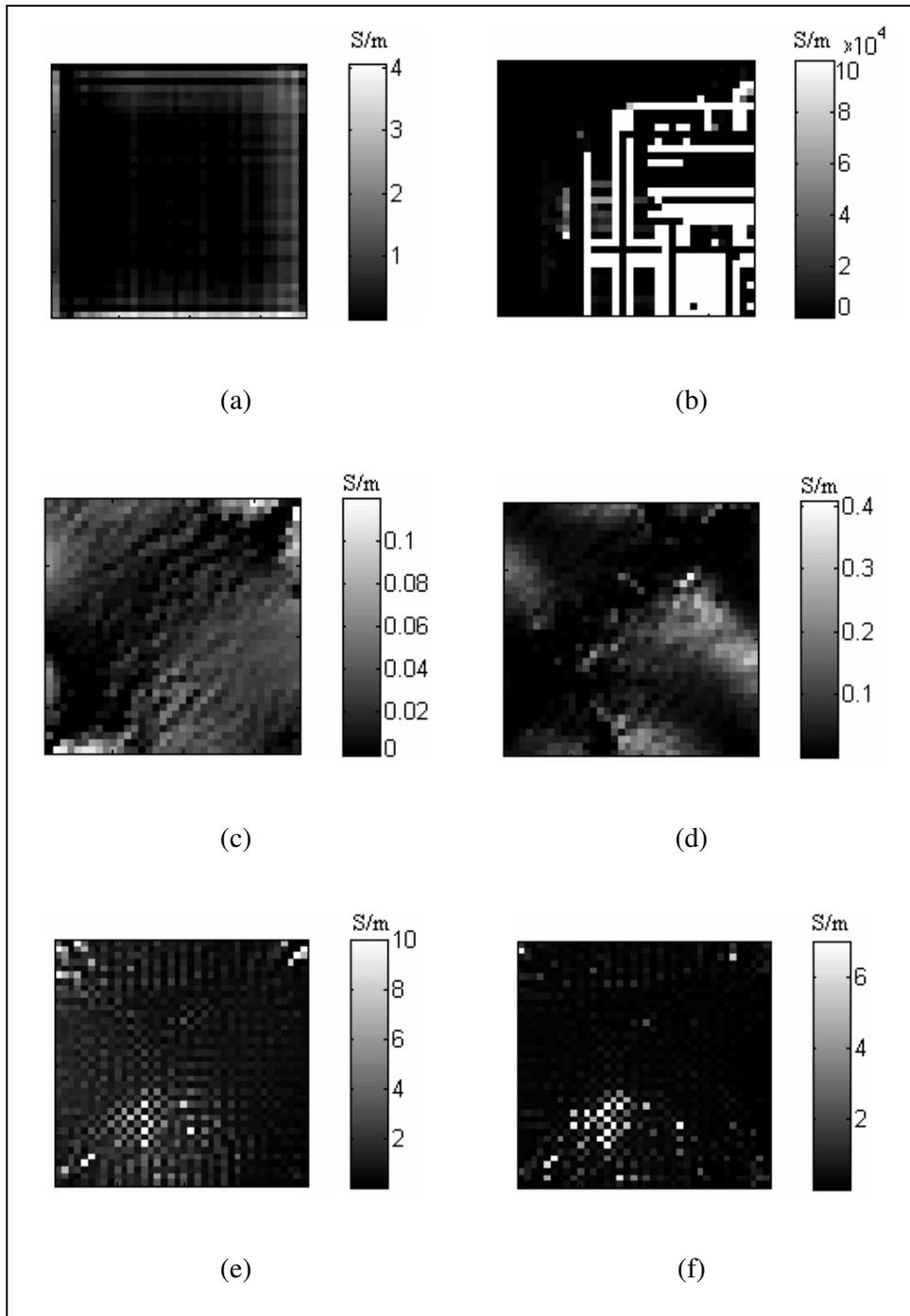


Figure 4.69: Reconstructed conductivity distributions for experimental phantom. (a) By Harmonic Bz, 1 iteration. (b) By Harmonic Bz, 4 iteration. (c) By VGBz, 1 iteration. (d) By VGBz, 4 iteration. (e) By Algebraic reconstruction, 1 iteration. (f) By Algebraic reconstruction, 4 iteration.

Then the conductivity values of the elements in experimental phantom are changed. The same conductivity value is used for background which is equal to 0.2 S/m. The conductivity value of circle object is increased to 1.9 S/m. And an insulator square object is used. With these changes, we increase the contrast between background and objects which are placed in background.

For new experimental phantom, the measured B_z data for vertical current injection pattern and horizontal current injection pattern are shown in Figure 4.70.

