Abdominal fat segmentation using graph cut methods

Anamaria Marta Sirghie s102133

Kongens Lyngby 2012
IMM-M.Sc.-2012-108

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk IMM-M.Sc.-2012-108

Abstract

This thesis presents methods for quantifying the subcutaneous adipose tissue (SAT) and visceral adipose tissue (VAT) volume from 3D DIXON MRI. The test data used was acquired as follow up to a study on puberty in Danish children conducted by Copenhagen Department of Growth and Reproduction.

Quantifying the SAT volume is performed by sequentially detecting the abdomen and SAT interior boundaries using a graph-cut based approach. A 3D weighted graph is constructed in which nodes represent voxels and the edge weights are obtained using a gradient based approach. Finding a surface in the 3D volume is equivalent to finding the minimum cut in the constructed graph. A special attention is dedicated to investigating the importance of adding *'in-slice'* and *'between-slices'* edge constraints in finding the sought surface.

VAT estimation is conducted using information from both fat and water images. First a clustering is performed on the fat image, followed by a thresholding of the fat ratio image obtained from the fat and water images.

A visual inspection of the SAT and VAT segmentation results is performed.

Preface

This thesis was prepared at the department of Informatics and Mathematical Modelling of the Technical University of Denmark in fulfilment of the requirements for acquiring an M.Sc. in Mathematical Modeling and Computation. The work was carried out between 6th of March 2012 and 6th of September 2012 and the work amounts to the equivalent of 35 ECTS credits.

The project was supervised by Professor, PhD, Rasmus Larsen and Professor Knut Conradsen from DTU Informatics.

Kongens Lyngby, September 2012

Anamaria Marta Sirghie s102133

Acknowledgements

I would like to thank my supervisors Professor, PhD, Rasmus Larsen and Professor Knut Conradsen for their support and assistance in the development of this project. Finally, a special thanks to Andrei Sabau and Madalina Ioana Sirghie for proofreading.

List of Abbreviations and Symbols

List of Abbreviations and Acronyms

SAT	Subcutaneous adipose tissue
VAT	Visceral adipose tissue
MR	Magnetic resonance
2PD	2-point DIXON reconstruction
E2PD	Extended 2-point DIXON reconstruction
3PD	3-point DIXON reconstruction
4PD	4-point DIXON reconstruction
EM	Expectation Maximization
AV	Abdomen Volume
SATV	Subcutaneous Adipose Tissue Volume
VATV	Visceral Adipose Tissue Volume

List of Symbols

G	Weighted graph
V	Set of graph vertices
E	Set of graph edges
S	Source tree
T	Sink tree
s	Source
t	Sink
Δ_θ	Difference between adjacent columns in θ-direction
Δ_z	Difference between adjacent columns in z-direction (between adjacent slices)
$c\theta$	Inter column cost in θ-direction
cz	Inter column cost in θ-direction
Col(x,y,z)	Entry at level z in the xy defined column
C(x,y,z)	Vertex cost in the multi-column Graph
F	Fat image
W	Water image
FR	Fat-Ratio image

Contents

Chapter 1

Introduction

1.1 Project Description

Being able to robustly quantify the volume occupied by the different types of abdominal adipose tissue is of major importance as the number of obese and overweight people has increased, and due to the strong correlation between central obesity (also known as abdominal obesity) and cardiovascular diseases. Also, throughout the years studies have been conducted to establish whether an increased Body Mass Index (BMI) causes an earlier onset of puberty for girls.

Abdominal adipose tissue is divided in subcutaneous (SAT) and visceral adipose tissue (VAT). SAT is found beneath the outermost layer of skin and can be further classified into superficial adipose tissue and deep adipose tissue by Scarpa's Fascia - a thin layer of convective tissue. VAT is found surrounding the abdominal organs.

Because of its positioning around the organs, scientists consider that visceral fat plays a key role in blood pressure problems, blood clotting, insulin resistance and increased cholesterol. For this reason, visceral adipose tissue is considered to be linked to cardiovascular diseases and diabetes.

Most often estimating the overall volume occupied by the different types of

abdominal adipose tissue is done through analysis of abdominal MRI. Image segmentation is of great importance for this type of tasks - before successfully estimating the volume occupied by a type of adipose tissue, this has to be detected accurately in the MR images.

The purpose of this project is to effectively segment the boundaries enclosing the SAT (the abdomen boundary and SAT interior boundary) and segmenting the VAT region in the intra-abdominal cavity from fat images obtained through a volume interpolated breath holding sequence with DIXON reconstruction.

1.2 Previous Work

This section provides a short description of some previous projects dealing with abdominal adipose tissue segmentation.

In [14] Thomas Mosbech separates the abdominal adipose tissue in three categories: deep SAT, superficial SAT and VAT using T1-weighted MRI data. Prior to segmentation of the adipose tissue volume a pre-processing of the data is performed to eliminate the intensity inhomogeneities.

The first step is consisted of a unsupervised classification of the abdominal adipose tissue through the use of the fuzzy c-means method and thresholding. Following this, the abdomen boundary and the interior SAT boundary are detected through the use of the active contour model using membership information from the unsupervised classification of the adipose tissue.

The SAT boundaries detection was performed slice-by-slice. The Scarpa's Fascia detection was done using dynamic programming on the polar transformed data. Some of the disadvantages of using active contour models is that it can get stuck in situations of local minima, it overlooks small features and for higher accuracy, longer computation times are needed.

In [12] a morphon-based multi-scale registration method is proposed for quantifying subcutaneous, visceral and non-visceral adipose tissue from DIXON MRI images. The morphon method is an iterative registration of a prototype image to a target image. The reader is referred to [22] for more information regarding the morphon registration technique.

The inputs needed for the algorithm are a prototype image, a manual segmentation of the prototype and a target image. Taking into consideration that the adipose tissue structures can have a high variability between individuals, the

water images were used in the registration process. The morphon registration was modified for a better detection of the visceral adipose tissue, by adding an extra step in which the prototype mask is registered to a binary mask of the target image internal abdominal cavity. The target image mask is obtained using thresholding. The deformation obtained from this registration is used as an initial estimate of the deformation field.

This master thesis project is a continuation of the work done by Josephine Jensen and Cecilie Benedicte Anker in their bachelor thesis [11].

In their project, detection of SAT interior boundary and Scarpa's Fascia was performed using a graph-cut based approach, constructing a 3D graph corresponding to the image volume using the same smoothness constraints for *'in-slice'* and *'between-slices'* edges.

For finding the VAT adipose tissue two methods were compared: a method based on Fuzzy C-Means clustering and a clustering using Graph Cuts. The graph cuts based approach uses mean and variance values for adipose and non-adipose tissue for finding the terminal links costs, and no cost was used for the non-terminal edges. The implementation was done per slice and the mean and variance values were found by manual selection of a region of interest (ROI) in one slice and used in segmenting all the slice.

They have observed that the graph-cut implementation has a tendency of over-estimating the VAT volume and proposed that for better results the mean and variance must be calculated for every slice.

The goal of this project is to improve the Graph Cut based algorithm for detection of the abdomen boundary and SAT interior boundary, by using a multi-column graph based segmentation for volumetric abdominal MRI data. Also, with respect to the VAT segmentation a new method is tested based on clustering using distribution models, calculation of the fat-ratio image from the fat and water images and thresholding.

1.3 Thesis Overview

This section provides a short description of the following chapters and appendices.

Chapter 2 describes the water-fat separation concept in MR images, with a focus on the DIXON reconstruction technique.

Chapter 3 presents the data used in the project, with concern on the quality of the images.

Chapter 4 introduces the graph-cut model approach for image segmentation, focusing on the graph construction for optimal surface detection. Section 4.4 provides a small 2D example of a surface detection problem.

Chapter 5 provides the theoretical aspects of the algorithms used in the VAT segmentation procedure.

Chapter 6 presents the implementation details of the graph-cut method for SAT segmentation as a two step process. The first step consists in using the graph-cut model for finding the abdomen boundary. Afterwards a re sampling of the data is performed and the graph-cut model is used to detect the SAT interior boundary. For each of the methods a discussion of the tests performed is provided.

Chapter 7 covers the method used in performing the VAT segmentation.

Chapter 8 provides the conclusions and describes the future work perspectives.

Appendix A holds additional information regarding the data.

Appendix B presents the first, median and final slices of the final segmentation results for the 11 subjects from the original datasets and in presence of noise.

Chapter 2

Water-Fat Separation in MRI

Magnetic resonance imaging (MRI) is a non-invasive medical imaging technique used to produce images of the inside of the human body. MR imaging has high contrast in soft tissues and can provide images highlighting different tissue types using different settings. MRI use is safer than CT or X-rays use as it does not expose the patient to radiation. On the other hand, MRI scanners are more expensive than CT scanners and the scanning time for an MRI is much higher than for a CT.

In an MRI scanner a powerful magnetic field is used to align magnetization of atomic nuclei present in the body and radio frequency fields are used to alter the alignment of the magnetization. The rotating magnetic field produced by the nuclei is detectable and used to reconstruct the imaged object.

Fat-water separation importance in MR images has increased in recent years and is nowadays a commonly used procedure in MR imaging. The interest in fat-water separation methods is related to two types of problems: fat suppression and fat separation.

Fat tissues appear as a bright signal in the standard pulse sequences (T1-weighted and T2-weighted). The high intensity of the fat signal can obscure other tissues that have high T1 and T2 signal intensities, causes chemical shift

artifacts and aggravates motion related artifacts. Hence fat suppression techniques are used in imaging most of the body parts for both tissue characterization purposes or improvement of contrast enhanced images. The role of fat suppression methods is to prevent the signal coming from fatty tissue from being acquired or eliminate it after being acquired.

The most common fat suppression methods can be divided into two categories, techniques that make use of the resonance frequency of the fatty tissue, known as chemically shift selective methods, or that make use of the T1 relaxation time, also known as inversion recovery methods. The reader is referred to [4] for more information about the fat suppression methods and their advantages and disadvantages.

On the other hand, fat separation techniques perform a post-processing on the data after it has been acquired, with the goal of separating water signal from fat. In the next paragraphs a short description of the DIXON method will be presented and some of the variants of this method will be enumerated.

The first paper on a water-fat separation technique has been published in 1984 by W.T. Dixon [5]. Most of the methods used today for performing water-fat separation are variations of the DIXON method.

The DIXON method is a chemical shift based water-fat separation method. This method makes use of the difference in resonance frequencies between fat and water protons. Images are acquired at different echo times (TE), one when the fat protons and water protons are in phase and another one when the water protons and fat protons are 180° out of phase. The acquired signal is the sum of the fat and water signals when the fat and water protons are in-phase and their difference when these are out-of-phase.

Figure 2.1 depicts the echo times for fat and water signals for a 1.5T field strength, where the resonance frequency difference between fat and water is equal to 225Hz.

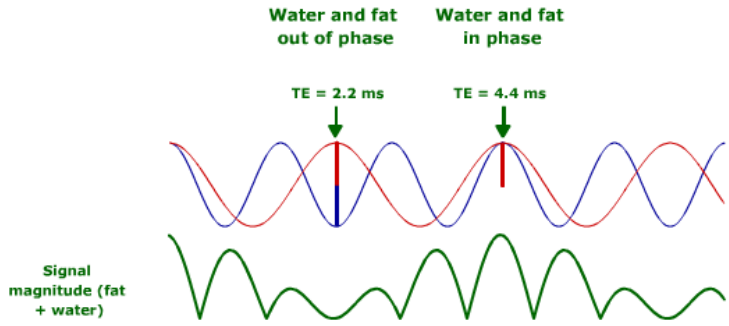
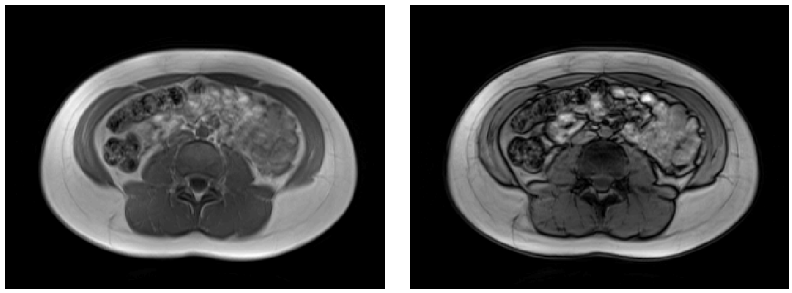


Figure 2.1: Combination of fat and water signal for a 1.5T field strength. Figure taken from [10]

Examples of in-phase and opposed phase images are found in Figure 2.2. The images are part of the data used in this project and have been acquired using a Siemens 3T MRI machine.



(a) In-phase image (b) Out-of-phase image

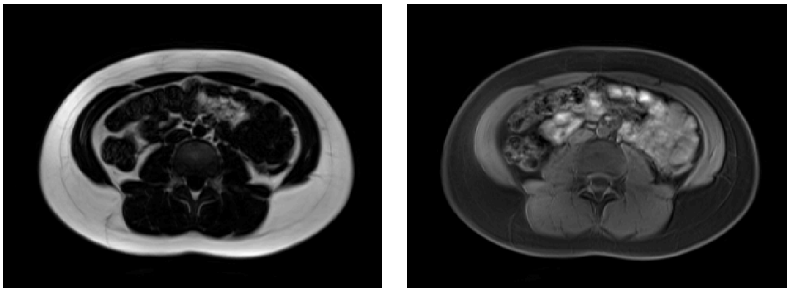
Figure 2.2: In-Phase/Out-of-Phase Images - Dataset 13, Slice 8

In Figure 2.2(b), the black line found at the border of fat and other tissue is caused by voxels containing an equal proportion of water and fat.The water and fat separation is done in the reconstruction phase through the use of a linear combination of the in-phase and out-of-phase images.

In the first approach of the DIXON method, also known as 2-point DIXON the fat and water images are obtained through the formulas described in Equation 2.1. The number of points in the naming of the method refers to the number of images acquired that have distinct water-fat phase differences.

$$F = \frac{IP - OP}{2}$$
$$W = \frac{IP + OP}{2} \tag{2.1}$$

In a fat image tissues containing fat have a high intensity value and are represented brighter (Figure 2.3(a)). In a fat saturated image, the water signal is brighter and the signal from fatty tissues is darker(Figure 2.3(b)).



(a) Fat MRI (b) Water MRI

Figure 2.3: Fat and Water Images - Dataset 13, Slice 8

A big disadvantage of the first DIXON technique was the B_0 inhomogeneity. The B_0 inhomogeneity is characterized by variations in the external magnetic field and causes intensity distortion artifacts. The value of a pixel whose signal is from only fat and one of a pixel whose signal is from only water, but is affected by B_0 inhomogeneity can not be distinguished, thus resulting in swapping of water and fat in the image.

Skinner and Glover in [19] proposed an extension to the 2-point DIXON method based on a model including phase errors due to B_0 inhomogeneities. The E2PD method works well except near water/fat boundaries where voxels represent an anatomical region with equal percentage of fat and water. When such areas are corrupted by noise estimating the phase shift caused by B_0 inhomogeneities is erroneous causing water and fat swapping. Another disadvantage of the E2PD method is the incomplete fat suppression in the water image explained in [1].

Further improvements have been made for solving the B_0 inhomogeneity problem by acquiring more images with phase differences between fat and water signals, such as 3-point DIXON (3PD) and 4-point DIXON (4PD).

In the 3PD method proposed by Schneider and Glover and described in [7], three

phase angles are used at (0°,180° and −180°), obtaining one in-phase image and two out-of-phase images. The difference between the two out-of-phase images is calculated, forming a B_0 inhomogeneity map, assuming that any differences that exist between the two out-of-phase images are due to the B_0 inhomogeneity. The B_0 inhomogeneity map calculated is used to eliminate the inhomogeneity phase shift in one of the out-of-phase images and further the E2PD or the 2PD method is used for obtaining the fat-only and water-only images.

An improved version of the 3PD method consists in acquiring two in-phase images at 0° and 360° and one out-of-phase image at 180° instead. Using the E3PD method the incorrect estimation of the B_0 inhomogeneity for the voxels where fat-water signal cancels out is overcomed. The E3PD method and a proposal of the 4PD method are described in [6].

The main disadvantage of the earlier versions of the DIXON technique is the increased scan time, which leads to degradation in quality of the fat and water images, characterized by increased image blurring or motion artifacts. An advantage of newer versions of the DIXON technique is the insensitivity to B_1 inhomogeneities and reliable removal of B_0 inhomogeneity effects. Improvements addressing both scan time and image quality using a DIXON sequence have been made and nowadays the DIXON method is present on machines provided by major vendors as Siemens, Philips and General Electric.

Currently the he water-fat separation technique implemented on Siemens MRI scanners is an extended 2PD on VIBE sequence and a 3-point DIXON on the Turbo Spin Echo sequence. Philips scanners use a modified 2PD method called mDixon and General Electric scanners provide a 3PD reconstruction technique.

As the version of DIXON reconstruction method used has an influence on the acquired fat and water images, this will also influence the algorithms used in estimating the SAT and VAT volumes.

Chapter 3

Data

The datasets used for this project was obtained as follow up to a study conducted by the Copenhagen Department of Growth and Reproduction, investigating the relationship between the Body Mass Index (BMI) and the age at which young girls reach puberty. A study finished in 2009 has shown that Danish girls reach puberty one year earlier comparative to 15 years ago. A group of girls with ages between 11-12 years old were scanned longitudinally as part of the follow up procedure.

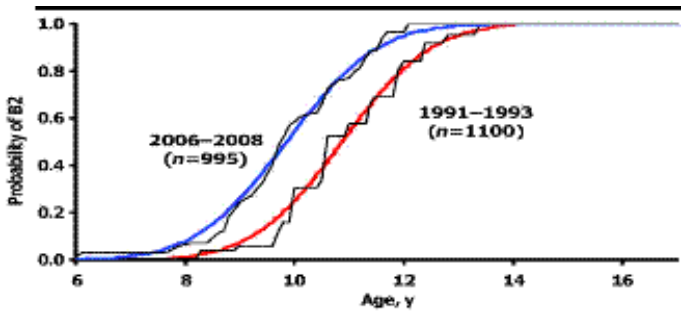


Figure 3.1: The probability of breast development(B2) according to age (in years) in Danish girls from the Copenhagen area investigated in, respectively, 1991-1993 and 2006-2008.[15]

The data consists of transversal abdominal MR scans from 13 girls with ages between 11-12 years old and was acquired using a Siemens Verio 70 cm bore 3T MRI scanner at Rigshospitalet. The data acquisition was done using the DIXON method integrated with the 3D-VIBE gradient echo sequence. The delivered data consists of in-phase, opposed-phase, water-only and fat-only images.

The scan area is a part of the lumbar region between L1 and L4 lumbar vertebraes, the region marked by a red ellipse in Figure 3.2.

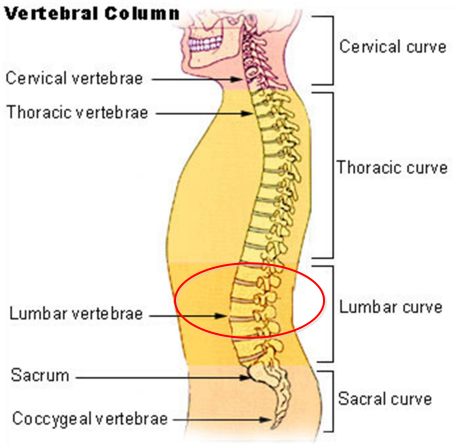
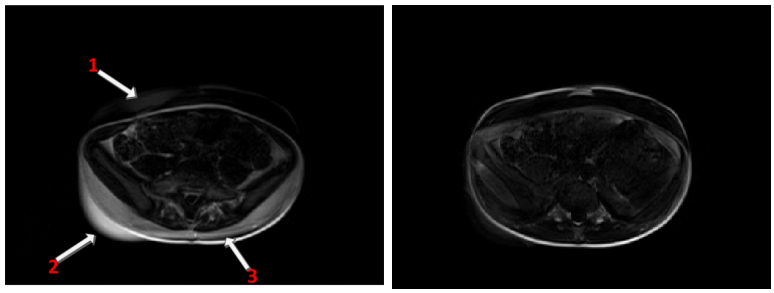


Figure 3.2: Abdominal MRI region of interest [21]

Table A.3 comprises a list of slice resolutions for each dataset along with number of slices and the slice thickness.

Out of the 13 datasets 2 have been excluded from the analysis as the fat and water images have been swapped and were highly affected by artifacts.

The quality of the scans provided is fairly good for most of the datasets. Although, it has been observed that in some of the datasets artifacts are present, such as: overlapping of the first and last slices in the dataset (1), a bright extended band of the left side of the posterior abdominal wall (2), and water-fat swap caused by B_0 inhomogenity (3) in first 4-5 slices of dataset 1 as described in Figure 3.3.



(a) Dataset 1,Slice 1 (b) Dataset 1,Slice 48

Figure 3.3: Artifacts in Dataset 1

Dataset 2 contains a bright extension of the entire posterior abdominal wall in slices 1-7 and a low intensity of the fatty tissue in slices 1-4. Another problem in dataset 2 is the fact that the abdomen center of mass does not coincide with the physical image center in any of the slices. In Figure 3.4 the red dot represents the physical image center and the green dot represents the approximate center of mass of the abdomen.

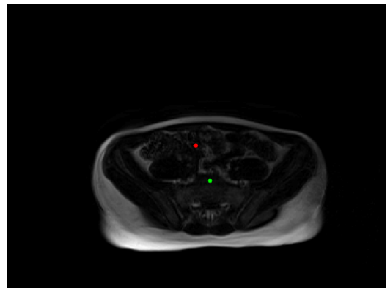
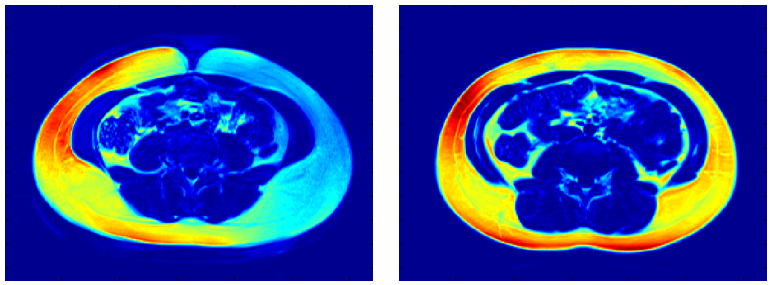


Figure 3.4: Artifacts in Dataset 2

This will prove to be a problem in the SAT interior boundary detection described in Chapter 6.