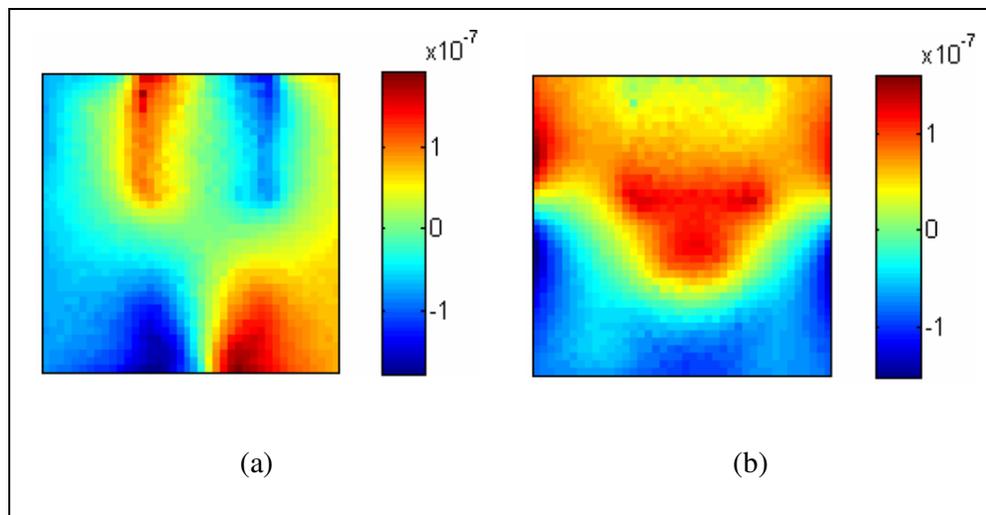


Figure 4.70: Experimental B_z data. (a) For vertical current injection pattern. (b) For horizontal current injection pattern.

For reconstruction of conductivity distribution from the new experimental data, SVD and LSM methods were used during the matrix inversion in Sensitivity Matrix algorithm. The true conductivity distribution and reconstructed conductivity distributions by Sensitivity Matrix algorithm are shown in Figure 4.71.

The error values for reconstructed conductivity distribution for Sensitivity Matrix reconstruction algorithm are shown in Table 4.27.

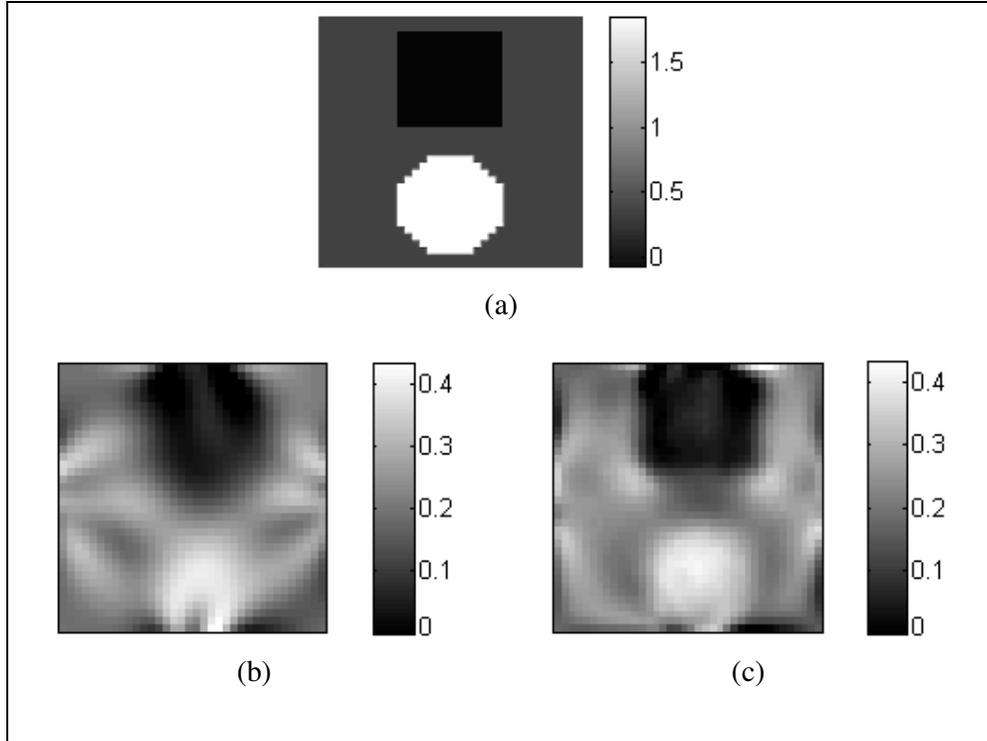


Figure 4.71: Reconstructed conductivity distributions for experimental phantom. (a) True conductivity distribution. (b) Reconstruction by Sensitivity Matrix algorithm with SVD. (c) Reconstruction by Sensitivity Matrix algorithm with LSM. (The unit of conductivity is $S.m^{-1}$)

Table 4.27: Error values of regions for Sensitivity Matrix algorithm

	<i>Sensitivity Matrix (by SVD)</i>	<i>Sensitivity Matrix (by LSM)</i>
$\mathcal{E}_{\sigma, total}$ (%)	80.60	81.00
$\mathcal{E}_{\sigma, background}$ (%)	39.18	58.23
$\mathcal{E}_{\sigma, square}$ (%)	392.57	391.13
$\mathcal{E}_{\sigma, circle}$ (%)	82.22	54.39

As seen from Table 4.27, the total error value of experimental data with SVD is very close to the total error value of experimental data with LSM. As seen from Figure 4.71, Sensitivity Matrix reconstruction algorithm successfully reconstructs conductivity distribution for experimental phantom which has elements with high conductivity differences.

The reconstructed conductivity distributions with Harmonic Bz reconstruction algorithm, VGBz reconstruction algorithm and Algebraic reconstruction algorithm are shown in Figure 4.72.