Other artifacts found in most of the datasets are slight ghosting artifacts (due to respiratory motion) and spatial intensity inhomogeneity (also known as bias field) caused by inhomogeneity of the B_0 magnetic field in the first 3-4 slices of the datasets as seen in Figure 3.5(a).



(a) Dataset 13,Slice 1 (b) Dataset 13,Slice 9

Figure 3.5: Visualization of two slices from Dataset 13 - Figure 3.5(a) affected by bias field on the right-hand side of the image and Figure 3.5(b) with less bias field

No preprocessing of the data has been done for removing the bias field from the first slices of the datasets. A better quality of the data with respect to spatial intensity homogeneity could be obtained using a better DIXON reconstruction technique.

From the images acquired through the DIXON technique, the fat images were used both in the SAT and VAT segmentation. In addition for the VAT segmentation the water images have been used for computing the fat ratio image.

Chapter 4

Graph Cuts in Image Segmentation

Graph cuts are used extensively in image processing and computer vision applications where the problem at hand can be formulated in the form of energy minimization. Solving an energy minimization type of problem consists of setting up a graph of the function that needs to be minimized. Furthermore finding a Maximum-Flow/Minimum-Cut in the setup graph is equivalent with the function minimization. Graph cut methods have been used for image restoration, stereo vision and segmentation.

The following chapter will describe the use of graph-cut based methods for image segmentation. In Section 4.1 the theoretical aspects behind graph-cuts are explained, followed by a description of the algorithm used to compute the graph-partition - a min-cut/max-flow algorithm implementation by Boykov and Kolmogorov [3] in 4.2. Section 4.3 explains the graph set up for optimal surface detection problem.

4.1 Theoretical aspects

In [8] Greig et al. describe the first use of Max-Flow/Min-Cut algorithms for finding a binary labeling that is globally optimal. To find the binary labeling the energy framework has been used for finding an estimate of the maximum a posteriori probability of a generalized Potts model Markov Random Field.

$$E(L) = \sum D_p(L_p) + \sum V_{p,q}(L_p, L_q) \tag{4.1}$$

In the above formula L is a labeling of the image P, D_p is a data penalty function,$V_{p,q}$ is an interaction potential and N is a neighborhood of pairs of neighboring pixels in the image P. The data penalty indicate a likelihood function and the interaction potentials force spatial coherence.

A graph $G = (V, E)$ consists of set of nodes or vertices and a set of edges, usually denoted V and E, that connect the nodes in V. In image analysis the nodes in V can correspond to pixels, voxels or features.

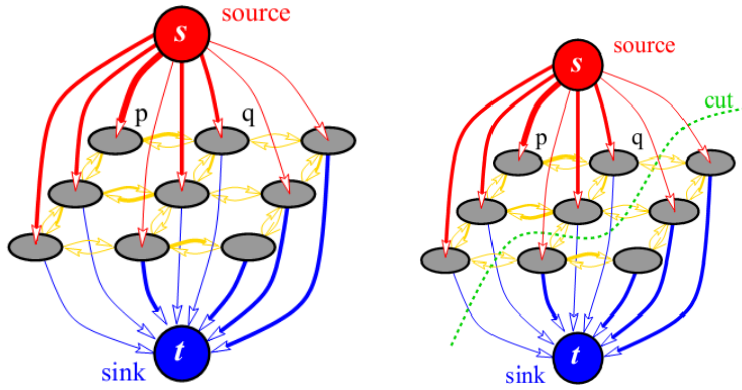
A graph cut is a partitioning of the set of vertices of a graph into two disjoint subsets also known as s-t cut (source-sink separation). To perform an s-t cut, to the graph vertices V in $G = (V, E)$ are added two additional nodes, called terminal nodes: s - source node and t - sink node. In image segmentation, these and represent the labels that can be assigned to pixels or voxels. The set E of the constructed graph $G_{st} = (V \cup s, t, E)$ is constructed of two types of edges: T-links and N-links. T-links (terminal links) connect the nodes in the graph to the terminal nodes s, t. N-links (neighboring links) connect pairs of neighboring pixels [13].

All the graph edges have associated a cost: in the case of directed graphs the cost of the edge (p,q) is different from the cost of the reverse edge (q,p). The cost function to be minimized is defined in the following manner:

$$E(f) = E_{data}(f) + E_{smooth}(f) \tag{4.2}$$

where E_{data} represents the costs associated to the t-links, and E_{smooth} the cost associated to the n-links.

The T-link edge costs correspond to the penalty associated to assigning the corresponding label to that pixel. The cost of an N-link represents the penalty for not assigning the two neighboring pixels to the same label. An s-t cut partitions the nodes into two subsets, such that no path can be established between source and sink and the cost of a cut is equal to the cost of the edges in the cut.



(a) Graph G (b) A cut on graph G

Figure 4.1: Graph and graph-cut [3]

4.2 Max-Flow/Min-Cut Algorithms

There are various algorithms for finding the Maximum-Flow/Minimum-Cut in a graph. These algorithms can be divided into two major groups: the 'push-relabel' type algorithms (Goldberg-Tarjan style) and 'augmenting paths' algorithms(Fold-Fulkerson style).

In [3], Boykov and Kolmogorov present a practical analysis of different Max-Flow/Min-Cut algorithms used for image analysis and computer vision problems, proving that their algorithm performs faster than others in most of the tests performed. The algorithm used in this project is the algorithm described in [3].

Standard 'augmenting paths' based algorithms work by pushing flow from the source to the sink through non-saturated paths in the graph. A path is considered to be saturated when no flow can be sent from source to sink through that path, i.e. if at least one of the edges is saturated. The flows represented by the cost/weight associated to the edges of the graph.

The information of the flow among the edge of the graph G is kept using a residual graph. The residual graph has exactly the same topology as the graph G, but the capacity of the edge is represented by the residual capacity given the flow amount run through that edge. In each iteration the shortest path from source to sink through non-saturated edges of the residual graph is chosen. When a path is found the maximum possible flow, which saturates at least one

edge on the path is pushed through the path and the residual capacities of the edges on that path are reduced by the pushed flow and the residual capacities of the inverse edges are increased by the flow. The maximum flow is computed by adding the pushed flow value to a total flow value. The algorithm stops when any path between source and sink has at least one edge that is saturated.

An edge is saturated when the weight associated to that edge in the residual graph is equal to 0 (no flow value can be sent through that edge). The minimum cut is found by choosing the saturated edges with the smallest capacity on all the paths from source to sink. Dinic's algorithm is based on this standard approach and uses breadth-first search to find the shortest path from source to sink on the residual graph. This type of algorithm starts the breadth-first search algorithm for paths of length k+1 from scratch, when all paths of length k have been saturated. Doing a breadth-first search involves scanning almost all the nodes of the graph, which in the context of a computer-vision application is an expensive operation if performed too often.

The algorithm developed by Boykov and Kolmogorov belongs to the 'augmenting paths' type of algorithms and it uses search trees for finding augmenting paths, same as Dinic's algorithm. The difference between the two algorithms is that in the Boykov-Kolmogorov approach two search trees are constructed - one starting from the source and one starting from the sink. Another difference consists in the fact that these trees are reused compared to building them from scratch when searching paths of length k+1. The disadvantage of this approach is that the found augmenting paths are not necessarily the shortest ones. The Boykov-Kolmogorov approach performs better in computer vision problems due to the reuse of the search trees.

The two search trees, with roots at the source and at the sink, are denoted by S and T. In tree S all the edges from each parent node to its children nodes are non-saturated. In tree T all the edges from the children nodes to its parent nodes are non-saturated. The nodes that do not belong to either of the trees are called 'free' nodes and in the beginning of the algorithm the intersection between the two search trees is the empty set, tree S containing only the source node and tree T containing only the sink node. The nodes in trees T and S can be denoted either 'active' or 'passive'. Active nodes are allowed to grow by adding children nodes on non-saturated edges. Also active nodes can also come in contact with nodes from the other tree, signaling that an augmenting path has been found. The algorithm consists in repeating in an iterative manner the three steps described below:

1. **Growth**

   In the growth step the trees S and T are expanded. The expansion pro-

cess is done through active nodes exploring adjacent non-saturated edges and acquiring new children from the set of the free nodes. The acquired nodes become active nodes in the corresponding search tree. An active node becomes passive when all the neighbors of a given active node are explored. The growth step stops when an active node finds a neighboring node belonging to the other tree.

The figure below presents the two search trees S and T and the detected augmenting path (yellow line) found. The active and passive nodes are denoted by letter and the free nodes are marked by a black circle.

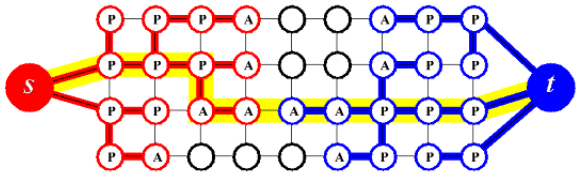


Figure 4.2: Search trees S and T at the end of growth step [3]

2. **Augmentation**

   After finding an augmentation path in the growth step, the path is augmented by pushing the largest flow possible (the minimum capacity of the edges on the path). As some of the edges will become saturated, some of the nodes in the S and T trees may become orphans. A node is orphan if the edges linking these nodes to its parents are saturated. During this step the S and T trees may be split into forests where the orphan nodes behave as roots of the trees in the forest.

3. **Adoption**

   After the augmentation step, the S and T trees might be split into forests. In the adoption step, the orphan nodes, that act as roots for trees in the formed forests will either be adopted to the S or T trees or declared free nodes. In this step a new valid parent is searched for each orphan node.

   The parent has to belong to the same tree as the orphan and should be connected to this through a non-saturated edge. If no parent is found for the orphan node, the node is removed from the tree it belongs to (S or T), being declared a free node. In this case all its former children are also labeled as orphan nodes and need to be submitted to the adoption process.

   The adoption step terminates when all orphan nodes have been either adopted or labeled as free.

   The role of this step is to restore the structure of the S and T trees to single-tree with roots in nodes s and t.

These three steps are iterated, until none of the search trees S and T can grow and the trees are separated by saturated edges only.

4.3 Optimal Surface Segmentation

Most of the problems related to medical images consist in making a quantitative analysis of a region of interest. For this reason, easily finding object boundaries is an important step of the segmentation and quantification process. Examples of this type of problems are estimating the volume occupied by a type of tissue or the shape characteristics of an organ.

The most utilized segmentation tools in medical image analysis are based on utilizing graph-searching principles in 2D images. Because of the predominance of segmentation methods for 2D images, 3D medical images are usually analyzed as a sequence of 2D slices. By doing this, the slice-to-slice contextual information is lost.

In [23] Li et. al. present a method of solving problems such as surface detection and image segmentation in the d-D space (d≥3) in an optimal manner. Wu and Chen's optimal net surface method gives globally optimal solutions and has the ability of detecting multiple surfaces in multiple dimensions through the use of graphs and maximum flow algorithms. The segmentation problem, or the surface detection problem is now transformed to the much simpler problem of finding a minimum cut into the constructed graph.

The main disadvantage of Wu and Chen's method is that the sought surface must be a terrain-like to start with or that it can be transformed into a terrain-like surface.

This requirement is directly related to the problem description as a multi-column graph and the use of the Max-Flow/Min-Cut type of algorithms. Performing a minimum cut in graph consists in separating the nodes in the graph into two groups (nodes belonging to a source and nodes belonging to a sink) as described in Section 4.1. For this reason, one point on the sought surface is associated to one column, a column representing all the possible solutions that the point on the surface can take.

The figure below shows an example of net surface problem in 2D.

The simplest method of transforming the segmentation problem from image

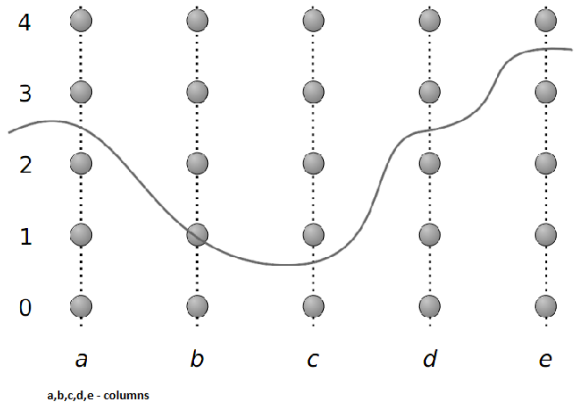


Figure 4.3: Disjoint set of columns and surface representation [16]

space to a column-based graph space is using a polar coordinates transform or by constructing flow lines intersecting with an initial rough segmentation of the structure as described in [17].

In the multi-column graph construction the edges in the graph G_{st} described in Section 4.1 are divided into to categories: intra column edges and inter column edges. As the name describes it, intra column edges are edges between nodes positioned on the same column. The inter column edges are edges between nodes on neighboring columns. These are the edges that enforce smoothness constraints of the sought surface, forcing two point from the sought surface to be on close levels in the columns they belong to.

The following subsection describes the graph setup for the 2D case.

4.3.1 Graph setup - 2D

The first step consists in adding the sink and the source nodes and connecting the first nodes in all the columns to the source and the last nodes in all the columns to the sink, and setting up the cost of these edges to ∞ to avoid a cut right above the source or right below the sink. This is followed by adding the intra column edges and their cost - costs between vertices of the same column known also as vertex cost. Figure 4.4 presents the setting of the intra column costs and costs for connection to source/sink.

By setting the reverse costs to ∞ the cut is forced to be done only once on a

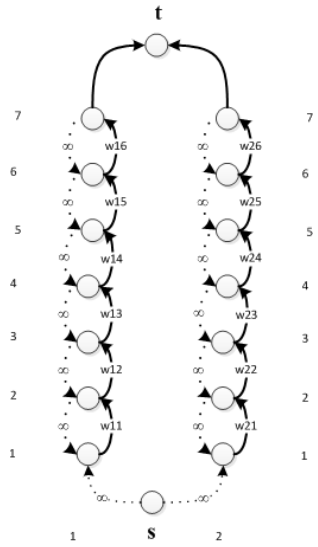


Figure 4.4: Graph Setup 2D - Intra column and Terminal edges

column, making the solution a net surface. A net surface in a graph G is a subset of vertices V, such that each vertex in N belongs to exactly one column in V and vice versa.

Another type of edges that need to be set up are the inter column edges - added between nodes belonging to two adjacent columns, either at the same level in the column or at different levels. This type of edge costs ensures the smoothness of the obtained surface.

In Figure 4.5 part of the intra column edges have been removed for an easier visualization of the inter column edges.

Most often the inter column edges are set up as depicted in Figure 4.5 either between nodes found at the same level on adjacent columns as the edges numbered with 0, or one node above or one node below as the edges numbered by 1, and so on. The cost to be set on these edge is subject to how much local similarity is to be enforced and the constraint types are divided into hard constraints and soft constraints. The hard constraints consist in having set the inter column cost to ∞. For example if setting an ∞ cost on the edges numbered with 1 in Figure 4.5 it is enforced that the cut from one column to the other can differ by at most one index.

In the case of the soft constraints, the cost attached to the inter column edges is

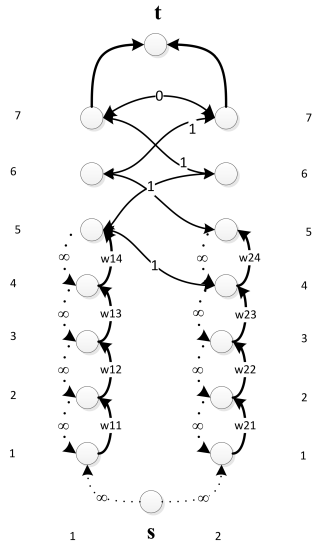


Figure 4.5: Graph Setup 2D - Inter column edges

related to the range of vertex costs on that column and the locality relationship. Nodes around the vertex having the smallest vertex cost in a column will have small node costs as well, so by setting an inter column cost to a value greater than the smallest cost on that column, the cut might be done above or below the vertex with the smallest node cost. An attempt to describe the behavior of the inter column cost on a 2D example is made in Section 4.4.

4.3.2 Graph Construction - 3D

The same approach is used to explain the graph setting for volumetric data. This subsection consists the main theoretical basis of the SAT boundary detection algorithm.

A volumetric image can be seen as a 3D matrix $I(x, y, z)$ and assuming that the sought surface is already represented as a terrain-like surface in this dataset we obtain the surface orientation depicted in Figure 4.6.

In Figure 4.6, the columns are represented by voxels having the same x and y coordinates and different z-coordinates, the net surface being described as $N : (x, y) \rightarrow N(x, y)$, where $x \in \mathbf{x} = \{0, ..., X - 1\}, y \in \mathbf{y} = \{0, ..., Y - 1\}$ and $N(x, y) \in \mathbf{z} = \{0, ..., Z - 1\}$. The notations and formulas for the 3D graph

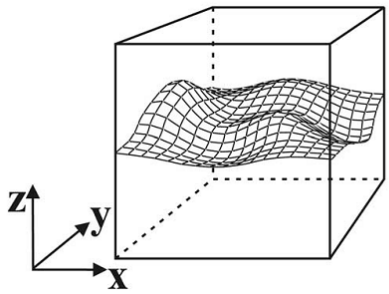


Figure 4.6: Surface orientation [13]

setting are the same as the ones described in [13].

For the 3D case the inter column and intra column edge costs are set in the following manner, as described in [13]:

1. **Intra column arcs**

$$E_{data} = \langle C(x,y,z), C(x,y,z-1)\rangle | z > 0 \tag{4.3}$$

where $C(x,y,z)$ is defined as the difference between the vertex cost of the endpoints of the edge as defined in Equation 4.4.

$$C(x,y,z) = \begin{cases} w(x,y,z), if z = 0 \\ w(x,y,z) - w(x,y,z-1), otherwise \end{cases} \tag{4.4}$$

The intra column cost of a voxel is inverse proportional to the likelihood of that voxel belonging to the sought surface, such that the optimal surface has the minimum cost over all the feasible surfaces.

In [13] a surface is defined as being feasible if it satisfies the smoothness constraint, which means that if we have two voxels on the feasible surface denoted by $I(x,y,z)$ and $I(x+1,y,z')$, then the difference between $|z - z^{'}| \leq \Delta_x$, and similarly for Δ_y.

2. **Intercolumn arcs**

For defining the inter column arcs we must refer to nodes belonging to two adjacent columns : e.g. $Col(x,y)$ and $Col(x+1,y)$ (in a similar manner two other adjacent columns would have been $Col(x,y)$ and $Col(x,y+1)$).

The adjacency of two columns is defined by a neighborhood relationship between points in an xy plane, such as a 4-neighborhood (N_4) or

8-neighborhood (N_8). The simplest case of the N_4 neighborhood will be considered in what follows. The cost set up on the inter column arcs is presented in Equation 4.5.

$$E_{smoothness} = \begin{cases} \langle Col(x,y,z), Col(x+1,y,max(0,z-\Delta_x))\rangle | \\ \qquad x \in [0,X-2], y \in [0,Y], z \in [0,Z] \\ \langle Col(x,y,z), Col(x-1,y,max(0,z-\Delta_x))\rangle | \\ \qquad x \in [1,X-1], y \in [0,Y], z \in [0,Z] \\ \langle Col(x,y,z), Col(x,y+1,max(0,z-\Delta_y))\rangle | \\ \qquad x \in [0,X], y \in [0,Y-2], z \in [0,Z] \\ \langle Col(x,y,z), Col(x,y-1,max(0,z-\Delta_y))\rangle | \\ \qquad x \in [0,X], y \in [1,Y-1], z \in [0,Z] \end{cases} \tag{4.5}$$

If in the original coordinate system the sought surface is cylindrical, it is needed to wrap around the surface in the transformed space along the unfolding axis (x or y). This signifies that smoothness constraints are added for the first and last row along the axis for which the wrap around is needed. For example for a x-wraparound, each node $Col(0,y,z)$ is connected to $Col(x-1,y,z-\Delta_x)$ and each node $Col(x-1,y,z)$ is connected to $Col(0,y,z-\Delta_x)$, with the associated costs.

This is applicable to the data in this project, as the abdomen volume viewed in 3D has a cylindrical shape. The Δ_x and Δ_y values represent the stiffness of the sought surface. Higher values set for these variables let the surface vary more from one column to the other.

4.4 2D Graph and cost setting example

In order to better exemplify the role of the inter column costs (smoothness cost) two simple examples have been designed. These examples provide some insight into the relationship between edge costs and number of edges needed and the results obtained, making it easier to tune these parameters in the case of a more complex application.

Figure 4.7 depicts the 2D graph setting that will be used as an example. This graph is constructed of 2 columns, denoted by A and B, each containing 5 nodes. The two extra nodes s' and t' represent the fake source and sink nodes added such that the connections between the source/sink to the start and terminal nodes of the two columns could be treated as non-terminal edges, imposing costs in both directions. As explained in Section 4.3, the costs of the edges from sink towards source are set to ∞ such that each column is cut only once.

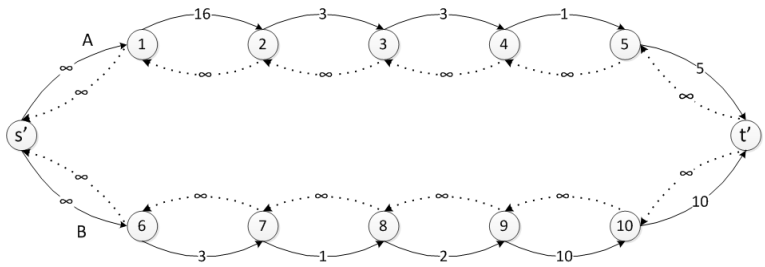


Figure 4.7: 2D graph example

When only using intra column edge costs the obtained cut looks as red line in Figure 4.8 with Max.Flow = 2.

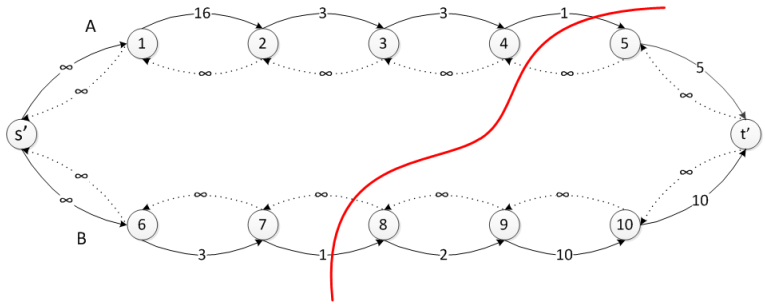


Figure 4.8: Cut obtained using only intra column edge weights

If a smoother cut is desired, a cost could be imposed along edges connecting the two columns. Several test have been performed for finding the minimum

number of inter column edges needed in order to obtain the same smoothness effect. The first test performed was using a displacement between the columns (Δ_x of 0) with a graph setting as depicted in Figure 4.9

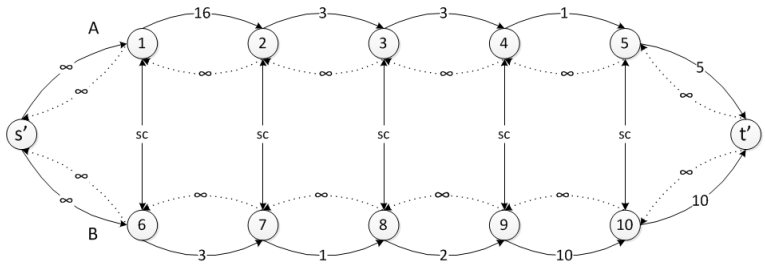


Figure 4.9: Graph setting with Δ_x=0

For a smoothness constraint smaller or equal than the minimum value of the minimum edge costs of the two columns ($cx \leq 1$) the same result is obtained as depicted in Figure 4.10 with an increased Max.Flow = 4, due to the additional inter column edges that need to be saturated.

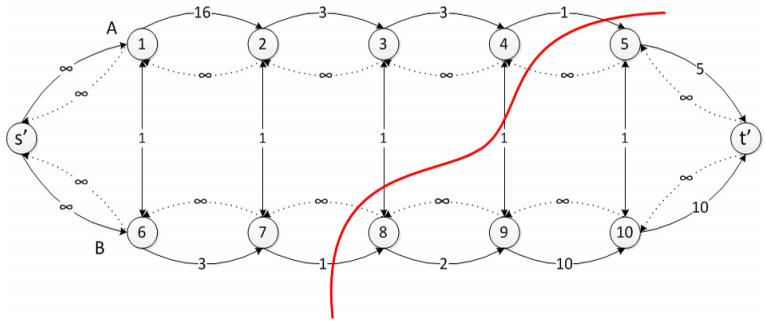


Figure 4.10: Cut obtained using cx=1 and Δ_x=0

When the cost of the intra column edges is greater the minimum value of the minimum edge costs of the two columns ($cx \geq 1$) the cut is forced to be made at the same level as depicted in Figure 4.11 with a Max.Flow = 4.

The next test performed was using a Δ_x value different than 0, such that the cut on column B is not lower than Δ_x nodes with respect to the cut on the column A. As the number of nodes in the test graph is fairly small a values of 1 has been chosen for Δ_x. Similarly edges can be added to enforce that the cut on the column B is not Δ_x nodes higher than the cut on the column A.

Figure 4.12 depicts a graph setting where the cut in column B can not be more

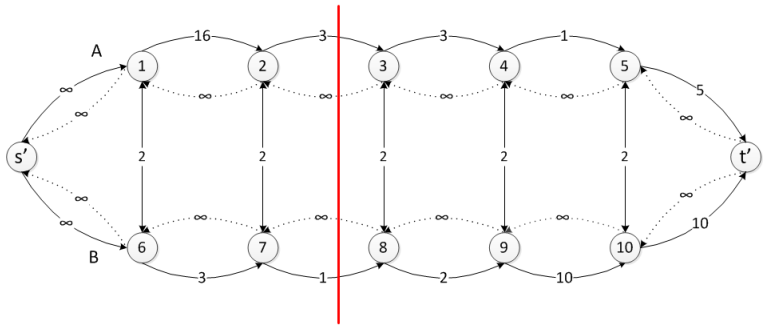


Figure 4.11: Cut obtained using cx=2 and Δ_x=0

than one node lower than the cut in column A.

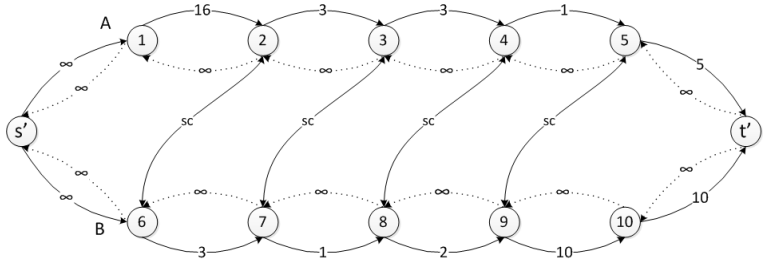


Figure 4.12: Graph setting Δ_x=1

In Figure 4.13 the cut obtained when using a smoothness constraint > 1 is presented.

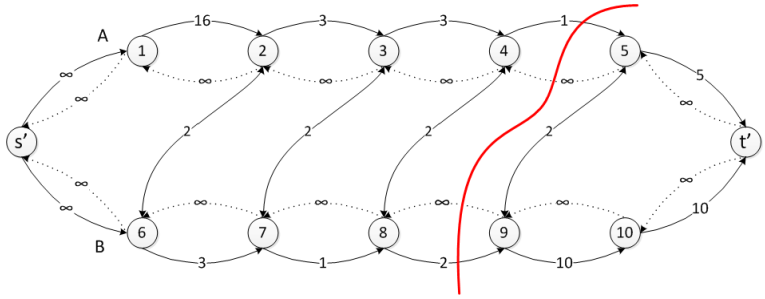


Figure 4.13: Cut obtained using Δ_x=1 and cx=2

The effect of having smoothness constraints on inter column edges with respect to both $+\Delta_x$ and $-\Delta_x$ gives similar results to a setting where having only $+\Delta_x$

and a higher smoothness cost value. When using a difference of $+\Delta_x$ or $-\Delta_x$ between adjacent columns it has been observed that a smoothness cost value smaller than in the case of having Δ_x value of 0 should be used.

The increase in execution time is influenced both by the smoothness constraint and the number of inter column edges added.

Chapter 5

Clustering

Clustering is the most common approach to performing unsupervised learning, i.e. finding structure in input data for which no label data is available. The purpose of clustering is to find groups of similar data points in a multi-dimensional space. Clustering is a method widely used in image segmentation. Amongst the different types of algorithms used for clustering two common choices are the k-means algorithm and clustering based on statistic distributions, also known as distribution models.

5.1 K-means

The k-means algorithm is one of the most popular algorithms used for clustering. This type of algorithm assumes that the data points are of the quantitative type and that distances between points can be computed. For a dataset of p-dimensional points, the most typical choice for inter-point distances is the squared Euclidean distance.

$$d(x_i, x_i') = \sum_{j=1}^{p} (x_{ij} - x_{ij}')^2 \tag{5.1}$$

The center of each cluster is represented by the mean value of the points belonging to the cluster in all dimensions - $\bar{x}_k = (\bar{x}_{1k}, ..., \bar{x}_{pk})$. The data points are associated to the cluster to which the average dissimilarity of the data point to the cluster center (cluster mean) is minimal. A description of the k-means algorithm is presented below:

Given that k (the number of clusters) is known the algorithm begins by initializing the cluster means.

1. Each observation is assigned to the closest cluster (the cluster for which the distance from the observation point to the cluster center is minimal)

2. After all the samples have been assigned to a cluster the cluster means are recomputed and continue with 1 until there is no change between cluster center computed in the current and previous iterations

The computational complexity of the k-means algorithm is O(NpkT), where N is the number of observations, p the number of features (the number of dimensions of each data point), k are the number of clusters and T the maximum number of iterations. Although the convergence of the algorithm is ensured, the result obtained can be a local minimum as the algorithm is an iterative descent type of algorithm. Often, the k-means algorithm is used for initializing the parameters in a distribution models type of clustering.

5.2 Distribution models

Another widely used type of clustering is the one based on Gaussian mixture models. In the Gaussian mixture model approach, each cluster is defined by a normal distribution characterized by mean and variance $N(\mu, \Sigma)$.

The parameters of the Gaussian distributions are unknown to start with, and are usually determined through maximum likelihood using the Expectation-Maximization (EM) algorithm based on the observed data. The number of clusters, and hence the number of Gaussian mixtures can be either specified or is to be automatically extracted from the data.

In what follows a short description of the normal distribution and mixture of Gaussian will be given, followed by the description of the EM algorithm and how clusters are extracted.

5.2.1 The Gaussian Distribution

The Gaussian Distribution, widely known as the normal distribution is a distribution model for continuous variables, usually denoted as $\mathcal{N}(x;\mu,\sigma^2)$ ($\mathcal{N}(\mathbf{X};\mu,\mathbf{\Sigma})$ in the P-dimensional case). In the 1D case μ represents the mean and σ^2 the variance, also known as the spread and the Gaussian distribution is expressed by the formula depicted in Equation 5.2.

$$\mathcal{N}(x;\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} \tag{5.2}$$

$$\mathcal{N}(\mathbf{X};\mu,\mathbf{\Sigma}) = \frac{1}{(2\pi)^{\frac{P}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(X-\mu^T)\mathbf{\Sigma}^{-1}(X-\mu)} \tag{5.3}$$

In the P-dimensional case μ is a vector of dimension P, $\mathbf{\Sigma}$ is a PxP covariance matrix and $|\mathbf{\Sigma}|$ represents the determinant of $\mathbf{\Sigma}$.

5.2.2 Mixture of Gaussians

A mixture of Gaussians is the linear combination of two or more Gaussian distributions. The formula describing a mixture of Gaussians is depicted in Equation5.4, where each Gaussian $\mathcal{N}(x;\mu_k,\Sigma_k)$ is a component of the mixture, having its own mean and covariance.

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x;\mu_k,\Sigma_k) \tag{5.4}$$

By integrating Equation 5.4 over x and the fact that the integral of a Gaussian function is equal to one, it can be deduced that $\sum_{k=1}^{K} \pi_k$ is be equal to 1. From $p(x) \geq 0$ and $\mathcal{N}(x;\mu_k,\Sigma_k) \geq 0$ it can be concluded that π_k is also ≥ 0.

Equation 5.4 can also be written as:

$$p(x) = \sum_{k=1}^{K} p(k)p(x|k) \tag{5.5}$$

where π_k can be seen as the prior probability of choosing the k-th Gaussian component ($p(k)$) and the $\mathcal{N}(x;\mu_k,\Sigma_k)$ represents the conditional probability of x with respect to k, or the likelihood of x belonging to the k-th component($p(x|k)$).

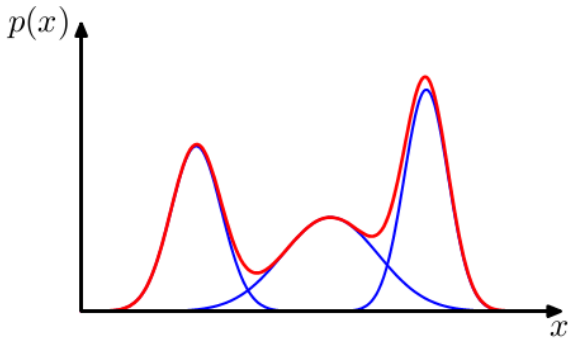


Figure 5.1: Example of a mixture of 3 Gaussians [2]

5.2.3 Maximum Likelihood Estimation

The parameters μ and σ^2 of a normal distribution can be estimated through the maximum likelihood estimation procedure, that will be explained in what follows.

Given a data set $X = (x_1, ..., x_N)^T$ and making the assumption that the observations x_n are drawn from a multivariate Gaussian distribution, the log-likelihood is written as described in Equation 5.6.

$$\ln \mathcal{N}(X; \mu_k, \Sigma_k) = -\frac{ND}{2} \ln 2\pi - \frac{N}{2} \ln |\Sigma| - \frac{1}{2} \sum_{n=1}^{N} (x_{\{}n\} - \mu)^T \Sigma^{-1} (x_{\{}n\} - \mu) \tag{5.6}$$

In order to find the maximum-likelihood estimators of the mean and variance parameters, Equation 5.6 is derived with respect to μ and subsequently with respect to Σ and the derivatives are set to 0.

The maximum-likelihood estimator of the mean is equal to $\mu_{MLE} = \frac{1}{N} \sum_{n=1}^{N} x_n$, the mean of the data points observed.

The maximum-likelihood estimator of the variance is described in Equation 5.7

$$\Sigma_{MLE} = \frac{1}{N} \sum_{n=1}^{N} (x_n - \mu_{MLE})(x_n - \mu_{MLE})^T \tag{5.7}$$

The MLE estimator of the variance is a biased estimate of the true variance of

the distribution. The reader is referred to [2] for more information regarding the maximum-likelihood estimation of the parameters of a distribution.

In the case of the mixture of Gaussians a new group of unknown parameters appears : $\pi = \{\pi_1, ..., \pi_K\}$, one π value for each of the Gaussian components in the mixture. The set of parameters involved in the mixture of Gaussians distribution is represented by $\pi = \{\pi_1, ..., \pi_K\}$, $\mu = \{\mu_1, ..., \mu_K\}$ and $\Sigma = \{\Sigma_1, .., \Sigma_K\}$. The log-likelihood function for the mixture of Gaussians distribution is depicted in Equation 5.8.

$$\ln p(X|\pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n; \mu_{\mathbf{k}}, \mathbf{\Sigma_k}) \tag{5.8}$$

As maximizing the log of the likelihood function for the mixture of Gaussians model (Equation 5.8) is more complex then the multivariate Gaussian distribution case, the Expectation Maximization(EM) algorithm provides a better method for finding the maximum likelihood estimator of the parameters described above. The EM algorithm is briefly described in Subsection 5.2.4.

5.2.4 The Expectation-Maximization (EM) Algorithm

The EM algorithm is an iterative algorithm alternating between two steps until convergence is reached, having the goal to maximize the likelihood function with respect to the parameters of the Gaussian mixture model.

The steps involved in the EM algorithm are:

1. The first step in applying the EM algorithm is the initialization of the parameters: μ_k, Σ_k and π_k and evaluate the log likelihood function described in Equation 5.8 for these values.

2. **Expectation Step(E)**
   In the E step the posterior probabilities are estimated for the current parameter values. As described in [2] the formulas for estimating the posterior probabilities can be calculated as described below.

   From Bayes' theorem it results that the posterior probabilies $p(k|x)$ can be calculated as follows:

$$p(k|x) = \frac{p(k)p(x|k)}{p(x)} \tag{5.9}$$

Making use Equation 5.4 and Equation 5.5 we can write the posterior probability formula depending on mixing coefficients and Gaussian components of the mixture.

$$p(k|x_n) = \frac{\pi_k \mathcal{N}(x_n; \mu_\mathbf{k}, \mathbf{\Sigma_k})}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_n; \mu_\mathbf{j}, \mathbf{\Sigma_j})} \tag{5.10}$$

3. **Maximization Step(M)**
   In the M step the posterior probabilities obtained in the expectation step are used to re-estimate the means, covariances and mixing coefficients for the mixture components. As in the maximum-likelihood estimation of the mean and variances for the multivariate Gaussian, the means are the first ones to be re-estimated, these being later used in re-estimating the covariances.

$$\begin{aligned}
\mu_k^{i+1} &= \frac{1}{N} \sum_{n=1}^{N} p(k|x_n) x_n \\
\Sigma_k^{i+1} &= \frac{1}{N} \sum_{n=1}^{N} p(k|x_n)(x_n - \mu_k^{i+1})(x_n - \mu_k^{i+1})^T \\
\pi_k^{i+1} &= \frac{N_k}{N} \\
N_k &= \sum_{n=1}^{N} p(k|x_n)
\end{aligned} \tag{5.11}$$

4. The last step in the EM algorithm consists in re-evaluating the log-likelihood function described in Equation 5.8 using the parameter values estimate in the maximization step and checking for covergence. Checking for convergence can be checked either on the values of the parameters or on the value of the log-likelihood function. If the convergence test is not passed, the algorithm is continued with step 2.

Amongst the differences between the K-means and EM algorithm is that K-means assigns a data point to only one cluster performing a so called hard assignment of the data points, as to which the EM algorithm assigns a data point to multiple clusters, based on the posterior probability values. The EM algorithm takes more iterations and each iteration requires more computations when compared to K-means. A speed-up of the EM algorithm can be achieved by finding a good initialization for the Gaussians in the mixture model.

Often the K-means algorithm is used for finding a good initialization for the means, covariances and mixing coefficient parameters whose values are to be

maximized through the EM algorithm.The means of the Gaussians are initialized with the centers of the clusters found, the covariances are set to the covariance calculated on the samples belonging to each cluster found and the mixing coefficients are set to the ratio between the number of data points assigned to a cluster and the total number of data points.

Chapter 6

Implementation of SAT segmentation

This chapter describes the implementation of the graph-cut algorithm described in Section 4.3 of Chapter 4 for the purpose of detecting the abdomen boundary and the SAT interior boundary from the volumetric MRI datasets.

In Section 6.1 the polar resampling of the input dataset is described, the result of which will be used in the detection of both surfaces. For each of the detected surfaces the cost calculation method and optimization of graph setup parameters are presented in Sections 6.3 and 6.4.

6.1 Polar Resampling

As described in Section 4.3 the sought surface must have a terrain-like shape. An easy transformation of the cylinder-like surfaces of the abdomen boundary and interior SAT boundary into a terrain-like surface is through a polar coordinates transform applied to each slice in the 3D dataset or through a cylindrical coordinates transform on the entire dataset. The cylindrical coordinates transform is merely a generalization of the 2D polar coordinates transform, the only difference being the addition of the z-axis coordinates.

For this project a 2D polar coordinates transform is implemented. For optimizing the implementation of the coordinates transform, the mapping between Cartesian coordinates and polar coordinates has been calculated only once and reused for all the slices.

The polar resampling is based on the transform between polar coordinates and Cartesian coordinates as described in Equation 6.1 where r is the radial distance from the origin and θ is the counterclockwise angle from the x-axis as described in Figure 6.1.

$$\begin{aligned} x &= r\cos\theta \\ y &= r\sin\theta \end{aligned} \tag{6.1}$$

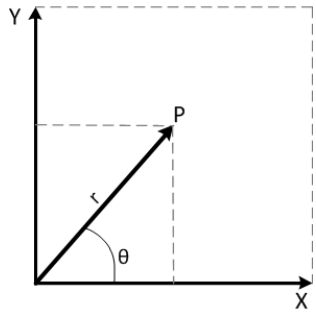


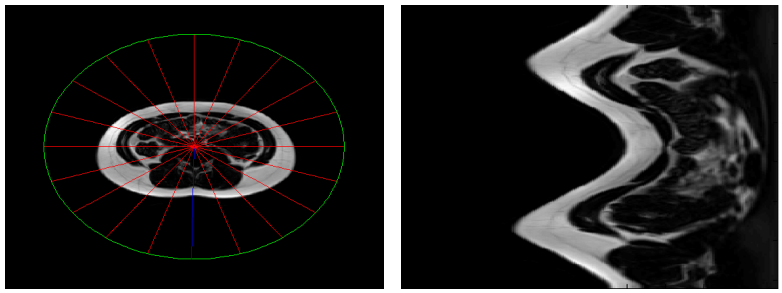
Figure 6.1: Mapping from two-dimensional Cartesian coordinates to polar coordinates

The polar coordinates sampling scheme is created by varying r between $[0, R]$ and θ between $[0, 360°)$ or $(-180°, 180°)$ (or in radians: $[0, 2\pi)$ or $(-\pi, \pi]$). The pixel intensities for each spoke (θ angle) are found through bilinear interpolation of the pixel intensities at Cartesian locations calculated by varying r and θ, as the sampling location does not match the pixel center.

The polar coordinates transform is dependent upon three parameters r, θ and the circle center. The circle center for the polar coordinates transform is chosen as being the center of the 3D dataset in x and y direction, or it can be manually selected from first slice of the volume.

The manual selection option is introduced because the abdomen center of mass for dataset 2 differs very much from the image center. Doing a polar coordinates transform with the off-center dataset proves to be a problem in the SAT interior boundary detection, caused by the optimizations performed on this method.

First elements in each line correspond to pixels further away from the image center. Figure 6.2 presents a plot of the evenly distributed spokes and an example of a transformed image.

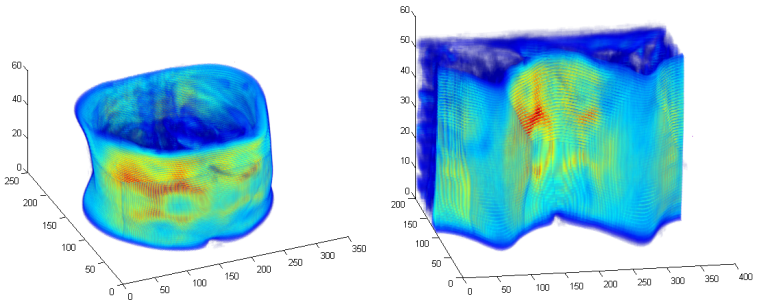


(a) Polar transform sampling (b) Result of the polar transform

Figure 6.2: Polar transform - Dataset 13, Slice 9

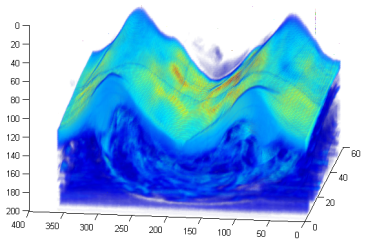
6.2 Node connections and Intercolumn Weights

After performing the polar transform the dataset is represented by $V(r, \theta, z)$ where z represents the slice index. At this point the sought surface is represented as a terrain-like surface. In order to have the same surface orientation as described in Figure 4.6 the dimensions of the transformed dataset must be rearranged to respect the following coordinates arrangement $V(\theta, z, r)$. Figure 6.3 presents 3D plots of the original dataset and of the polar coordinates transformed and re-arranged volumes. As described in Section 4.3 the intercolumn



(a) Original MR dataset $V(x, y, z)$

(b) Dataset in polar coordinates $V(r, \theta, z)$



(c) Dataset in rearranged polar coordinates $V(\theta, z, r)$

Figure 6.3: 3D Views of Dataset 13

edge weights are set up for adjacent columns, where the adjacent columns are described by a neighborhood system. The neighborhood system appropriate for finding the abdomen outer boundary and the SAT interior boundary was considered to be the N_4 neighborhood system as depicted in Figure 6.4.