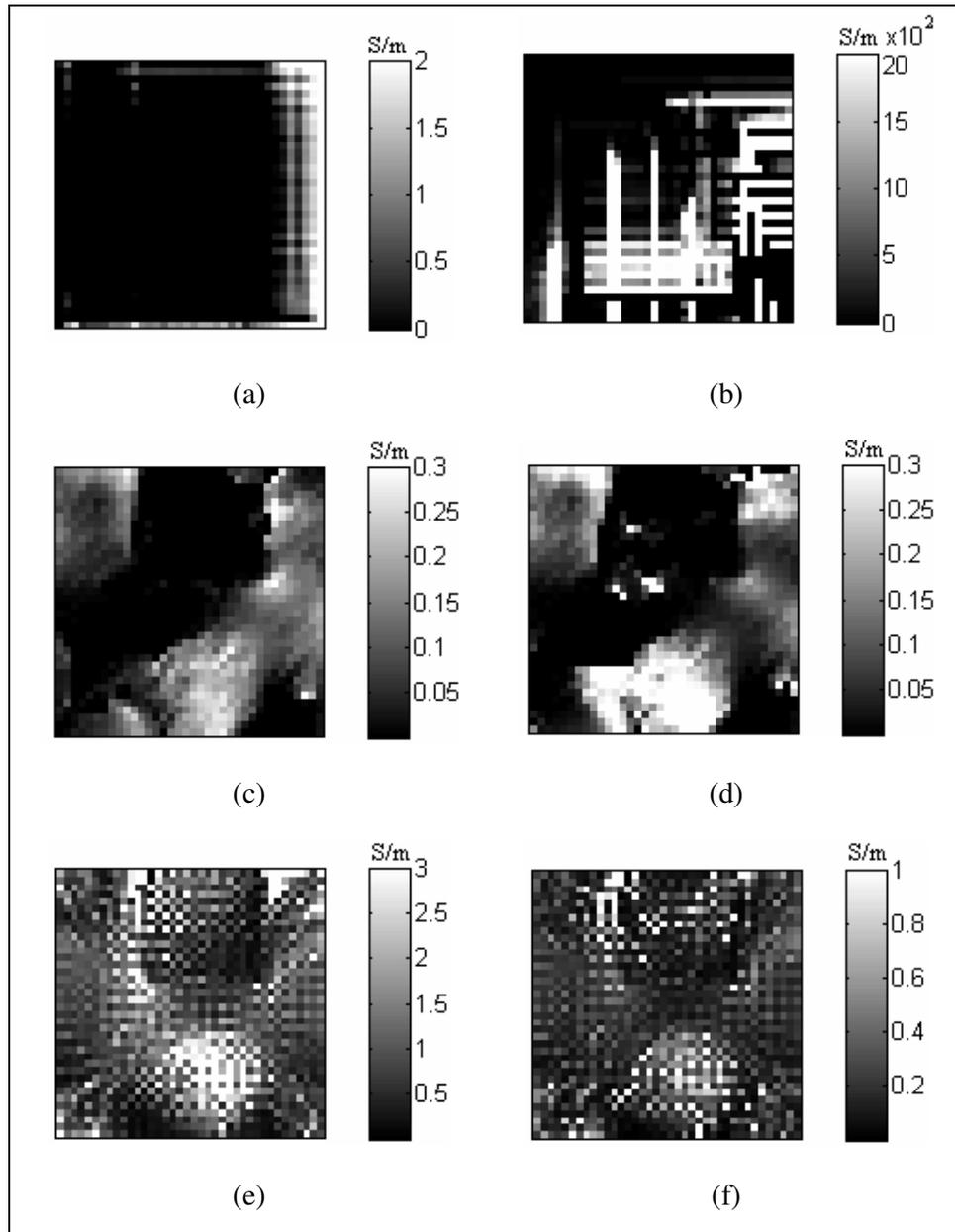


Figure 4.72: Reconstructed conductivity distributions for experimental phantom. (a) By Harmonic Bz, 1 iteration. (b) By Harmonic Bz, 4 iteration. (c) By VGBz, 1 iteration. (d) By VGBz, 4 iteration. (e) By Algebraic reconstruction, 1 iteration. (f) By Algebraic reconstruction, 4 iteration.

As seen in Figure 4.72, the Harmonic B_z reconstruction algorithm could not reconstruct conductivity distribution for experimental data. Harmonic B_z reconstruction algorithm is very sensitive to noise. Since, this algorithm uses twice differentiation of noisy B_z data and line integral.

For VGBz reconstruction algorithm and Algebraic reconstruction algorithm, the total reconstruction errors of all regions are shown in Table 4.28.

Table 4.28: Error values of all regions for VGBz algorithm and Algebraic reconstruction algorithm

	VGBz		Algebraic	
	<i>After 1 iteration</i>	<i>After 4 iteration</i>	<i>After 1 iteration</i>	<i>After 4 iteration</i>
$\mathcal{E}_{\sigma, total}$ (%)	96.20	100.14	191.41	102.96
$\mathcal{E}_{\sigma, background}$ (%)	85.01	237.99	646.83	231.38
$\mathcal{E}_{\sigma, square}$ (%)	6.25	33.14	1354.2	641.38
$\mathcal{E}_{\sigma, circle}$ (%)	459.74	409.30	397.46	385.33

As seen from Figure 4.72 and Table 4.28, reconstruction performance of Algebraic reconstruction algorithm and VGBz algorithm is very poor for experimental data. The phantom used in experiments uses electrodes with small sizes. Because of the small sized electrodes, VGBz reconstruction algorithm could not reconstruct good conductivity distribution for experimental data. Especially at the corner of the image there are artifacts. These artifacts spread into the image at iterations. Also Algebraic reconstruction algorithm's reconstruction performance is not satisfactory. In this algorithm twice differentiation of noisy B_z is used during reconstruction. For low SNR values, the effect of noise on Algebraic reconstruction algorithm is very high.