Having performed the coordinates rearrangement, the x,y,z coordinates in the intra column edges formulas described in Equation 4.5 are replaced as follows: $x = \theta$, $y = z =$ number of slices and $z = r$. In what follows when referring to a

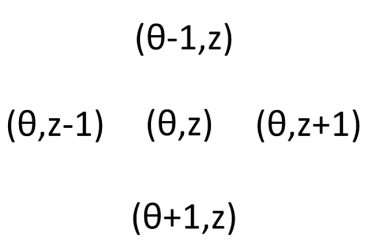


Figure 6.4: 4-Neighborhood

column the notation using the new coordinates system will be used (e.g. when referring to column (x,y) the notation (θ, z) will be used).

Several tests are performed using inter column weights only between (θ, z, r) and $(\theta + 1, z, r - \Delta_\theta)$ columns, between $\langle(\theta, z, r), (\theta - 1, z, r - \Delta_\theta)\rangle$ and $\langle(\theta, z), (\theta - 1, z, r-\Delta_\theta)\rangle$, to which later were added inter column weights between $\langle(\theta, z, r), (\theta, z, r-\Delta_z)\rangle$.

The relationship between (θ, z, r) and $(\theta \pm 1, z, r \pm \Delta_\theta)$ columns in the resampled coordinates system corresponds to changes in the (x,y) direction for the same slice z between two feasible boundary points. So by imposing a smoothness constraint in θ direction the locality relationship of the boundary points in the xy-plane of the Cartesian coordinate system is constrained.

On the other hand, imposing a smoothness constraint between (θ, z, r) and $(\theta, z \pm 1, r \pm \Delta_z)$ columns enforces the smoothness of the boundary detected between adjacent slices, assuming that the difference in boundary location between two adjacent slices is not greater than the Δ_z value. For all the tests performed the execution time over the graph setting and graph-cut algorithm is registered for comparison.

The inter column edge weights are set to a constant between nodes in θ direction and the cost is referred to as $c\theta$. Similarly the inter column weights in z-direction are set to a constant value and are denoted by cz. Specifying whether two nodes in adjacent columns are connected by an inter column edge at a certain level is controlled through Δ_θ for the θ direction and Δ_z for the z-direction.

6.3 Abdomen Boundary Detection

6.3.1 Intra column weights

The cost used for the intra column edges, making up for the E_{data} element in Equation 4.2 has been derived from the gradient function. The gradient has been applied on the 3D polar transformed dataset. As observed in Figure 6.5 the outer abdomen boundary has a high gradient response.

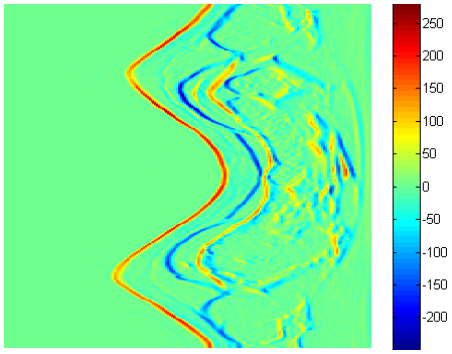


Figure 6.5: Gradient image of a slice : Dataset 13, Slice 9

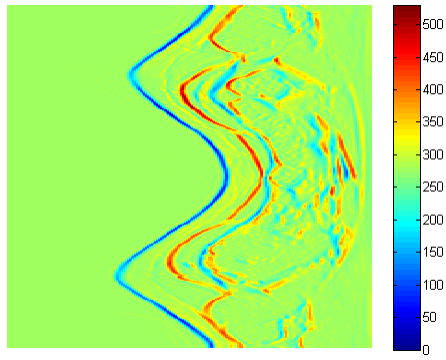
Having a cut between two neighboring pixels on a column is equivalent to having a saturated edge connecting the pixels in the graph representation of the dataset. An edge is likely to be saturated if the edge's maximum capacity is the lowest on a path from source to sink. Because of this the gradient function of the polar image must be transformed such that a point on the abdomen boundary has a positive minimum cost value on every column.

Equation 6.2 presents the formula describing the cost function of every slice in the MRI volume and Figure 6.6 presents an example of the cost function.

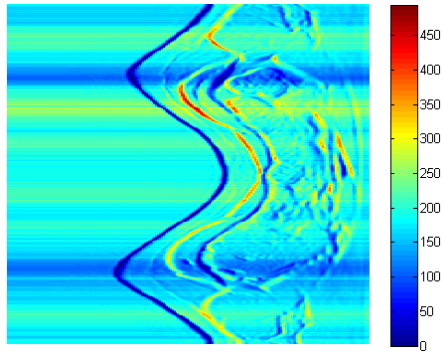
$$C(\theta, r) = max(\frac{\partial I}{\partial \theta \partial r}) - \frac{\partial I}{\partial \theta \partial r} \tag{6.2}$$

In computing the cost function initially the normalization was done using a per-slice maximum value, but it was later observed that the graph-cut algorithm is faster if a normalization with the maximum per slice and θ dimension is used instead (Equation 6.3), as lower values for the intra column edge weights are used. Figure 6.6 presents the vertex cost for one slice using both normalization methods.

$$C(\theta, r) = max(\frac{\partial I}{\partial \theta \partial r}(\theta)) - \frac{\partial I}{\partial \theta \partial r}(\theta) \tag{6.3}$$



(a) Gradient normalization with maximum per slice



(b) Gradient normalization with maximum per slice and θ direction

Figure 6.6: Abdomen boundary - Intra column cost for one slice : Dataset 13, Slice 9

6.3.2 Parameter tunning - Abdomen Boundary Detection

1. **Test** #1

   The first set of performed tests uses the inter column edges only in θ direction as depicted in Figure 6.7.

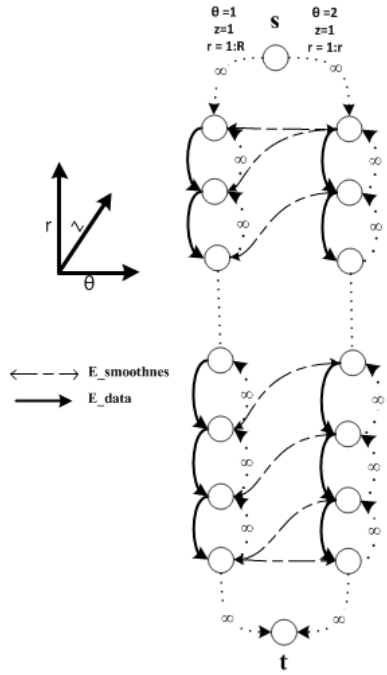


Figure 6.7: Intercolumn Edge Weights Test #1

The edge weights from the added source node to the first nodes in all the columns and the ones from the last nodes in the columns to the added sink node have been set to ∞ in order to avoid obtaining a cut through these edges.

Different values have been tested for the inter column edge cost and the results obtained for all the datasets have been visually inspected. In what follows results from all the dataset are presented for the problematic slices in the datasets, explaining the behavior of the graph-cut algorithm with respect to the used setting.

The set of values used for the $c\theta$ parameter is $\{15, 25\}$. For better explaining how the values of the inter column edge weights are selected, the intra column cost values computed in Section 6.3.1 for a problematic slice in dataset 13 is presented below.

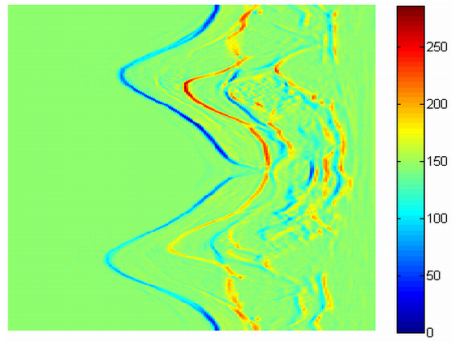


Figure 6.8: Intra column Edge Weights - Dataset 13, Slice 2

As it can be seen in Figure 6.8 around $\theta \in [179, 191]$ the response of the abdomen boundary has a much greater cost value than expected, due to boundary orientation and some bias field present in the MRI image (see Figure 6.9).

For this reason we need a sufficiently high cost for the inter column edges such that the boundary is not allowed to have a jump greater than 1 pixel between adjacent columns, but in the same time not to put to much constraint on the regions with acceptable intra column cost values.

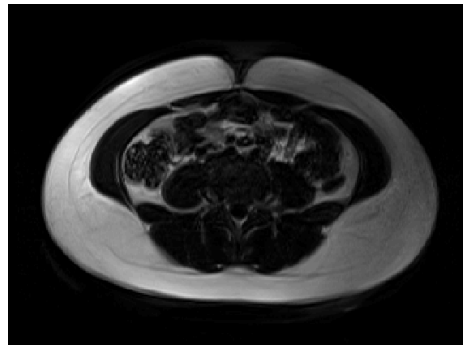


Figure 6.9: Dataset 13, Slice 2

Test #1 with $c\theta = 15$

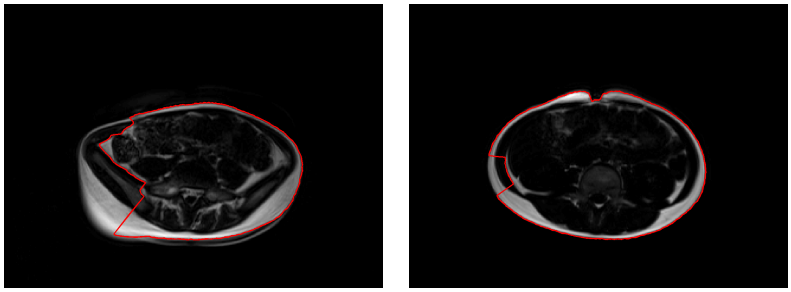
Table 6.1 describes the errors in each dataset when using $c\theta = 15$ and the reason for this errors.

Table 6.1: Abdomen Boundary Detection - Errors for Test #1, $c\theta$=15

Dataset no.	Slice no.	Observations
1	1-3,19,20	Errors in slices 19 and 20 are caused by a too small smoothness constraint cost Errors in slices 1-3 are caused either by a too small smoothness constraint cost, or a too simple graph construction
3	16-19	Errors in slices 16-19 are caused by a too small smoothness constraint cost
4	28,48	Errors in slices 28 and 48 are caused by a too small smoothness constraint cost
6	11-14,17	Errors in slices 11-14 and 17 are caused by a too small smoothness constraint cost

Figures below contain 1-2 slices for each of the datasets described in Table 6.1.

Dataset #1



(a) Slice 2 (b) Slice 19

Figure 6.10: Test #1, $c\theta = 15$ - Dataset 1

Dataset #3

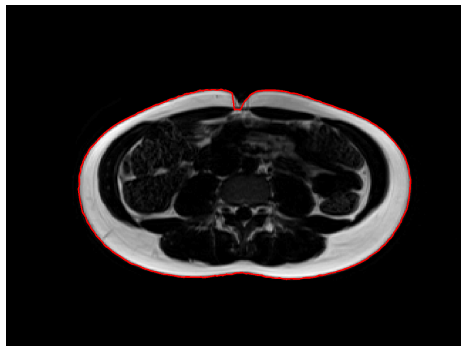


Figure 6.11: Test #1, $c\theta = 15$ - Dataset 3, Slice 19

Dataset #4

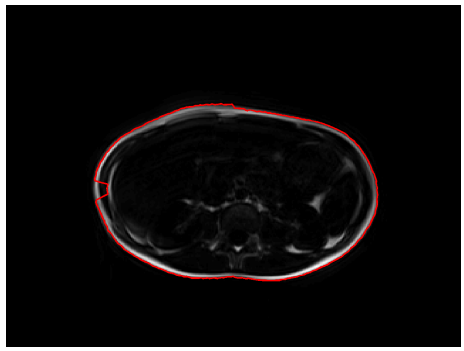


Figure 6.12: Test #1, $c\theta = 15$ - Dataset 4, Slice 28

Dataset #6

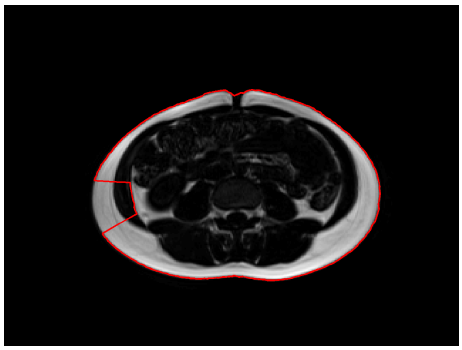


Figure 6.13: Test #1, $c\theta = 15$ - Dataset 6, Slice 17

As described in Table 6.1, errors as the ones in Figures 6.10(b), 6.11, 6.12 and 6.13,are caused by having set a too low value for the inter column edge weights, which is the cause of most of the errors for this graph setting.

Test #1 with $c\theta = 25$

Table 6.2 describes the errors in each dataset when using $c\theta = 25$ and the reason for this errors.

Table 6.2: Abdomen Boundary Detection - Errors for Test #1, $c\theta$=25

Dataset no.	Slice no.	Observations
1	1-3	Errors in slices 1-3 are caused by a too simple graph construction
3	16-18	Errors in slices 16-18 are caused by a too simple graph construction
4	48	Errors in slice 48 are caused by a too small smoothness constraint cost
6	11-12, 17	Errors in slices 11-12 and 17 are caused by a too small smoothness constraint cost

Most of the errors that have been registered when using the graph setting from Figure 6.7 with a $c\theta$ cost of 25 are similar to those obtained when using a $c\theta$ cost of 15.

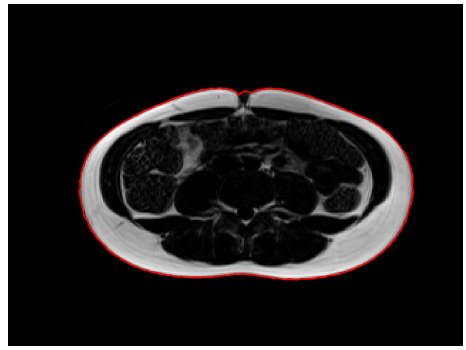


Figure 6.14: Test #1, $c\theta = 25$ - Dataset 3, Slice 16

In the case of dataset #1, slice 2, the errors have not been resolved by increasing the inter column edge weights value from 15 to 25. This error is attempted to be solved with Test #2 and Test #3.

The errors in dataset #3 for slices 17-18 are caused by having a too low smoothness prior cost, and the error found in slice 16 is caused by having a too high smoothness prior cost for that slice. The error in slice 16 of dataset #3 (Figure 6.14), indicates that a too simple graph setting and a too high cost value are used. Decreasing the cost value smaller than 10 would cause problems in other slices of the dataset.

For Test #1, the best results have been obtained using $c\theta = 25$. Fixing the errors caused by having a too simple graph setting can be done either by using more edges in θ-direction or by using additional smoothness constraints in z-direction. These two options will be investigated in Test #2 and Test#3.

2. **Test** #2

   This set of tests have been performed using an added set of inter column edges in the θ-direction as described in Figure 6.15 with different cost values.

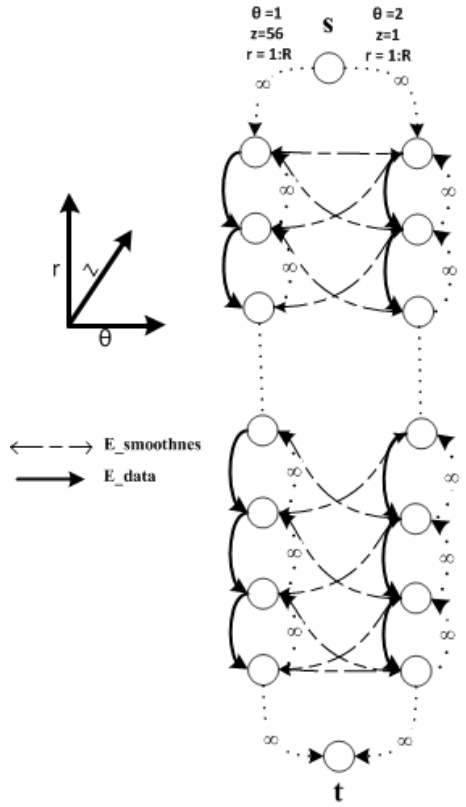


Figure 6.15: Inter column Edge Weights Test #2

As seen in Section 4.4, in the simple 2D example, when increasing the complexity of the graph construction, the smoothness constraint cost must be decreased. For this reason, testing the inter column edge setting described in Figure 6.15 has been done using an edge weight value of 15. The value of the smoothness constraint cost found to be optimal in the case of Test #1 was 25.

The type of errors depicted in Figure 6.10(a) and Figure 6.13, which could not be solved using any of the smoothness constraint cost values from Test #1, have been resolved using the new graph configuration.

Dataset #6

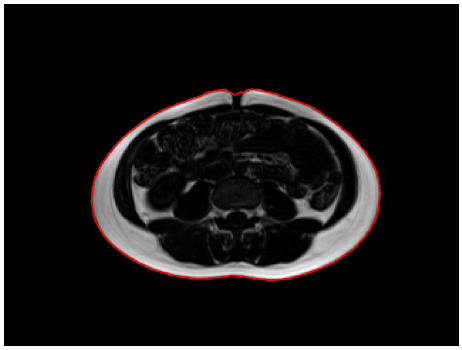


Figure 6.16: Test #2, $c\theta = 15$ - Dataset 6, Slice 17

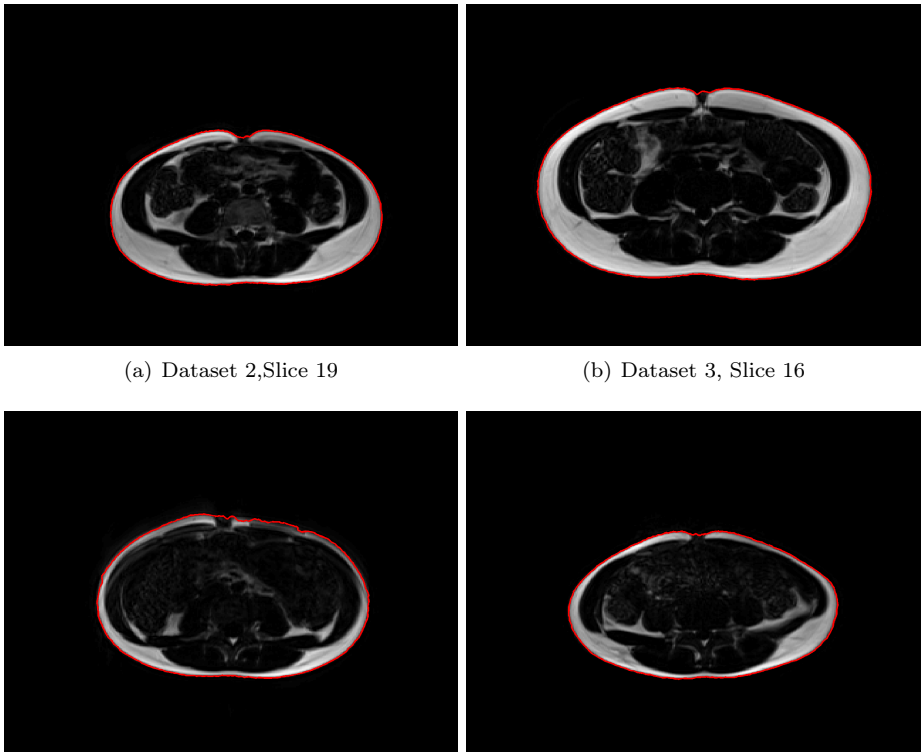
Good results are obtained for all datasets using the current graph construction and a $c\theta$ value of 15. The only visible issue appears to be a tendency of over smoothing in some of the MRI slices having a deep cut in the subcutaneous adipose tissue, for almost all of the datasets used in the test.

Table 6.3 presents the slice numbers in which the cut in SAT appears and the slice numbers in which the boundary around this region is over smoothed.

Table 6.3: Abdomen Boundary Detection - Errors for Test #2, $c\theta$=15

Dataset no.	Slice no. with cut in SAT	Slice no. with over smoothing
1	16-25	-
2	17-26	18,19
3	14-23	15,16
4	8-14	13,14,15
5	1-10	2,5,6,7
6	12-21	13-19
7	2-9	2-6
8	2-9	-
10	13-19	13,14
12	2-8	3
13	1-2	1

The images presented in Figures 6.17 and 6.18 depict the over smoothing problem described above for some of the slices enumerated in table 6.3.



(a) Dataset 2,Slice 19

(b) Dataset 3, Slice 16

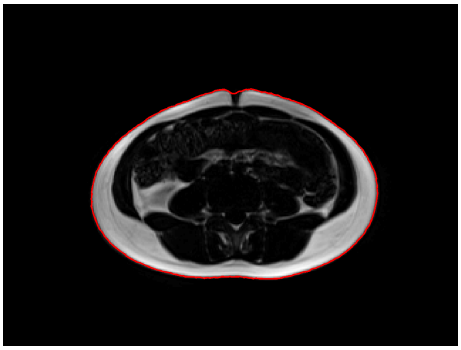
(c) Dataset 4, Slice 12

(d) Dataset 5, Slice 5

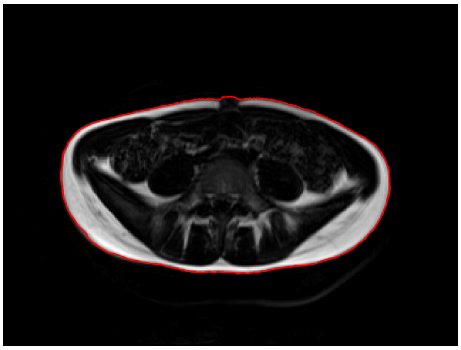
Figure 6.17: Errors for Test #2, $c\theta = 15$ - Datasets 2,3,4,5

Using the configuration from Test #2 and decreasing the smoothness constraint cost value is not a possibility in resolving the over smoothing issue, as this will cause problems as having a too small smoothness constraint cost in other regions of the MRI slices (errors as the ones depicted in Figures 6.10(b), 6.11, 6.12 and 6.13). Solving the over smoothing problems are attempted in Test #3 where inter column edges for connecting adjacent slices will be added and other later tests, in which the normalization of the gradient is done both per slice and per angle.