4.10 Comparison of Simulation and Experimental Results

In this section, reconstructed conductivity distributions are shown for simulated and experimental data.

As explained in Section 4.9, two phantoms are used during experiments. Each phantom has different conductivity regions. The conductivity values for each phantom are shown in Table 4.29.

Table 4.29: Experimental phantoms' conductivity values

Region	Object	Conductivity values (S/m)	
		<i>Phantom-1</i>	<i>Phantom-2</i>
1	Background	0.2	0.2
2	Square	0.1	Insulator
3	Circle	0.4	1.9

The reconstructed conductivity distributions of simulated data and experimental data for *Phantom-1* are shown in Figure 4.73.

Birgöl *et al* [24] shows that the system SNR for 0.15 T METU MRI scanner is equal to 13. A random Gaussian noise, which is explained in Section 4.4, is used for simulations. Several random noise distributions with SNR 13 are produced and the noise distribution, which gives the best reconstructed conductivity distribution, is used for simulated data.

For *Phantom-2*, the reconstructed conductivity distributions of simulated data and experimental data are shown in Figure 4.74.

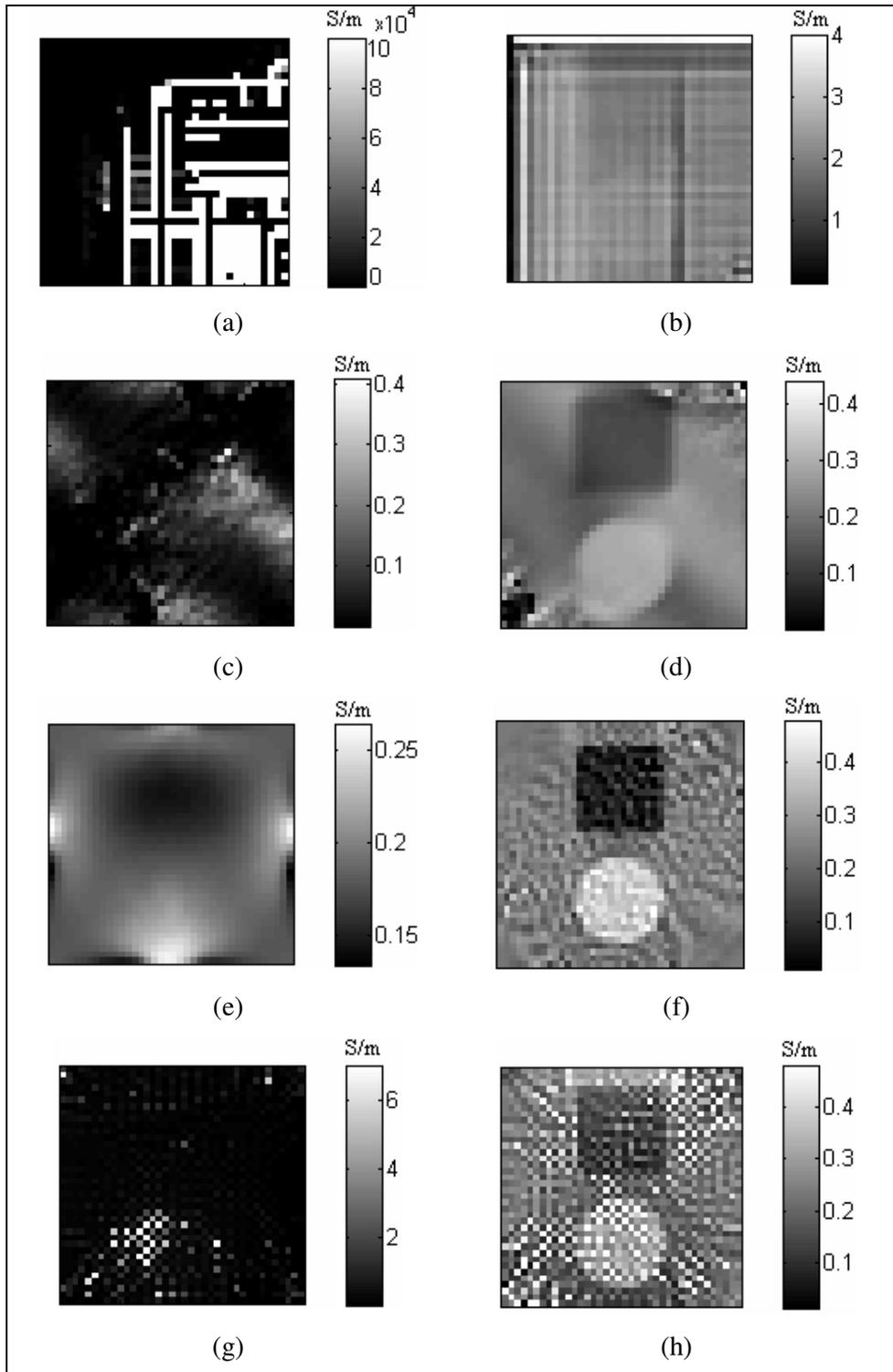


Figure 4.73: Reconstructed conductivity distributions for *Phantom-1* (after 4 iterations). (a) Experimental, Harmonic Bz. (b) Simulation, Harmonic Bz. (c) Experimental, VGBz. (d) Simulation, VGBz. (e) Experimental, Sensitivity. (f) Simulation, Sensitivity. (g) Experimental, Algebraic. (h) Simulation, Algebraic.

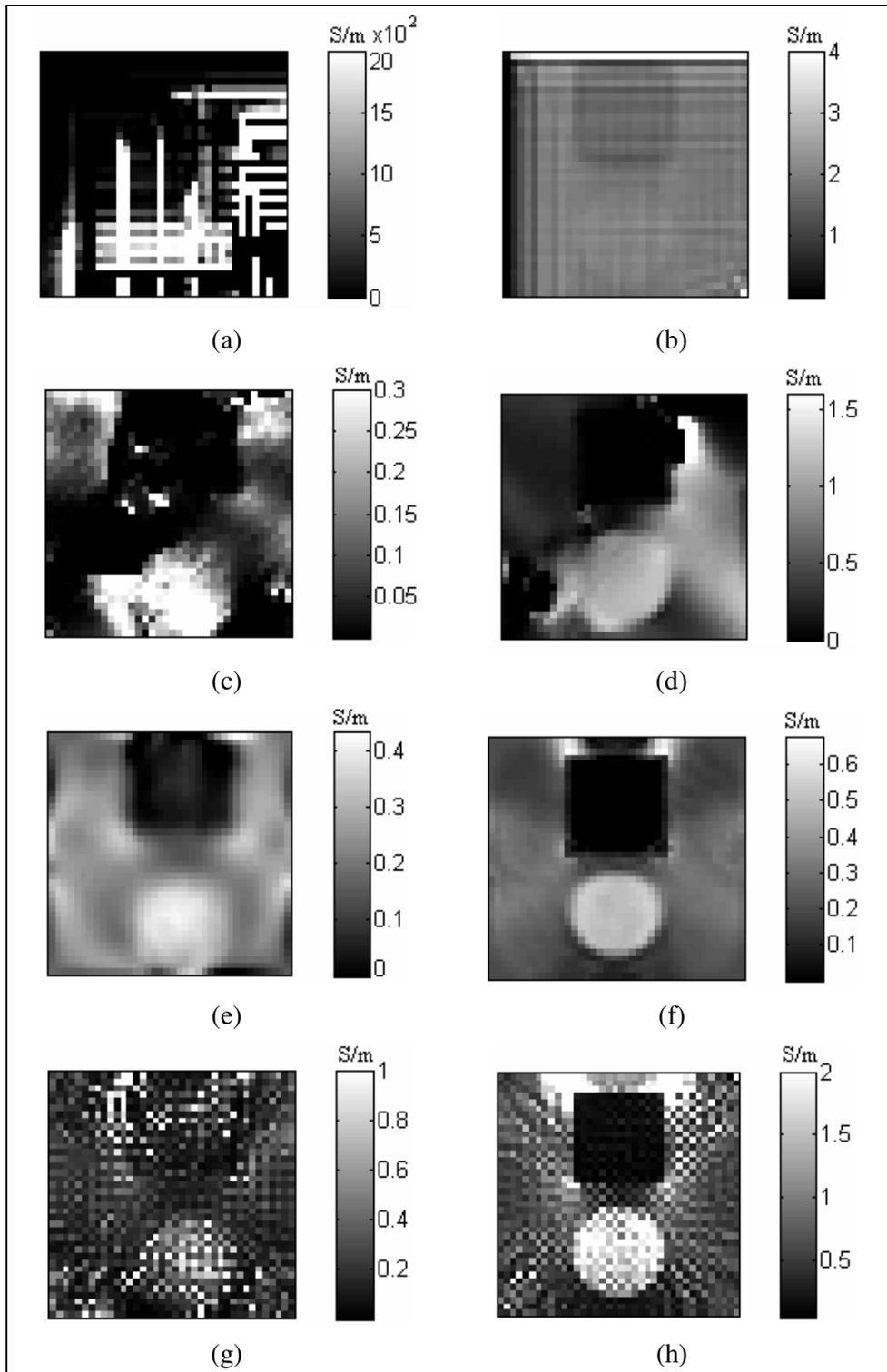


Figure 4.74: Reconstructed conductivity distributions for *Phantom-2* (after 4 iterations). (a) Experimental, Harmonic Bz. (b) Simulation, Harmonic Bz. (c) Experimental, VGBz. (d) Simulation, VGBz. (e) Experimental, Sensitivity. (f) Simulation, Sensitivity. (g) Experimental, Algebraic. (h) Simulation, Algebraic.

As seen in Figure 4.73 and Figure 4.74, the Harmonic B_z reconstruction algorithm could not reconstruct conductivity distribution for experimental data. Also the performance of Harmonic B_z reconstruction algorithm is very poor for simulated data. Harmonic B_z reconstruction algorithm is very sensitive to noise. Since, this algorithm uses line integral and twice differentiation of noisy B_z data.