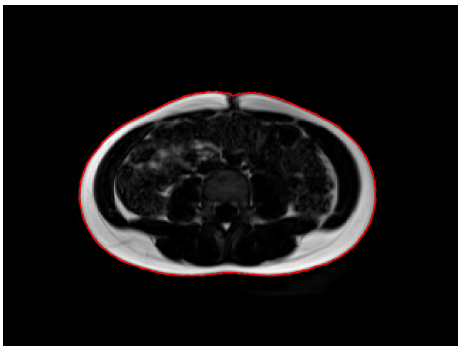
Doubling the number of edge connections in Test #2 also results in an increase in execution time of the graph construction and graph cut algorithm. The average execution times for Test #1 and Test #2 are 18.96, respectively 47.48 seconds.



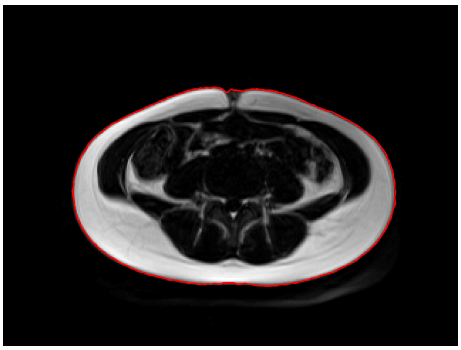
(a) Dataset 6, Slice 14



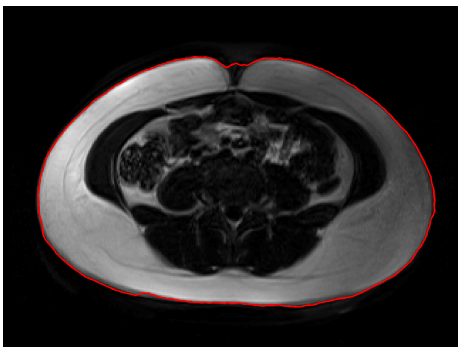
(b) Dataset 7, Slice 4



(c) Dataset 10, Slice 14



(d) Dataset 12, Slice 3



(e) Dataset 13, Slice 1

Figure 6.18: Errors for Test #2, $c\theta = 15$ - Datasets 6, 7, 10, 12, 13

3. Test #3

Th next test starts from the graph construction in Test #2, to which new edges are added between adjacent columns in the z-direction. The graph construction used in this test is depicted in Figure 6.19. For a better visualization the complete set of edges present in Test #2, is depicted only for the first group of nodes of adjacent columns in the θ direction.

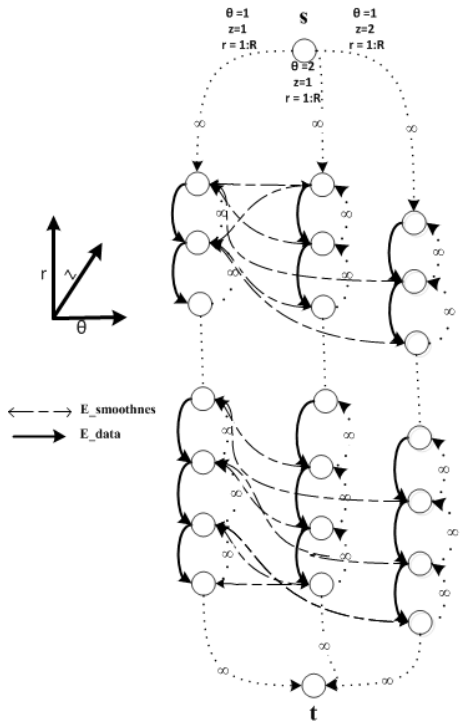


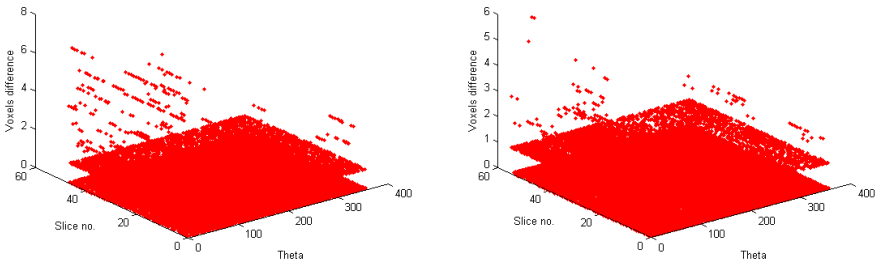
Figure 6.19: Inter column Edge Weights Test #3

From the results obtained in Test #2 it can be concluded that a value of ± 1 pixel between adjacent columns is the appropriate value for the Δ_θ parameter. To find what the value of the Δ_z parameter should be, the results for Dataset 1 and Dataset 8 from Test #2 have been analyzed.

For each of these two datasets, the difference between adjacent slices has been performed on their polar mask outputs. The polar mask is a 3D volume containing values 1 for voxels inside the abdomen and 0 elsewhere.

Figure 6.20 presents the differences between adjacent slices in the polar mask for each θ angle and for each two adjacent slices, where the differences are represented by the z-axis values.

Most of the differences in the abdomen boundary between adjacent slices are of ± 1 voxel, but there are some higher ones. This indicates that the value to be used for Δ_z could be greater than 1 and the value of the cz cost must be set to a fairly low value.



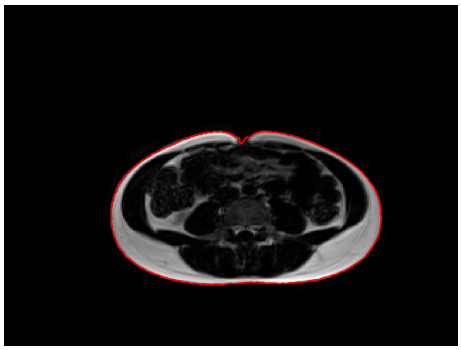
(a) Dataset 1 (b) Dataset 8

Figure 6.20: Polar mask difference in z-direction

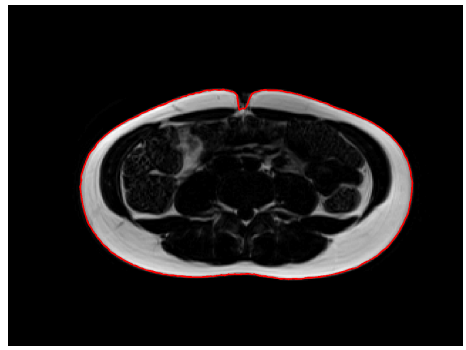
As in Test #2, when adding a new set of inter column edges to the graph construction, the cost used for the θ direction must be decreased. The values used in Test #2 where $c\theta = 10$ and $c\theta = 15$, and using both of these proved to have an over smoothing effect on the deep cut in SAT, which indicates that the cost to be used for this test should be smaller than 10.

A small test using the configuration from Test #2 was performed using values smaller than 10 for $c\theta$ in order to observe the value of $c\theta$ for which the deep cut in the SAT region is not over smoothed. The value found, $c\theta = 3$ for $\Delta_\theta = 1$ is then further used combined with the inter column edges in the z-direction. The same approach has been taken for finding good Δ_z and cz values. The appropriate value for Δ_z was found to be 2, as in all the tests where a value smaller than this has been used with a very small value for cz over smoothing occurred. Next, for finding the value of cz to be used a similar process has been performed, finding cz equal to 3.

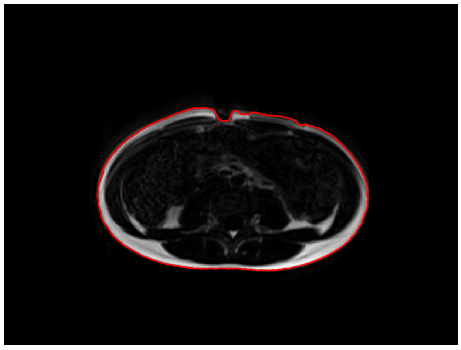
Figures below present some of the results of the Test #3, corresponding to the negative results presented for Test #2.



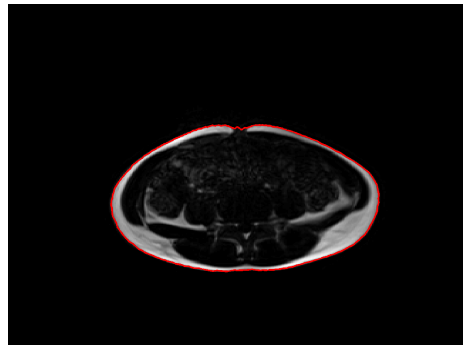
(a) Dataset 2,Slice 19



(b) Dataset 3, Slice 16

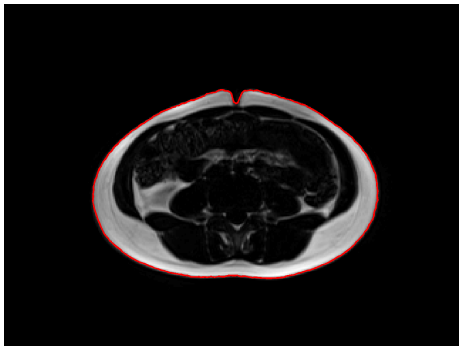


(c) Dataset 4, Slice 12

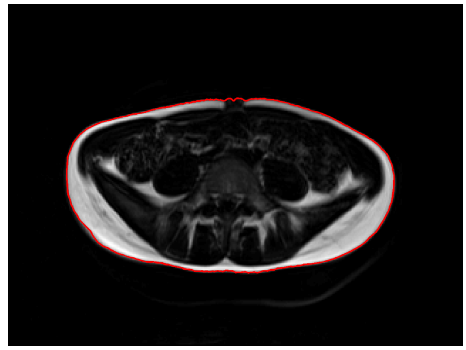


(d) Dataset 5, Slice 5

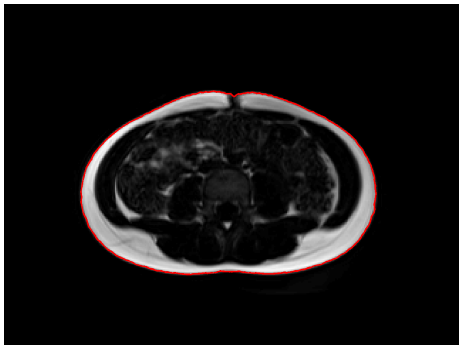
Figure 6.21: Results for Test #3 - Datasets 2, 3, 4, 5



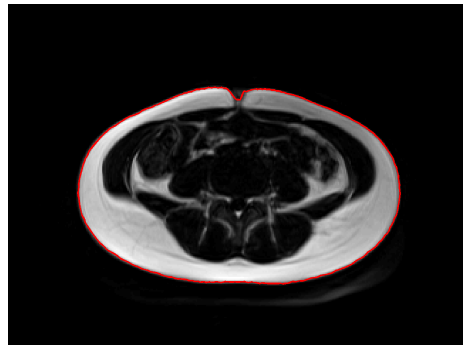
(a) Dataset 6, Slice 14



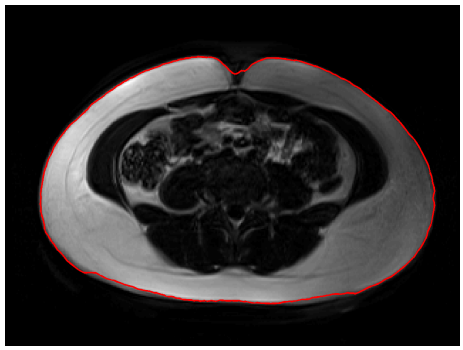
(b) Dataset 7, Slice 4



(c) Dataset 10, Slice 14



(d) Dataset 12, Slice 3



(e) Dataset 13, Slice 1

Figure 6.22: Results for Test #3 - Datasets 6, 7, 10, 12, 13

The main disadvantage of using the graph construction described in Test #3 is the increased execution time due to the added inter column edges and using a greater Δ_z value. The average execution time over all the

datasets for Test #3 is of 419.06 seconds, compared to 47.48 seconds for Test #2.

For the following two tests the graph configurations from Test #2 and Test #3 are used using the normalization of the gradient image using the maximum per slice and per θ direction for obtaining the intra column cost. For setting the inter column edge weights the same values as the ones used in Test #2 and Test #3 have been used.

4. **Test** #4

   When using the new gradient normalization method combined with the graph construction from Test #2, the results obtained are the same as the ones depicted in Figures 6.17 and 6.18 except for Dataset 2, Slice 19 which is slightly improved (Figure 6.23).

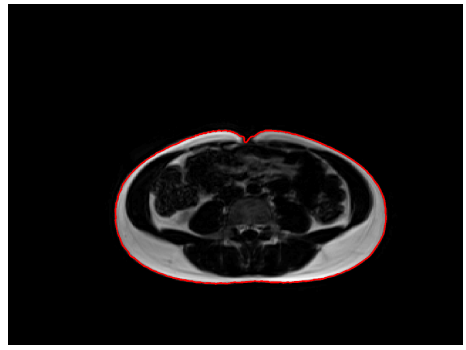


Figure 6.23: Test #4 - Dataset 2, Slice 19

   The main advantage of the settings used in Test #4 compared to Test #2 is the decrease in execution time by half, the average execution time for the tests performed using $\Delta_\theta = 1$ and $c\theta = 15$ is of 21 seconds. This improvement in execution time is directly influenced by the decrease in the intra column edge weights.

5. **Test** #5

   The results obtained using the graph construction and inter column edge weights described in Test #3, but different intra column edge weights resulted in having the exact same results as in Test #3. The improvement in execution time has been similar to the one for Test #4, the average execution time for these test being 133.62 seconds, approximately three times smaller than the average execution time registered for the Test #3 results.

The tests performed have proved the importance of several factors in obtaining a

good abdomen boundary detection. Amongst these factors the most important one is having a good cost value estimation for the intra column edge weights. The smaller the cost for the pixels belonging to the surface to be detected, the faster and easier the optimization process becomes. But as this is rarely possible in real life, the next most important factor is having an appropriate graph setting of the inter column edges and appropriate smoothness constraint cost.

As observed in Test #2, a fairly simple graph construction can give good results, but if better results are needed more constraints need to be imposed and a better choice of parameters needs to be made.

Choosing the smoothness constraint displacement (Δ) and cost is not an easy task, especially when dealing with both *'in slice'* and *'between slices'* constraints as these constraints are not independent of one each other. Setting a too high cost on the *'between slices'* constraints will force a higher smoothness of the boundary between slices, instead of the same slice. Still the easiest way in making a good parameter selection is through an iterative selection and setting of the parameters, as the one described in Test #3.

The data to be used in the next section, as an input to the SAT interior boundary detection process is the output resulted from Test #5.

6.4 SAT Interior Boundary Detection

Segmenting the subcutaneous adipose tissue (SAT) has been designed as a two step process. Firstly the abdomen boundary is detected, followed by a resampling of the dataset through which pixels outside the abdomen boundary are discarded and applying a the graph-cut algorithm a second time for finding the interior boundary of the subcutaneous fat.

The description of the SAT interior boundary detection procedure is explained from the point of view of this design. Another possible implementation of the SAT boundary detection would be through using a multiple surface segmentation. The disadvantage in performing a multiple surface segmentation is the increased memory needed, as for each of the sought surfaces a graph is constructed. Also edges connecting nodes between the two graphs need to be added for imposing locality constraints between the two surfaces.

6.4.1 Dataset resampling

As described previously, a prior step to the detection of the SAT interior boundary consists in resampling the dataset such that the voxels outside the detected abdomen boundary are excluded.

For this procedure the polar rearranged dataset and the coordinates of the abdomen boundary in polar coordinates. First the minimum coordinate value in the radial direction, denoted by r_{min}, is found from the abdomen boundary coordinates. The length of the new dataset in the radial direction is now equal to $R - r_{min}$, and the intensity values for each angle are interpolated to this length. Figure 6.24 presents a slice from the resampled dataset.

By keeping the extrapolation value for the interpolation on each angle column set to NaN, we can find the minimum coordinate value on the radial direction when the NaN values appear and discard the voxels from to the maximum length in the radial direction from the graph setup. Reducing the number of voxels in the dataset reduces the execution time of the graph-cut algorithm.

6.4.2 Intracolumn weights

The cost function for the SAT interior boundary detection can be computed either by resampling the polar coordinates dataset and recalculating the gradient

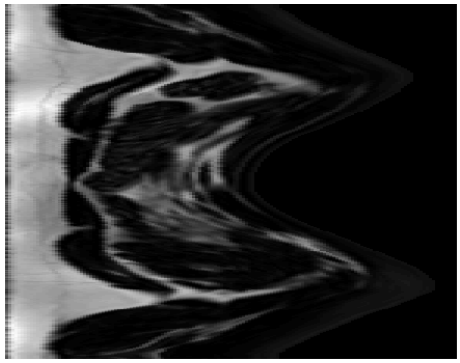


Figure 6.24: Resampling of the polar image - Dataset 13, Slice 9

of the resulted dataset or by applying the resampling directly on the gradient of the polar dataset calculated in Section 6.3. For the current implementation the resampling of the gradient information is used. Figure 6.25 describes the gradient response of the interior boundary of the SAT.

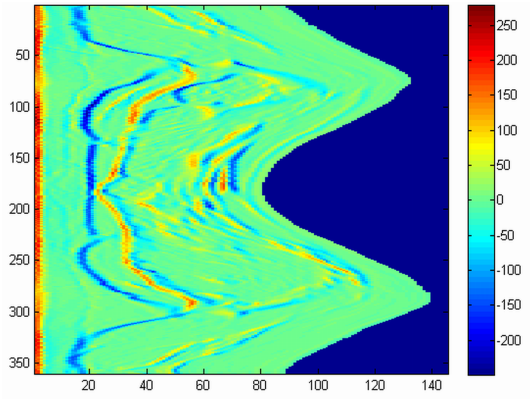


Figure 6.25: Resampling of the gradient image - Dataset 13, Slice 9

As it can be observed, the SAT interior boundary is represented by a low gradient response. A similar procedure as for obtaining the abdomen boundary cost is used - a scaling of the gradient data is performed such that the cost values are ≥ 0 and the interior boundary surface is represented by minimum cost values. The transformation used is simply adding the absolute value of the minimum value in each slice to the gradient image of that slice according to the formula presented in Equation 6.4.

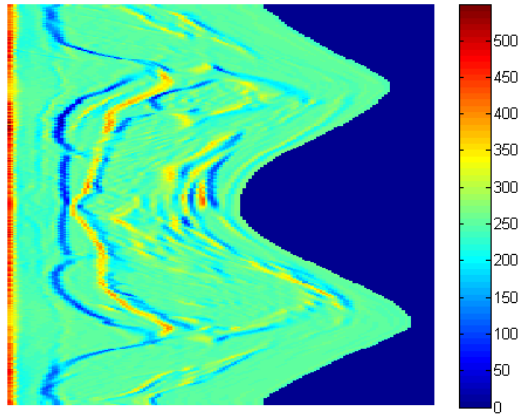
$$C(\theta, r) = |min(\frac{\partial I}{\partial \theta \partial r})| + \frac{\partial I}{\partial \theta \partial r} \tag{6.4}$$

As for the intra column edge cost used in detecting the abdomen outer boundary, an implementation using a normalization per slice and θ direction is provided also for the SAT interior boundary detection problem. Equation 6.5 presents the formula used for the second normalization method.

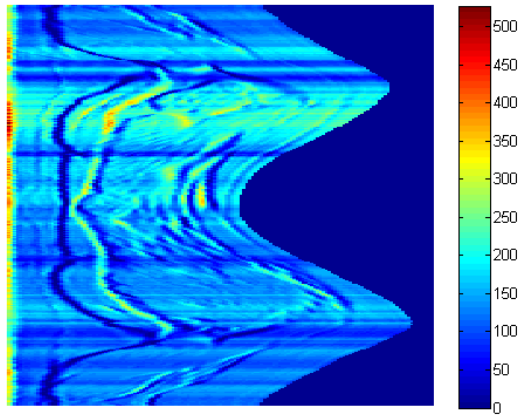
$$C(\theta, r) = |min(\frac{\partial I}{\partial \theta \partial r}(\theta))| + \frac{\partial I}{\partial \theta \partial r}(\theta) \tag{6.5}$$

Figure 6.26 depicts the cost function for the SAT interior boundary for slice 9 of dataset 13 using both normalization methods described above.

For the tests performed in Section 6.4.3, the normalization method described by Equation 6.5 will be used.



(a) Gradient normalization with absolute minimum per slice



(b) Gradient normalization with absolute minimum per slice and θ direction

Figure 6.26: SAT Interior boundary - Intra column cost for one slice: Dataset 13, Slice 9

6.4.3 Parameter tunning - SAT interior boundary detection

For estimating the smoothness constraint parameters and graph construction for finding the SAT interior boundary two tests are performed. The graph construction is chosen based on the results obtained in the parameter estimation for finding the abdomen boundary. The best performing graph constructions in Section 6.3 are the ones used in Test #2 and Test #5.

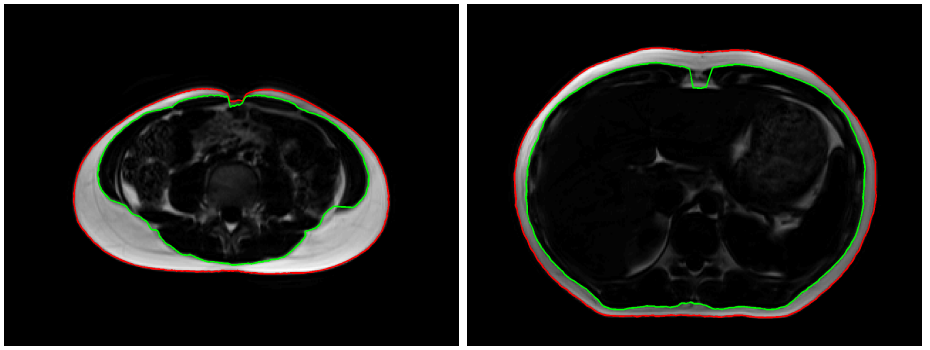
1. **Test** #1

   For the first test, the graph construction depicted in Figure 6.15 is used, setting up edges between adjacent columns in θ-direction. The Δ_θ values tested are 1 and 2 voxels displacement and values for the $c\theta$ parameter where taken from the set $\{10, 15, 25\}$. A short description of the pitfalls of each of the $c\theta$ parameter combined with a value of 1 for Δ_θ are summarized in Table 6.4.