Reconstruction performance of Algebraic reconstruction algorithm and VGB_z reconstruction algorithm is very poor for experimental data. The phantoms used in experiments use electrodes with small sizes. Because of the small sized electrodes, VGB_z reconstruction algorithm could not reconstruct good conductivity distribution for experimental data. For simulated data, the effect of electrode size can be seen in Figure 4.73(d) and Figure 4.74(d). Especially at the corner of the image there are artifacts. These artifacts spread into the image at iterations. Also, Algebraic reconstruction algorithm's reconstruction performance is not satisfactory. In this algorithm, twice differentiation of noisy B_z is used during reconstruction. For SNR 13, the effect of noise on Algebraic reconstruction algorithm is very high.

The Sensitivity Matrix reconstruction algorithm successfully reconstructs conductivity distributions for experimental data. Sensitivity Matrix algorithm has good reconstruction performance for experimental *phantom-2* which has elements with high conductivity differences. For simulated data, the performance of this algorithm is satisfactory for two phantoms.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

MREIT reconstructs images of electrical conductivity distribution based on magnetic flux density (\vec{B}) measurements. This magnetic flux density is generated by an externally applied current on the object. And generated \vec{B} is measured by MRI scanner. There are several methods to reconstruct conductivity distribution. MREIT reconstruction algorithms are classified into two groups. The first group use current density distributions. To produce current density distribution, three components of \vec{B} are used. For this purpose, object must be rotated inside the MRI scanner. These kinds of reconstruction algorithms are called J-based reconstruction algorithms. On the other hand, the second group reconstruction algorithms use only one component of \vec{B} , which is parallel to main magnetic field of MRI system. These kinds of reconstruction algorithms are called B-based reconstruction algorithms. Using only one component of \vec{B} , eliminates the object rotation inside MRI scanner.

In this thesis, four magnetic flux density based MREIT reconstruction algorithms were examined. The examined algorithms are: Harmonic B_z reconstruction algorithm, Variational Gradient B_z (VGB $_z$) reconstruction algorithm, Sensitivity Matrix reconstruction algorithm and Algebraic reconstruction algorithm. These reconstruction algorithms were proposed by several research groups.

To test the performance of reconstruction algorithms, different cases were used. Algorithms were tested with noise-free data, SNR 30 data, SNR 20 data and SNR 13 data. Also different models were used. Four different computer models and

one experimental phantom were used. For each model four electrodes were used to inject two different and orthogonal current. Reconstructed conductivity images, reconstruction errors and spatial resolutions were compared for four different reconstruction algorithms.

First, the algorithms are compared for noise free data. All reconstruction algorithms successfully reconstruct conductivity distributions for noise free data. Algebraic reconstruction algorithm gives the lowest total error which is equal to 27.05 %. Also, the total error values for Sensitivity Matrix algorithm and Harmonic B_z algorithm are very close to each other. The total error values are 31.39 % and 33.83 %, respectively. As said before, three of the reconstruction algorithms are iterative algorithms, Harmonic B_z , VGB_z and Algebraic reconstruction. All of the iterative algorithms have high reconstruction errors for small object, which has small conductivity value. Harmonic B_z algorithm and VGB_z algorithm have blurring effects. The Harmonic B_z result seems good. If the conductivity image for Harmonic B_z algorithm is compared with the Sensitivity Matrix algorithm or Algebraic Reconstruction algorithm, it can be seen that the edges of objects for Sensitivity Matrix algorithm and Algebraic reconstruction algorithm are sharper than Harmonic B_z image. For VGB_z reconstruction algorithm, the reconstructed conductivity distribution at the corners of the imaging slice is not good as the reconstructed conductivity distribution at the other areas. This is a result of the differentiation of B_z along x and y directions. Especially at the edges of electrodes, differentiation can affect the reconstructed conductivity distribution. This artifact starts from electrode edges and goes to the corners of image. If we increase the size of the electrode, we will reduce this artifact on corners of image.

For iterative algorithms, if the number of iteration increases, the reconstructed error values will decrease. Because this kind of algorithms correct themselves at each iteration until they converge a point. For this reason, algorithms give good results for larger iteration numbers. Because of the large computation time, iteration sometimes is a disadvantage. For example a non-iterative, Sensitivity Matrix algorithm needs 822 seconds to reconstruct a conductivity distribution. An iterative Harmonic B_z algorithm needs 314 seconds for one iteration; this

means for 7 iterations it needs $7 \times 314 = 2198$ seconds to reconstruct a good conductivity distribution.

The spatial resolution performances of all algorithms are tested by calculating the Full-Width-at-Half-Maximum (FWHM) values. The calculated FWHM values are very near to one pixel size for Algebraic reconstruction algorithm. Also Sensitivity Matrix reconstruction algorithm has similar results with Algebraic reconstruction algorithm.

To test the noise performance of reconstruction algorithms, SNR 30, SNR 20 and SNR 13 are used. The reconstructed conductivity distributions are not good as noise free data. Especially Harmonic B_z algorithm is very sensitive to the noise. For noisy data, the performance of Harmonic B_z algorithm is poor because of the twice differentiation of noisy B_z data. Signal-to-noise ratio must be high to take good results. Sensitivity Matrix algorithms' total reconstruction error value is the lowest error. It is equal to 32.41 %. The effect of noise is very low for this algorithm. Also Algebraic reconstruction algorithms' error value is very near to Sensitivity Matrix error value. The total error is 33.14 %. These two algorithms are not sensitive as Harmonic B_z algorithm.