   Table 6.4: SAT Interior Boundary - $\Delta_\theta = 1$, $c\theta = \{10, 15, 25\}$

Dataset no.	$c\theta = 10$	$c\theta = 15$	$c\theta = 25$
1	1	1	1
2	1-6,9-18	1-6,9-18	1-6,9-18
8	4-7,10	5-6	1,6
13	12,19,30,32,37-39	37	1

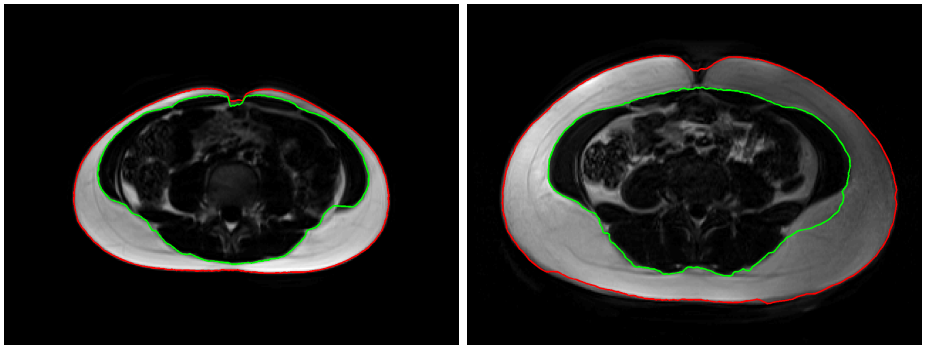
   Most of the errors registered for $c\theta = 10$ are caused by having the inter column edge cost set to a too low value, allowing the cut to be made through the edges with mimum cost on a column, which do not correspond to SAT interior boundary points. Figure 6.27 presents this type of errors for datasets 8 and 13.



(a) Dataset 8,Slice 6 (b) Dataset 13, Slice 37

Figure 6.27: Errors for $\Delta_\theta = 1$, $c\theta = 10$

The errors remaining when increasing $c\theta$ value to 15 are of the same type as the ones described for $c\theta = 10$ for datasets 8 and 13. It can be observed in the last column of Table 6.4, that the errors for datasets 8 and 13 from the previous tests have been solved, but others have appeared.



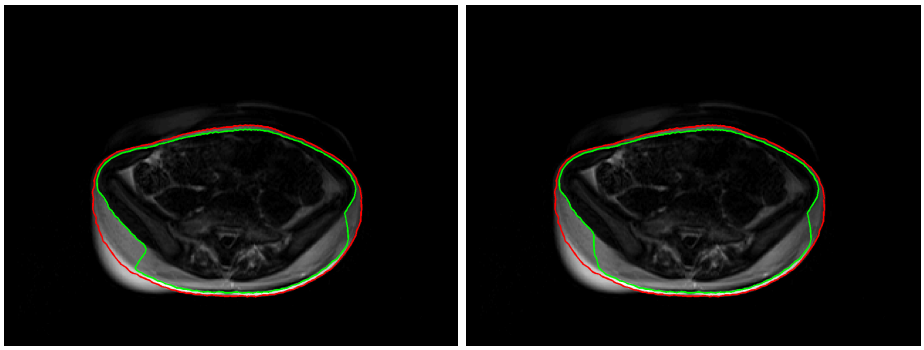
(a) Dataset 8,Slice 6 (b) Dataset 13, Slice 1

Figure 6.28: Errors for $\Delta_\theta = 1$, $c\theta = 25$

The error in slice 6 of dataset 8 has gone from being an error caused by a small cost value to a error caused by having a too high cost value (Figure 6.28(a)). That combined with the fact that the boundary in that region is not separated clearly from the VAT, the error is found indicates that a higher Δ_θ value should be used. The same applies to the error that appeared in slice 1 of dataset 8 (Figure 6.28(b)).

As with respect to the errors for datasets 1 and 2, these are caused by entirely different reasons than cost setting or graph construction.

Figure 6.29 depicts the results obtained for slice 1 of dataset 1, when using $c\theta = 10$ and $c\theta = 25$, and following this an explanation for these errors is given.



(a) $c\theta = 10$ (b) $c\theta = 25$

Figure 6.29: Errors in Dataset 1, Slice 1 for $c\theta = 10$ and $c\theta = 25$

The cause of the error in slice 1 of dataset 1 the inversion of the fat and water in the posterior abdominal region, close to the abdomen boundary. Besides this, the image is highly affected by bias field, making the cost of the voxels between the regions marked by red in Figure 6.30 and the true SAT interior boundary to have very low values.

Figure 6.30 the polar coordinates transform of the first slice of dataset 1. The regions marked by red circles are the regions causing the errors

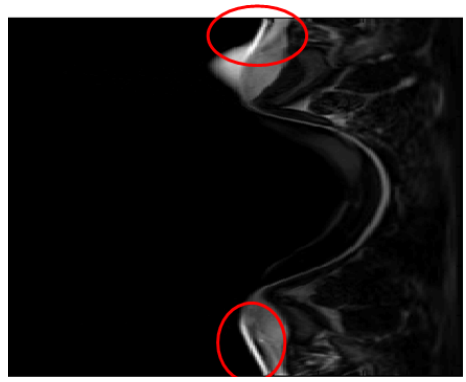
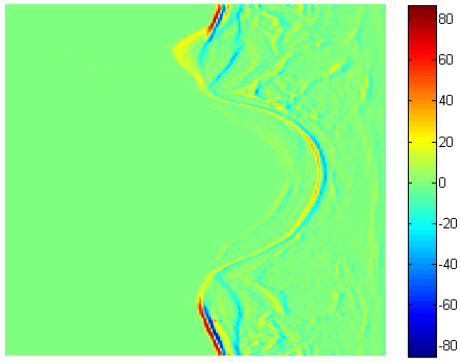


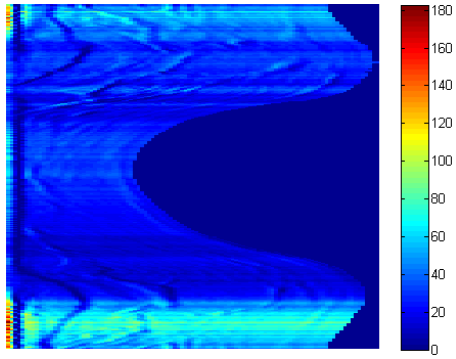
Figure 6.30: Dataset 1, Slice 1 - Polar image

in segmentation, because of the similarity between those regions and the passing from the SAT to the abdomen cavity region. Moreover the gradient value in the marked regions is much smaller than in the SAT interior boundary region, as the latter is highly affected by a bias field.

Figure 6.31 depicts the gradient value of the polar image and the cost image for the interior boundary detection. The cost image is obtained through remapping and normalization (described in Section 6.4.1 and Section 6.4.2) of the polar image.



(a) Polar image gradient



(b) Cost image

Figure 6.31: Dataset 1, Slice 1 - Gradient and Intra Column Cost

For dataset 2, the errors are caused by the cut off performed on the set of voxels described in Subsection 6.4.1, where the index of the first NaN value in all the slices and over all angles in the polar image is found. This cut off optimization works for all other datasets, but in the case of dataset 2 it cuts off parts of the SAT surface. This problem is caused by the bad alignment of the abdomen image.

Figure 6.32 depicts the first slice in dataset 2, and it can be easily observed that the center of mass of the abdomen is not the same as the physical center of the image.

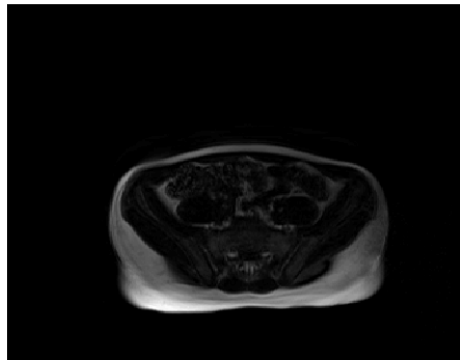
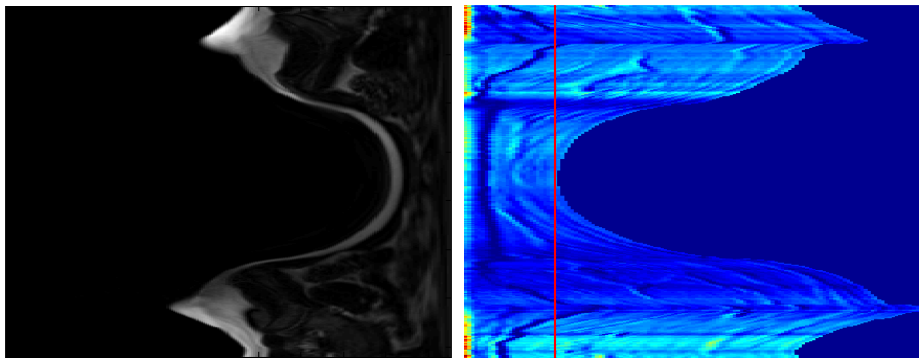


Figure 6.32: Dataset 2, Slice 1

The artifact present in Figure 6.32 combined with the fact that the center of mass of the image is not the same with the center of mass of the abdomen causes a distorted polar transform and a jump between adjacent columns in the resampled image. In Figure 6.33 the red line corresponds to the cut off on the voxels to be used in performing the graph cut.



(a) Polar image (b) Cost image

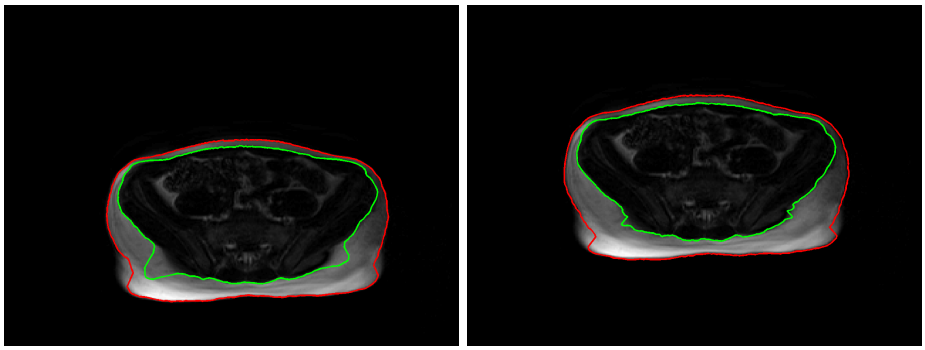
Figure 6.33: Dataset 2, Slice 1 - Polar and Intra Column Cost images

Using the abdomen boundary Cartesian mask the translation of the entire dataset has been computed:(tx,ty)=(-28,-9), being the largest translation from all datasets. Appendix A presents the translations computed for all datasets.

In Section 6.3 it has been specified that the user can select the center of the abdomen in the first slice, assuming this is consistent over all slices, and the polar coordinate transform is performed centered on the selected coordinates.

All the tests performed in the **Parameter tunning** subsection of Section 6.3 have used the original MRI scans from dataset 2 and successfully detected the abdomen boundary. An alternative approach to the manual selection of the center of mass is obtaining the rigid transform and applying this transform to the MRI scans, followed by the recalculation of the abdomen outer boundary.

Then, the new output of the abdomen boundary detection function can be used in successfully detecting the SAT interior boundary in the same manner as for all datasets. To be noted that from this point onward when referring to results obtained from dataset 2, the transformed version of the dataset is considered.



(a) Before centering (b) After centering

Figure 6.34: Dataset 2, Slice 1 - $c\theta = 15$

For the next performed test the Δ_θ value has been set to 2 and the same three $c\theta$ values from the set $\{10, 15, 25\}$ have been used.

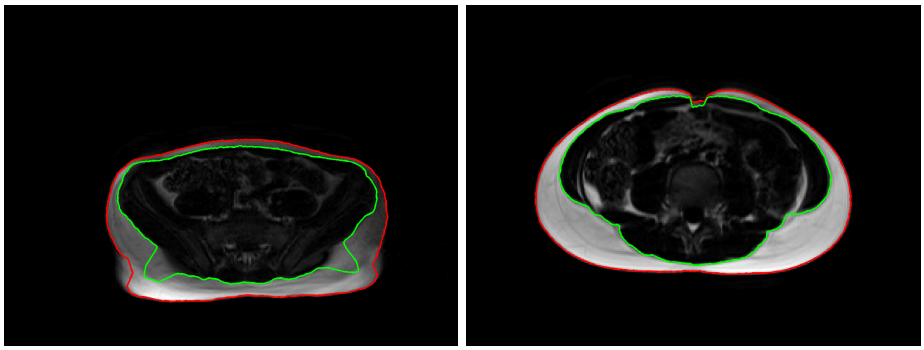
Table 6.5 presents a short description of the obtained results.

Table 6.5: SAT Interior Boundary - $\Delta_\theta = 2$, $c\theta = \{10, 15, 25\}$

Dataset no.	$c\theta = 10$	$c\theta = 15$	$c\theta = 25$
1	1	1	1
2	1	1	1
8	5-7,10	5-6	6
13	32,37-39	36	-

Fairly good results were obtained using $\Delta_\theta = 2$ and $c\theta = 25$, the only remaining errors being finding the wrong boundary due to bias field and inversion artifacts in the first slices of datasets 1 and 2, and a nonexistent boundary between SAT and VAT in slice 6 of dataset 8.

Errors in datasets 2 and 8 (Figure 6.35) can be easily corrected by changing the graph construction through addition of intercolumn edges in z-direction, constraining locality between adjacent slices.



(a) Dataset 2, Slice 1 (b) Dataset 8, Slice 6

Figure 6.35: Results for $\Delta_\theta = 2$,$c\theta = 25$

For the next performed tests the graph construction depicted in Figure 6.19, that has been used in Test #3 and Test #5 of the abdomen boundary detection, is tested for the SAT interior boundary detection. Parameter Δ_θ will be set to the optimal value found in the previous tests ($\Delta_\theta = 2$ and the value of the parameter $c\theta$ will be set to 10.

2. Test #2

In addition new inter column edges will be added constraining the SAT interior boundary between adjacent slices in the dataset. Observing the differences between adjacent slices of the boundary coordinates for one of the problematic datasets it has been observed that the most common difference between coordinates in adjacent slices is of 1 or 2 voxels, so the first test performed was set up to use $\Delta_z = 2$. For the cz values, it is known that values smaller than the $c\theta$ values need to be set, as it is considered that from the two smoothness constraints the *'in slice'* one should have a higher impact.

The following test configurations have been set up:

2.1 $\Delta_\theta = 2$, $c\theta = 10$, $\Delta_z = 2$, $cz = 5$

2.2 $\Delta_\theta = 2$, $c\theta = 10$, $\Delta_z = 2$, $cz = 10$

2.3 $\Delta_\theta = 2$, $c\theta = 15$, $\Delta_z = 2$, $cz = 5$

2.4 $\Delta_\theta = 2$, $c\theta = 15$, $\Delta_z = 2$, $cz = 10$

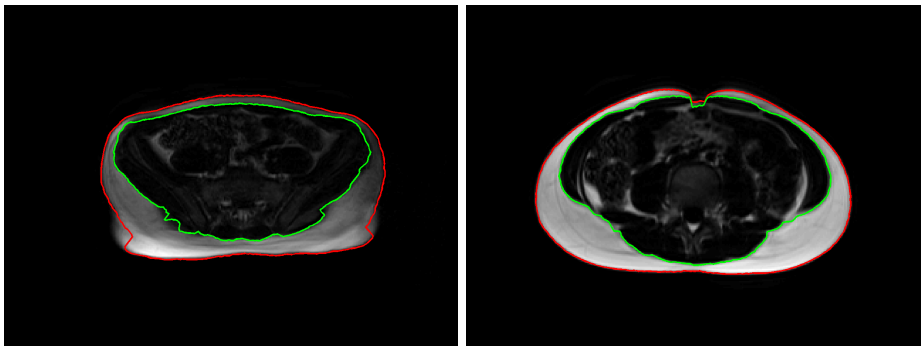
Table 6.6 presents an overview of the performance of each test set, indicating the slice number(s) for which the configuration has not performed well. The output data for all datasets from the fours tests is available on the attached CD in folder SAT interior.

Table 6.6: SAT Interior Boundary - $\Delta_\theta = 2$, $c\theta = \{10, 15\}$,$\Delta_z = 2$,$cz = \{5, 10\}$

Dataset no.	**Test** #2.1	**Test** #2.2	**Test** #2.3	**Test** #2.4
1	1	1	1	1,2
2	4	-	-	1-4
3	1	1	1	1
4	1-2	1-2	1-2	1-3
5	1-2	1	1	1-4
6	-	-	-	1
7	1-4	1-4	1-4	1-6
8	5-7	-	-	1
10	-	-	-	1
12	-	-	-	1
13	36-38	-	-	1,2

In most of the datasets better performance is obtained using the parameter setting from items 2 and 3. The errors in item 1 test are caused by having a too low cost value in both θ and z directions, while tests for item 4 are characterized by having a too high smoothness constraint value in the z-direction.

Figure 6.36 presents the results of Test 3 for datasets 2 and 8 for the images with errors in the previous test.

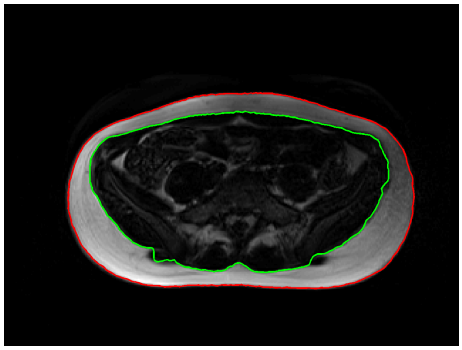


(a) Dataset 2, Slice 1 (b) Dataset 8, Slice 6

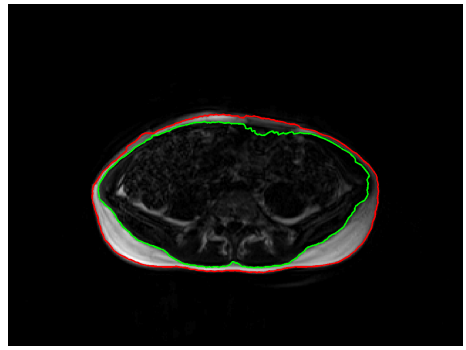
Figure 6.36: Results for $\Delta_\theta = 2$,$c\theta = 25$ for Datasets 2 and 8

The errors in datasets 4,5 and 7 are caused by having a too small smoothness constraint in the upper region of the interior SAT boundary, and can be solved by imposing a higher smoothness constraint in z-direction, as for example using $\Delta_z = 1$ and with the same cost $cx = 10$.

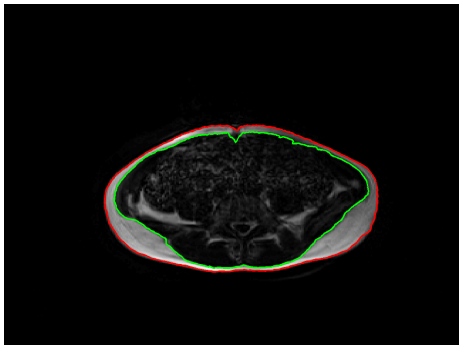
The error in slice 1 of dataset 3 is a minimal error caused by both having a high smoothness constraint cost and a not so powerful intra column cost value in the lower region of the SAT interior boundary. Figure 6.37 presents the errors found in datasets 3,4,5 and 7.



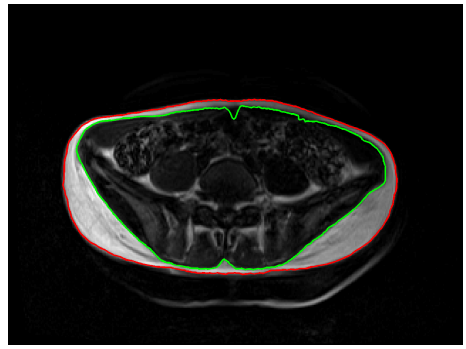
(a) Dataset 3, Slice 1



(b) Dataset 4, Slice 1



(c) Dataset 5, Slice 1



(d) Dataset 7, Slice 1

Figure 6.37: Results for $\Delta_\theta = 2$,$c\theta = 25$ for Datasets 3,4,5 and 7

6.5 Boundary detection in noisy MRI

In this section results of the graph-cut SAT boundary detection implementation on noisy MRI images will be presented.

Noise in MRI images follows a Rice (also known as Rician) distribution, as the noise in the real and imaginary parts of the k-space data are governed by a Gaussian distribution, but a non-linear operation is used to obtain the MRI image.

The Gaussian distributed noise in the real and imaginary components has zero mean and equal variance. The probability density function of the Rician distribution is presented in Equation 6.6, where ν is a noncentrality and σ is a scale parameter.

$$f(x|\nu,\sigma) = \frac{x}{\sigma^2}\exp\left(\frac{-(x^2+\nu^2)}{2\sigma^2}\right) I_0\left(\frac{x\nu}{\sigma^2}\right) \tag{6.6}$$

It is common in practice to assume that the noise in an MRI image is Gaussian noise, as the Rice distribution approaches a Gaussian distribution when the signal-to-noise ratio is high. Also, when the signal-to-noise ratio tends to zero, the Rice approaches a Rayleigh distribution.

The probability density function of a pixel x in a magnitude MRI is given by 6.7, where $A = \sqrt{A_R^2 + A_I^2}$ is the noise free signal and σ represents the equal standard deviation in A_R and A_I.

$$p_M(x|A,\sigma) = \frac{x}{\sigma^2} e^{\frac{x^2+A^2}{2\sigma^2}} I_0(\frac{Ax}{\sigma^2}) \tag{6.7}$$

As described in Equation 6.7, the noise in an MRI image depends on the data itself.