The reconstruction algorithm, which gives third lowest error, is the Variational Gradient B_z algorithm. Because VGB $_z$ algorithm uses only one differentiation of B_z . So it works successfully for noisy data. But there is a blurring effect on VGB $_z$ reconstruction image.

The number of current patterns, amount of injected current and the size of the electrode can be increased to improve results for noisy data.

The performance of four magnetic flux density based reconstruction algorithms can be compared from performance chart which is shown in Figure 5.1. The related parameters for this performance chart are shown in Table 5.1.

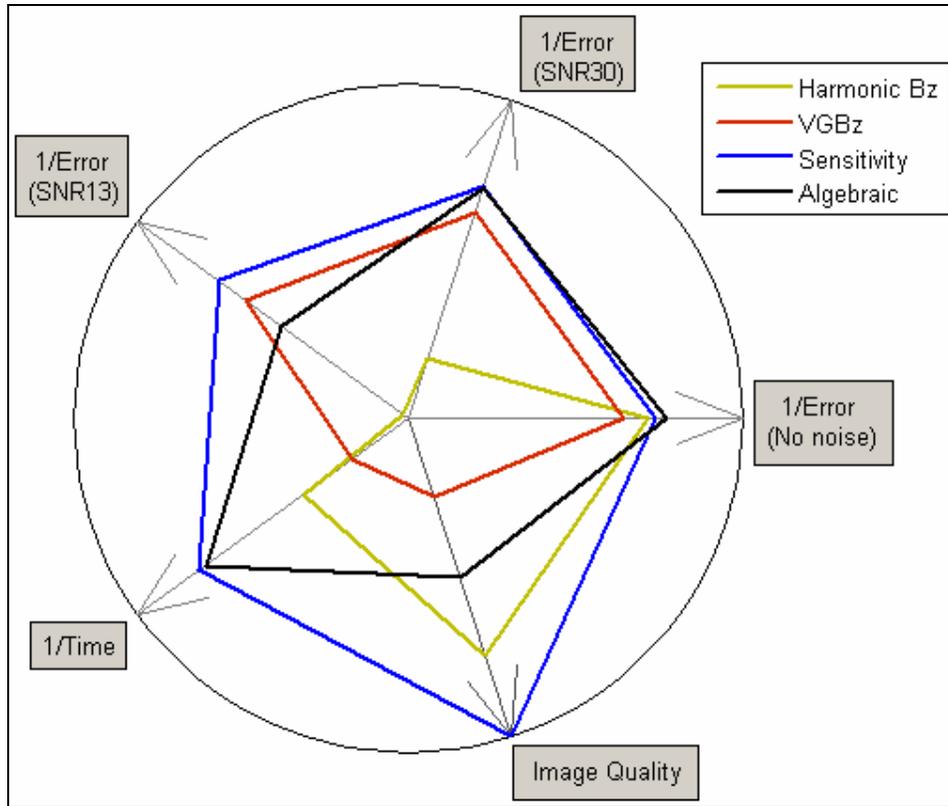


Figure 5.1: Performance chart for four B-based reconstruction algorithms.

Table 5.1: The parameters used for performance chart

	Harmonic Bz	VGBz	Sensitivity Matrix	Algebraic
$\epsilon_{\sigma, total}$ (%) (No noise)	33.83	42.44	31.39	27.05
$\epsilon_{\sigma, total}$ (%) (SNR 30)	192.19	42.67	32.41	33.14
$\epsilon_{\sigma, total}$ (%) (SNR 13)	463.33	47.75	35.93	63.71
<i>Reconstruction Time</i> (minute)	36.67	47.60	13.70	15.00
<i>Reconstructed Image Quality</i>	2	4	1	3

In Figure 5.1, the algorithm, which is placed outer part of the performance chart, has the better reconstruction performance. The algorithm, which is placed near to the origin of the performance chart, has the worse reconstruction performance.

So Sensitivity Matrix reconstruction algorithm is the best reconstruction algorithm in four B-based MREIT reconstruction algorithms.

5.2 Future work

In this thesis study, four magnetic flux density reconstruction algorithms are explained, implemented and tested with different models and conditions. Some of the possible future work can be:

- i) Testing these reconstruction algorithms with more experimental data.
- ii) Improve the noise performance of these reconstruction algorithms, by this way small amount of currents can be injected to object.