For simulating Rician noise in MRI images, a function provided by Ged Ridgway was used. This can be found on the attached CD in the Rician folder, under Code. The formula used for simulating the Rician noise is presented in Equation 6.8, where E_1 and E_2 are random numbers from $\mathcal{N}(0, \sigma^2)$ distribution, IM is the original MRI image and N_IM is the obtained noise corrupted image.

$$N\_IM = \sqrt{(IM + E_1)^2 + E_2^2} \tag{6.8}$$

Information regarding the Rice distribution and Rician noise in MRI data was obtained from [20], [9] and [18].

The graph configurations and parameter settings from Test #5 described in Subsection 6.3.2 and Test #2 (configuration 3) from Subsection 6.4.3 were used for estimating the SAT boundaries when Rician noise is present in the MR images. The results obtained using these configuration with the original input data is considered the ground-truth for this test and the estimated SAT volume estimation is compared between the two methods for all datasets for noise levels equal to $\{10, 15, 25\}$.

Table 6.7 presents the differences (as a percent of the SAT boundary in the original detection) between the boundaries detected in the noise-free MR images and boundaries detected in the noisy MR images.

Table 6.7: Errors (%) for the SAT boundary detection in noisy datasets

Dataset no.	$\sigma = 10$	$\sigma = 15$	$\sigma = 25$
1	3.0795	4.6430	7.4056
2	5.8786	5.9796	6.6644
3	1.3523	3.1210	3.1833
4	3.1534	4.4306	7.2147
5	2.3952	3.4471	5.5687
6	1.8073	2.5246	4.0673
7	2.3889	3.5395	5.4700
8	1.9225	2.8387	4.4142
10	1.8652	2.6624	4.1531
12	1.7203	2.5379	3.9409
13	0.8507	1.2289	2.0261

Most of the segmentation errors using the noise corrupted data occur in the first 5-6 slices of the datasets, as these slices are more noisy to begin with. In the rest of the dataset, even when using a high sigma value for the simulated noise, the segmentation procedure produces fairly good results. The results obtained for $\sigma = 15$ are presented in B.1

Chapter 7

Implementation of VAT Segmentation

The VAT segmentation procedure relies on the results obtained in Sections 6.3 and 6.4, mainly the abdomen mask found in the abdomen boundary detection and the SAT mask. The input data to be used for segmenting the VAT consists of the fat and water datasets in Cartesian coordinates.

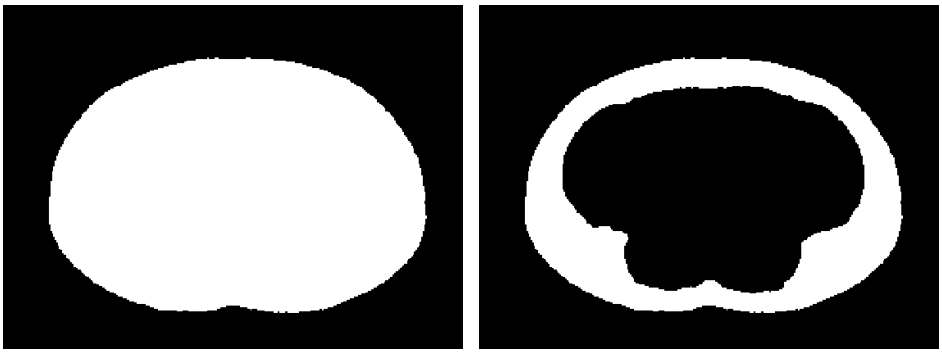
The procedure is divided into three steps:

1. Clustering the intensity values of the abdominal cavity into three classes (water, fat and bone marrow)
2. Computing the fat ratio values for the pixels belonging to the fat class
3. Thresholding the fat-ratio image calculated at step 2.

Before performing the clustering of the abdominal cavity pixel intensities, a mask is formed using the abdomen mask and the SAT mask, through the operation described in Equation 7.1, where $\bigotimes$ represents the pixel by pixel multiplication (bitwise AND operation) and $C(SAT\_mask)$ represents the binary complement of the SAT mask image.

$$mask = abdomen\_mask \bigotimes C(SAT\_mask) \tag{7.1}$$

Figure 7.1 depicts the abdomen mask, the SAT mask and the interior abdominal cavity mask.



(a) Abdomen mask

(b) SAT mask

(c) Abdominal cavity mask

Figure 7.1: Abdominal Cavity Mask - Dataset 13, Slice 9

After finding the mask for the abdominal cavity, the next step consists in clustering the pixel intensities in this region, using the mixtures of Gaussians and the EM algorithm described in Section 5.2.

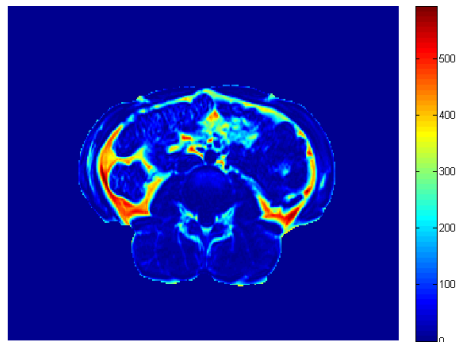


Figure 7.2: Abdominal Cavity Region Extracted from the fat image

Figure 7.2 contains the abdominal cavity region extracted from a fat-selective MRI slice belonging to dataset 13. As specified in Section 2, in a fat MR image the pixels representing the fat tissue have a high response, and the pixels representing the water have a low response. As it can be observed in Figure 7.2, there are other tissue types having responses in between the responses of the two classes. This indicates that the number of cluster k should be greater than 2.

A good way of finding the number of clusters is by plotting and evaluating the histogram of the intensity values of the region of interest. Figure depicts the histogram of intensity values for slice 9 in dataset 13 for the abdominal cavity region.

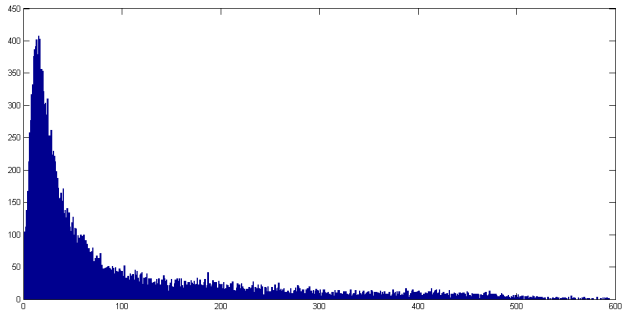


Figure 7.3: Histogram of intensity values for image in Figure 7.2

By analyzing Figure 7.2 and Figures 7.3 and 7.4 the number of clusters was selected equal to 3 as it was observed that the spinal canal and the vertebral body often appear in the fat images with medium intensity values.

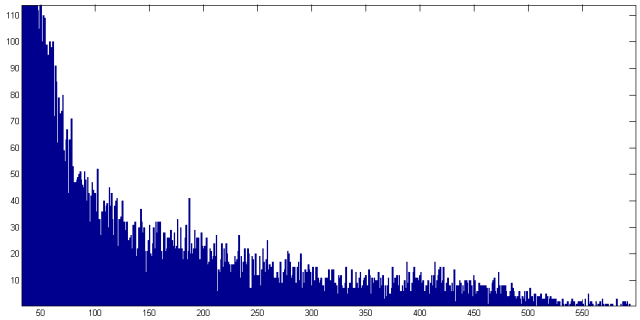


Figure 7.4: Histogram of intensity values for image in Figure 7.2 - Zoom on high intensity values

The implementation for VAT segmentation allows using the K-means algorithm for obtaining an initialization for the means, covariances and mixing coefficients for the Gaussian distributions. For performing the parameter estimation the MATLAB function gmdistribution.fit was used. The gmdistribution.fit function uses the EM algorithm to find the maximum likelihood estimates for the parameters of the Gaussian mixture.

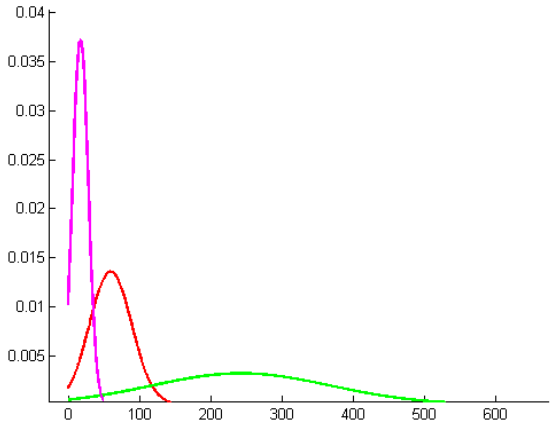
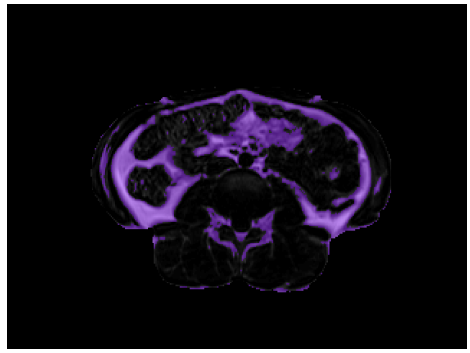


Figure 7.5: Plot of Gaussians - Dataset 13, Slice 9

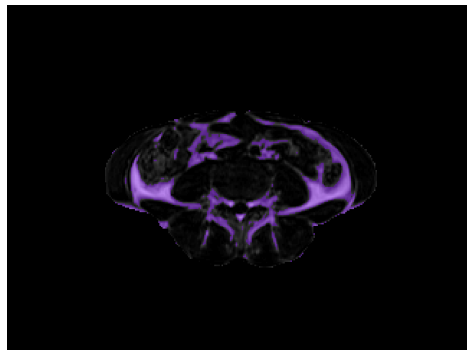
After estimating the parameters for the mixture of Gaussians, the clustering is performed using the function gmdistribution.cluster, which assigns the observed data to the component for which the observation has the highest posterior prob-

ability weighted by the mixing coefficient of the component. From the obtained clusters, the cluster representative for fat is the one having the highest intensity mean value.

Figure 7.6 presents results of the clustering for the fat signal using 3 classes for datasets 12 and 13.



(a) Dataset 13, Slice 9



(b) Dataset 12, Slice 4

Figure 7.6: Superimposed clustering result

The results when clustering into 3 different classes, classifies more than fat pixels, especially around the spinal canal region and slightly around the regions representing fat, due to the high variance of the cluster representative for VAT. A further improvement of this algorithm should take into consideration a bias-field correction of the data prior to the VAT segmentation algorithm.

The provided dataset covers the partially the lumbar region of the abdomen, the region between L1 and L4, and as no anatomical sub-region of interest has been specified, it was attempted to segment visceral fat as well as possible in all the slices of the dataset.

It has been observed that in some of the datasets the spinal canal region appears in the MRI scan with an intensity close to the one of the visceral fat and it is classified as being fat by the clustering algorithm using 3 components mixture of Gaussians.

Such examples are depicted in Figures 7.7 and 7.8.

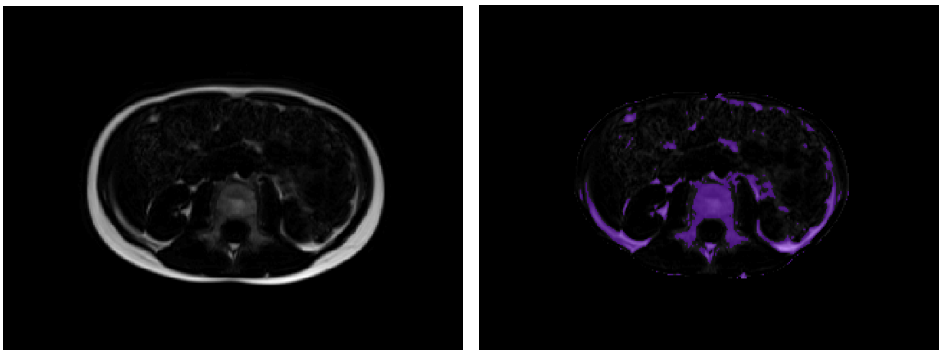


Figure 7.7: VAT volume from Clustering - Dataset 8 Slice 18

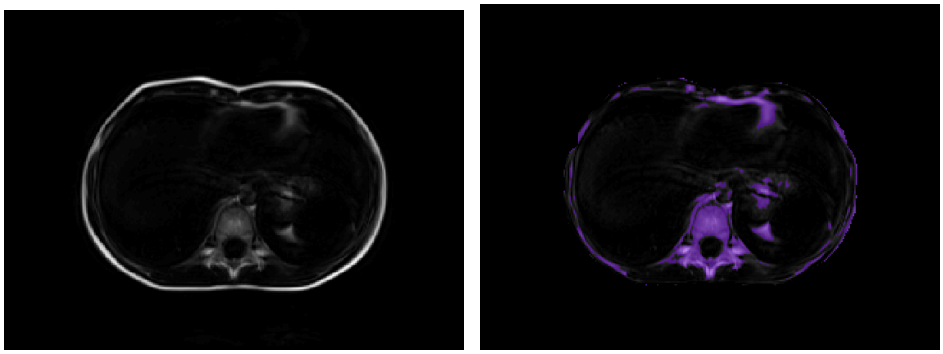


Figure 7.8: VAT volume from Clustering - Dataset 8 Slice 43

This type of problem is present in all of the datasets to some extension. For this reason, only performing mixture of Gaussians based classification of the visceral adipose tissue does not provide viable results. In order to enhance the robustness of the developed method an extra processing step has been added.

The added processing step consists in using water selective dataset and calculating the fat ratio values of the voxels previously classified as VAT. The formula through which the fat ratio image is calculated is described in Equation 7.2, where F and W represent a fat-selective and a water-selective images.

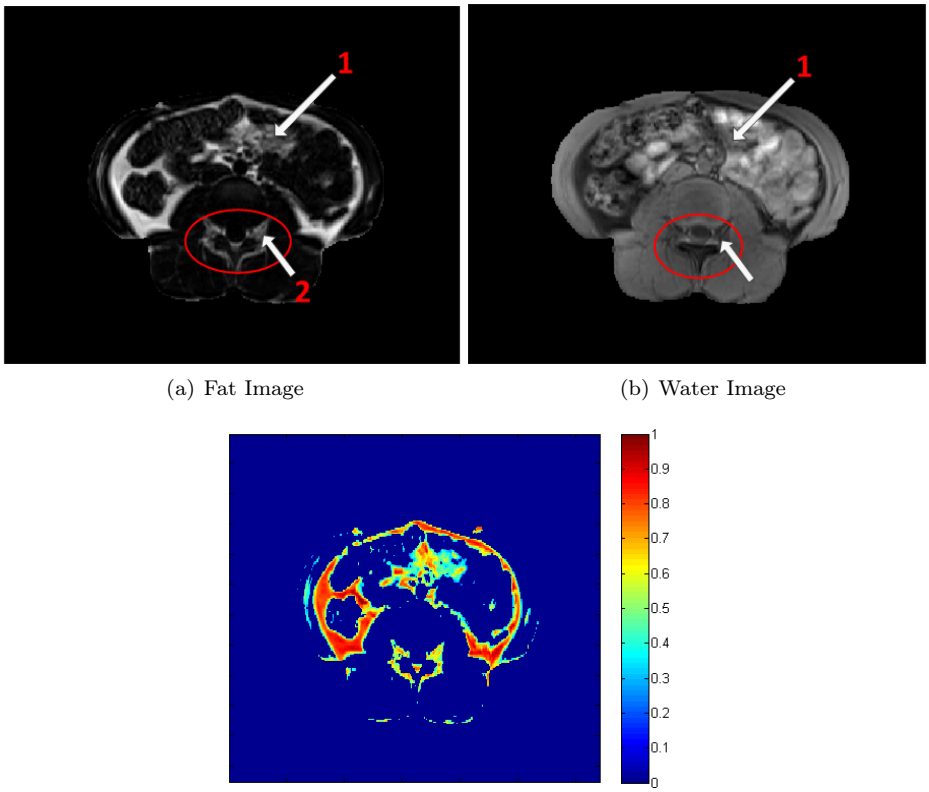
$$FR = \frac{F}{F + W} \tag{7.2}$$

Figure 7.9 presents this procedure for dataset 13, slice 9

Voxels representing the visceral adipose tissue should have values closer to 1 in the fat ratio image. As observed in Figure 7.9(c), most of the voxels representing VAT have a high probability value in the fat ratio image. Some problems appear in the region marked by 1 and 2 in Figures 7.9(a) and 7.9(b).

The region marked by 1 has a higher intensity than other areas representing VAT in Figure 7.9(b), but lower than the surrounding areas which are characterized by a higher water concentration. One of the known problems of earlier implementations of the DIXON reconstruction method is the incomplete fat suppression in the water image ([1]), which resembles the problem previously described.

The last step of the VAT segmentation procedure consists in thresholding the fat ratio images for all the slices. The value of the threshold used is the same



(a) Fat Image (b) Water Image

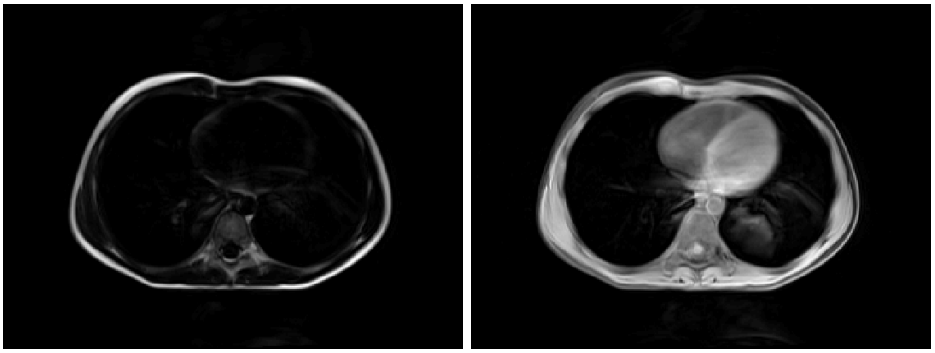
(c) Fat-Ratio image

Figure 7.9: Fat-Ratio image calculation - Dataset 13, Slice 9

for all the slices and for all the datasets. The threshold value has been selected after visual inspection of the VAT segmentation results for different threshold values. The optimal threshold value was found to be equal to 0.65.

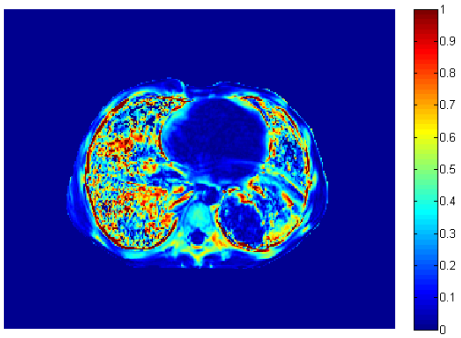
It was observed that in some of the slices, parts of the visceral adipose tissue were not segmented at all. This problem is caused by having an incomplete fat suppression in the water images, which results in a low value of the fat ratio image.

Using a segmentation based only on the thresholding of the fat ratio image would not be possible as it results in high overestimation of the VAT volume, due fat and water swapping in the higher number slices of the water datasets as depicted in Figure 7.10.



(a) Fat Image

(b) Water Image



(c) Fat-Ratio image

Figure 7.10: Fat-Ratio image - Dataset 8, Slice 53

The advantage of this method over using a simple clustering method is the ability to reduce the missclassification of regions as the vertebral canal as being visceral adipose tissue. The disadvantage is that by using also the water images, the method is sensitive to any errors in the fat or water images, as the incomplete fat suppression problem.

Quantitative results for the SAT and VAT volumes together with visualizations of some of the slices are found in Appendix B.2.

Chapter 8

Conclusions

The purpose of this thesis was to develop a method for accurate segmentation of the abdominal adipose tissue from 3D MRI obtained using the DIXON reconstruction.

For the subcutaneous adipose tissue segmentation a multi-column graph-cut based approach was used and a detailed description of the graph structure and parameter selection was provided.

The factors influencing an accurate segmentation using the implemented graph-cut method were identified as being mainly the cost function for the intra column edge weights, followed by the graph construction and selection of the inter column edge weights and the allowed variance between adjacent columns.

Good results have been obtained for all datasets in a fairly reasonable execution time. The tricky part in using a graph-cut based method as the one described for the SAT segmentation is the parameter selection. For this project it was possible to find a configuration working well for all datasets, but readjustment of the parameters might be needed for different datasets.

For the visceral adipose tissue segmentation, a clustering based approach was used. The clustering of the voxels was performed using the mixture of Gaussians model. Analyzing the datasets it was observed that in some of the slices

the vertebral body and the spinal cord have high intensities in the fat images, causing misclassification of these regions when only using the clustering method.

For solving this problem additional information from the water images was used, through the calculation of the fat ratio image for the voxels classified as fat by the clustering. The final VAT segmentation was performed through a thresholding of the fat ratio image. Ideally the threshold value should be ≥ 0.8 (for pure adipose tissue voxels).

As no ground truth segmentation was available, multiple threshold values were tested, and through visual validation the best threshold for these datasets was used.

The disadvantage of this methodology is that, having set a too high threshold value caused under estimation of the VAT volume, and as the threshold was decreased, areas in the vertebral body were misclassified as fat. The low fat-ratio values in regions covered by VAT are thought to be caused by an incomplete fat suppression in the water image. The implementation of the VAT segmentation algorithm is done per slice.

No bias field correction was performed on the data, and good results were obtained for the SAT segmentation. The VAT segmentation, on the other hand seems to be affected by the lack of bias correction. Having a bias correction prior to these procedures would increase the differences between the classes and better results might be obtained using a simple clustering method.

The MRI scanner could affect the quality of the input data, as different vendors implement different methods for the DIXON reconstruction. But as the Siemens MRI scanner used for acquiring the data implements a 2-point DIXON method, comparable results are expected on evaluation of data from other scanners implementing a 2-point DIXON method and better results when a later version of the DIXON reconstruction method is implemented.

8.1 Future Work

An improvement could be brought to the SAT segmentation algorithm in what it concerns the execution time. The highest execution time occurs when detecting the abdomen boundary segmentation. In the current implementation, all the voxels from the polar transformed dataset are used as nodes in the graph.