REFERENCES

- [1] Holder D (ed) 2005 *Electrical Impedance Tomography: Methods, History and Applications* (Bristol: Institute of Physics Publishing).
- [2] Henderson R. P. and Webster J. G., “An impedance camera for spatially specific measurements of the thorax”, *IEEE Trans. Biomed. Eng.*, vol. 25, pp. 250-254, 1978.
- [3] Barber D. C. and Brown B. H., “Applied Potential Tomography (Review Article)”, *J. Phys. E. Sci. Instrum.*, vol. 17, pp. 723 – 733, 1984.
- [4] Freeston L. and Tozer R. C., “Impedance imaging using induced currents”, *Physiol. Meas.*, vol. 16, pp. A257-66, 1995.
- [5] Köksal A. and Eyüboğlu B. M., “Determination of optimum injected current patterns in electrical impedance tomography”, *Clin. Phys. Physiol. Meas.*, vol. 16, suppl. A, pp. 99-109, 1995.
- [6] Scott G. C., Joy M. L. G., Armstrong R. L., and Henkelman R. M., “Measurement of non-uniform current density by magnetic resonance”, *IEEE Trans. Med. Imaging*, vol. 10, pp. 362-74, 1991.
- [7] Zhang N., “Electrical impedance tomography based on current density imaging”, Master’s thesis, University of Toronto, Dept. of Electrical and Electronics Eng. Toronto, Canada, 1992.
- [8] Ider Y. Z., Onart S., and Lionheart W. R. B., “Uniqueness and reconstruction in magnetic resonance-electrical impedance tomography (MR-EIT)”, *Physiol. Meas.*, vol. 24, pp. 591-604, 2003.
- [9] Kwon O., Woo E. J., Yoon J. R., and Seo J. K., “Magnetic resonance electrical impedance tomography (MREIT): simulation study of J-substitution algorithm”, *IEEE Trans. Biomed. Eng.*, vol. 49, pp. 160-7, 2002.

- [10] Özdemir M. S., Eyüboğlu B. M. and Ozbek O., “Equipotential projection-based magnetic resonance electrical impedance tomography and experimental realization”, *Physics in Medicine and Biology*, 49, 4765-4783, 2004.
- [11] Boyacıoğlu R., Eyüboğlu B. M., “J-substitution and filtered Equipotential projection based hybrid MREIT reconstruction algorithm”, *Proc. of Med. Physics and Biomed. Eng. World Congress*, 2009.
- [12] Ider Y. Z. and Birgül Ö., “Use of magnetic field generated by the internal distribution of injected currents for electrical impedance tomography (MR-EIT)”, *Electric Turkish J. of Elec.Eng and Comp. Sci.*, 6, no 3, 215-225, 1998.
- [13] Birgül Ö., Eyüboğlu B. M. and Ider Y. Z., “Experimental results for 2D magnetic resonance electrical impedance (MR-EIT) using magnetic flux density in one direction”, *Phys. Med. Biol.* 48, 3485-3504.
- [14] Seo J. K., Yoon J. R., Woo E. J. and Kwon O., “Reconstruction conductivity and current density images using only one component of magnetic field measurements”, *IEEE Trans. On Biomedical Engineering*, 50, 1121-1124, 2003.
- [15] Oh S. H., Lee B. I., Woo E. J., Lee S. Y., Cho M. H., Kwon O. and Seo J. K., “Conductivity and current density image reconstruction using Harmonic B_z algorithm in magnetic resonance electrical impedance tomography”, *Phys. Med. Biol.*, 48, 3101-3116, 2003.
- [16] Park C., Park E. J., Woo E. J., Kwon O. and Seo J. K., “Static conductivity imaging using variational gradient B_z algorithm in magnetic resonance electrical impedance tomography”, *Physiol. Meas.*, 25, 257-269, 2004.
- [17] Kwon O., Park C., Park E. J., Seo J. K. and Woo E. J., “Electrical conductivity imaging using a variational method in B_z -based MREIT”, *Inverse Problem*, 21, 969-980, 2005.

- [18] Ider Y. Z., Onart S., “Algebraic reconstruction for 3D magnetic resonance–electrical impedance tomography (MREIT) using one component of magnetic flux density”, *Physiol. Meas.*, 25, 281-294, 2004.
- [19] Jeon K., Lee C. O., Woo E. J., Kim H. J. and Seo J.K., “MREIT conductivity imaging using the local harmonic B_z algorithm with inpainting”, *EIT Conference*, 2008.
- [20] Huixian W., Yuyu W., Tao S. and Yongbo H., “A new image reconstruction algorithm for magnetic resonance electrical impedance tomography using one component of measured magnetic field”, *Eng. in Med. and Bio. 27th Annual Conference*, 2005.
- [21] Onart S., “Electrical impedance tomography based on current density imaging”, Master’s thesis, Bilkent University, Dept. of Electrical and Electronics Eng. Ankara, Turkey, 2003.
- [22] Scott G. C., Joy M. L. G., Armstrong R. L. and Henkelman R. M., “Sensitivity of magnetic resonance current-density imaging”, *J. of Magnetic Resonance*, vol. 97, pp. 235-254, 1992.
- [23] Lorca D. M., “Implementation and comparison of reconstruction algorithms for magnetic resonance-electric impedance tomography (MR-EIT)”, Master’s thesis, METU, Dept of Electrical and Electronics Eng. Ankara, Turkey, 2007.
- [24] Birgül Ö., Eyüboğlu B. M. and Ider Y. Z., “Current constrained voltage scaled reconstruction (CCVSR) algorithm for MR-EIT and its performance with different probing current patterns”, *Phys. Med. Biol.*, vol. 48, pp. 653-71, 2003.
- [25] Wikipedia, “Least square method”,
http://en.wikipedia.org/wiki/Least_square-method, 27.08.2009
- [26] Oilcenter, “TX150, TX151”,
<http://www.oilcenter.com/international/industry.asp?industry=Other>,
01.09.2009
- [27] Mikrobiyoloji,”Agar-agar”,
<http://www.mikrobiyoloji.org/genelpdf/920020280.pdf>