Excluding voxels outside the abdomen boundary can be easily done by user input

of the maximum radius length to be used in the polar coordinates transform, or finding this radius automatically by a rough thresholding and finding the largest connected component. Otherwise, growing an ellipse in the Cartesian coordinates transform could be used as described in [11] or finding a rectangular bounding box in the Cartesian coordinates system as described in [14].

The main problem in the VAT segmentation is the misclassification of the vertebral body as fat and slightly bright shadows in the higher number slices.

In making the VAT segmentation method more robust a more clear separation of the classes is needed, in this case a clustering based approach would give good results.

A graph-cut based approach using the mean and variance estimated automatically for each slice through the mixture of Gaussians clustering could be another option for improving the VAT segmentation. This approach allows the addition of edges between voxels belonging to adjacent slices and imposing of constraints between these voxels.

Appendix A

Appendix

List of identification numbers associated to each dataset, used for an easier reference of the MRI figures.

Dataset name	Dataset no.
10160000	1
10410000	2
17230000	3
57040000	4
36240000	5
59450000	6
13060000	7
14500000	8
14410000	9
14290000	10
14040000	11
13470000	12
13340000	13

Table A.1: List of dataset identification numbers

Dataset no.	Size (Voxels)	No. slices	Slice thickness (mm)
1	250 x 320	48	3
2	270 x 320	48	3
3	250 x 320	48	3
4	270 x 320	48	3.5
5	290 x 320	48	3
6	250 x 320	48	3
7	230 x 320	56	4
8	230 x 320	56	4
9	230 x 320	56	4
10	230 x 320	56	4
11	230 x 320	56	4
12	230 x 320	56	4
13	230 x 320	56	4

Table A.2: List of size in voxels and slice thickness for all datasets

Dataset no.	Tx	Ty
1	-18	1
2	-28	-9
3	-7	-8
4	-15	-2
5	-23	-2
6	-11	-5
7	-9	1
8	-5	-1
10	-12	-2
12	-13	0
13	-8	-2

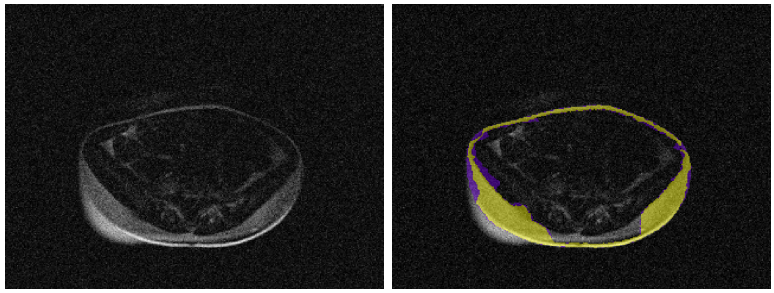
Table A.3: List of translations of the abdomen boundary with respect to the image center

Appendix B

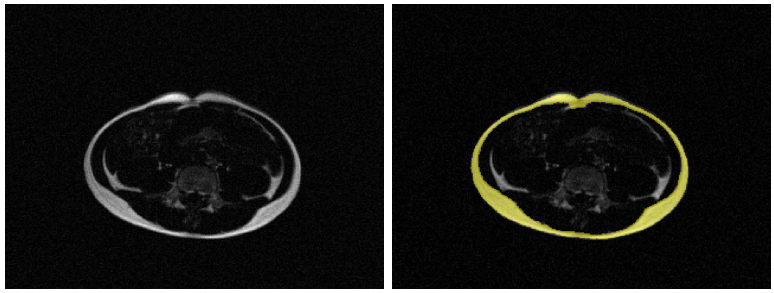
Appendix

B.1 SAT detection in noisy MRI

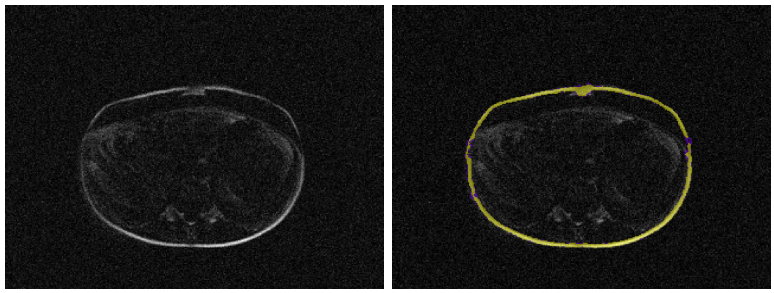
The following pages present results for all the datasets using what are considered to be the optimal parameter setting through visual inspection of the results for the subcutaneous adipose tissue segmentation on the original dataset. The figures described the SAT segmentation using the noise corrupted datasets for $\sigma = 15$ in yellow and the differences between the noise-free and noisy segmentations is colored in violet.



Slice 1

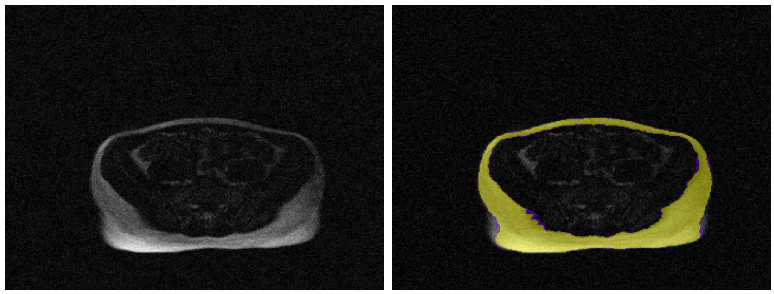


Slice 16

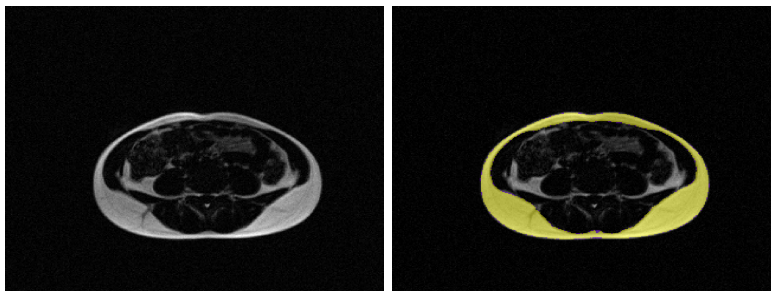


Slice 48

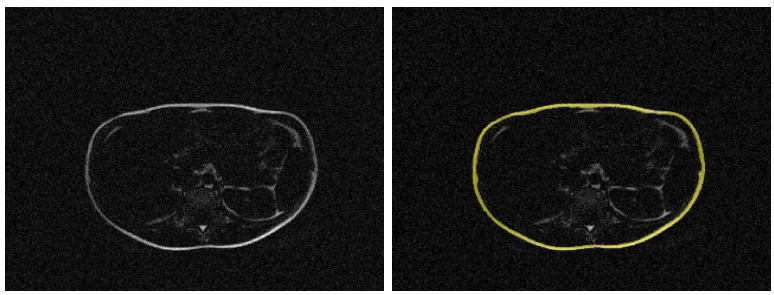
Figure B.1: SAT Segmentation for Dataset #1 in the presence of noise



Slice 1

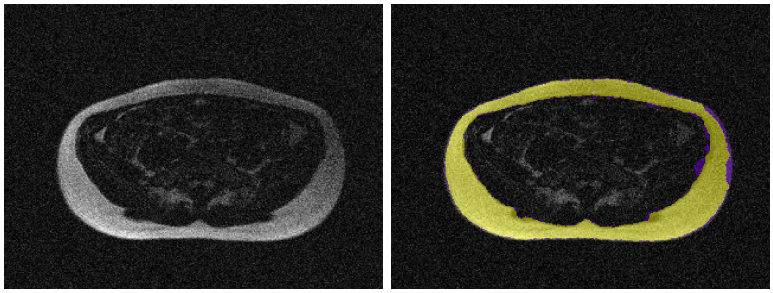


Slice 16

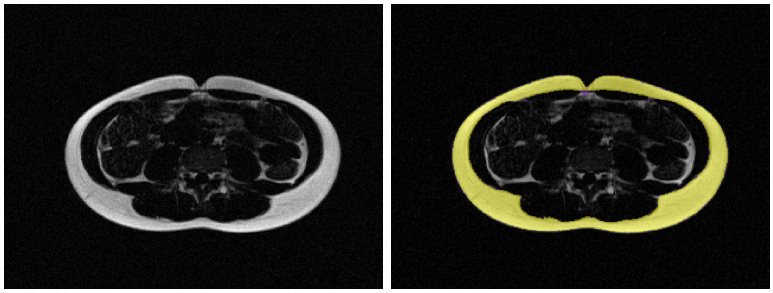


Slice 48

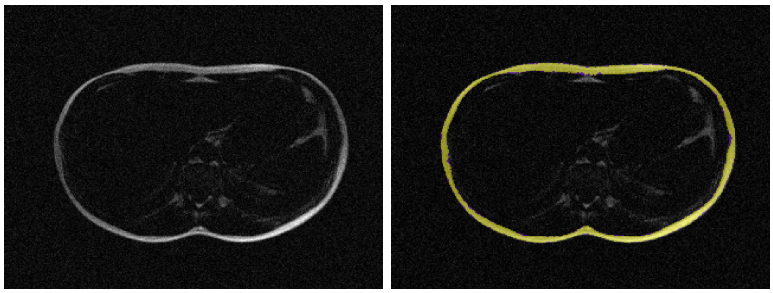
Figure B.2: SAT Segmentation for Dataset #2 in the presence of noise



Slice 1



Slice 19



Slice 56

Figure B.3: SAT Segmentation for Dataset #3 in the presence of noise

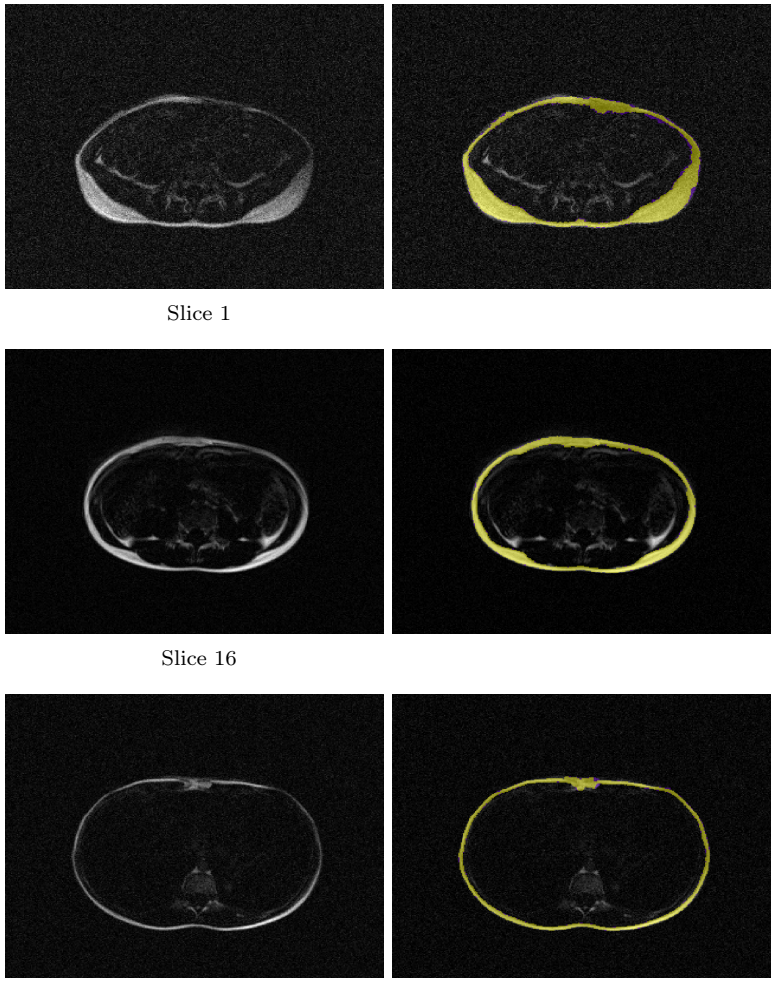
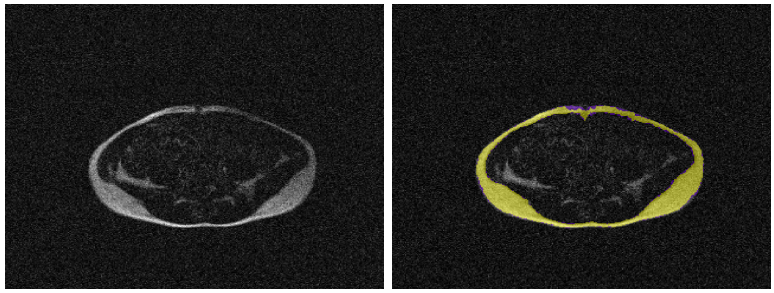
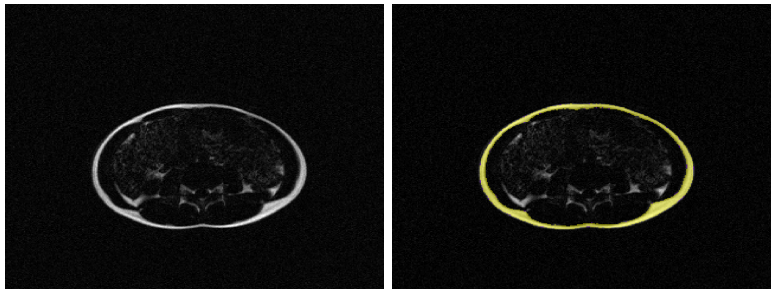


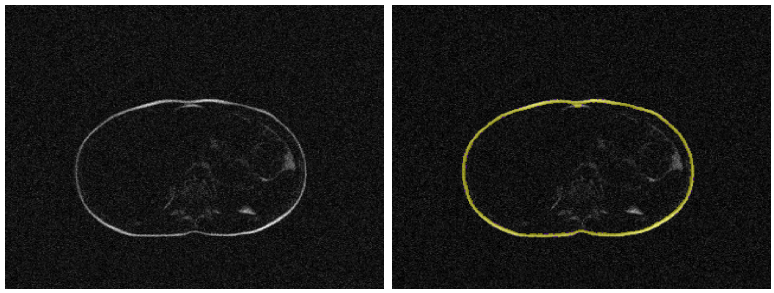
Figure B.4: SAT Segmentation for Dataset #4 in the presence of noise



Slice 1

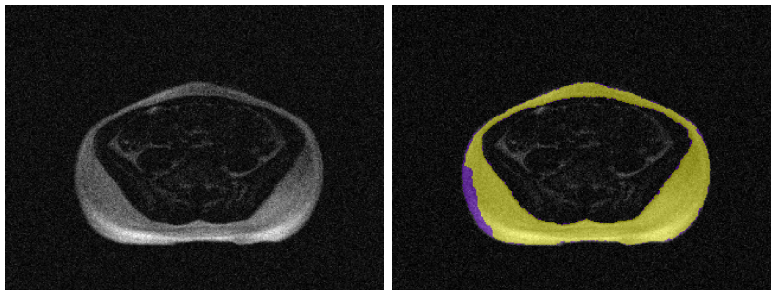


Slice 16

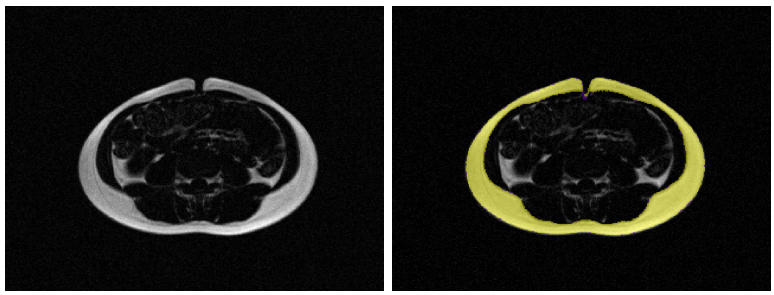


Slice 48

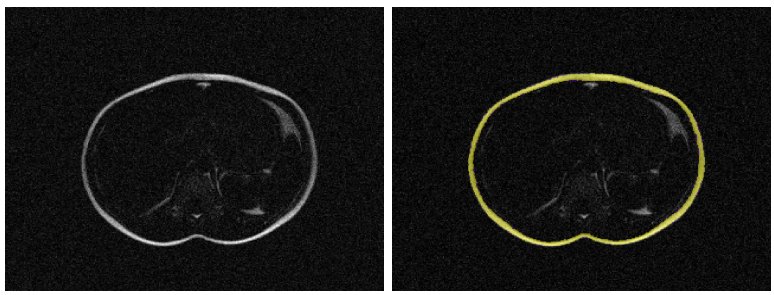
Figure B.5: SAT Segmentation for Dataset #5 in the presence of noise



Slice 1

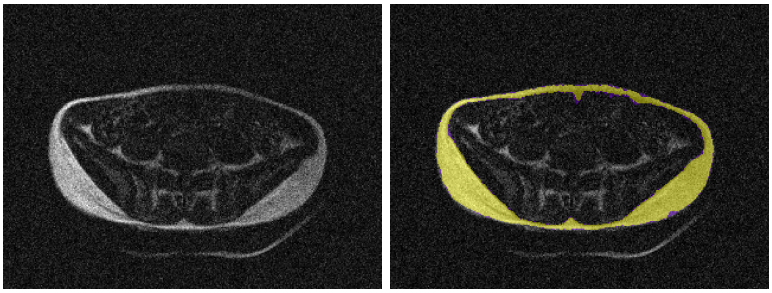


Slice 16

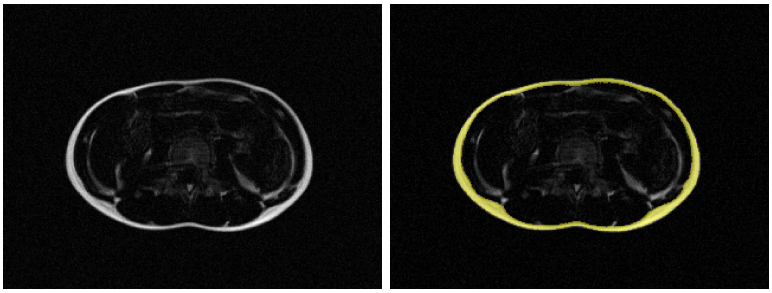


Slice 48

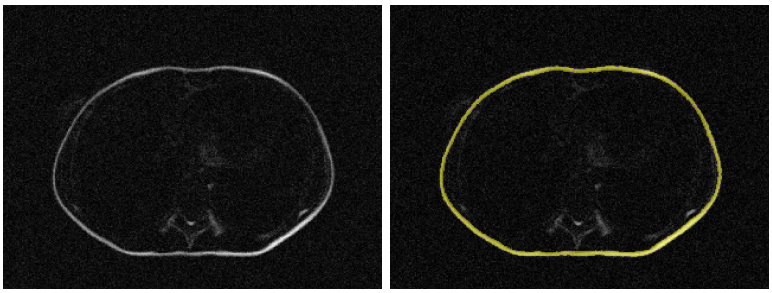
Figure B.6: SAT Segmentation for Dataset #6 in the presence of noise



Slice 1

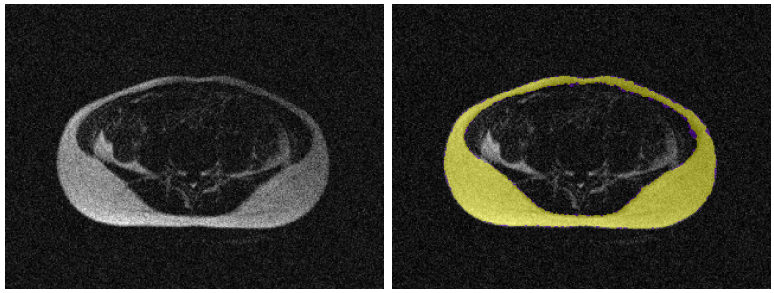


Slice 19

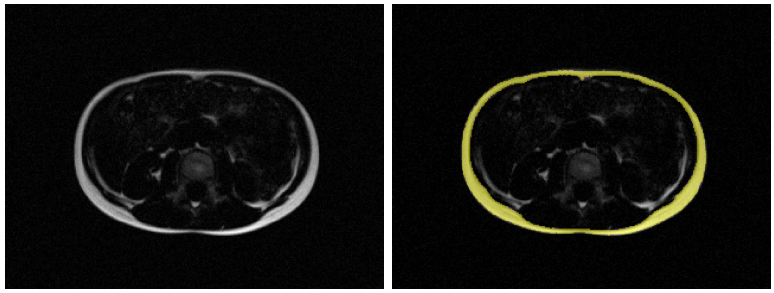


Slice 56

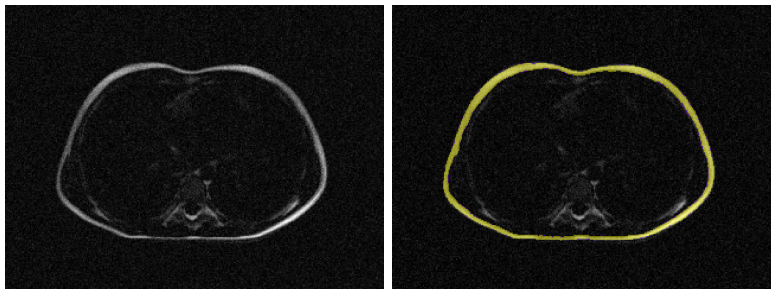
Figure B.7: SAT Segmentation for Dataset #7 in the presence of noise



Slice 1

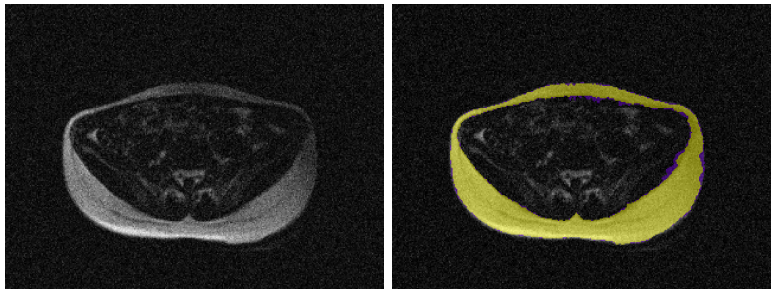


Slice 19

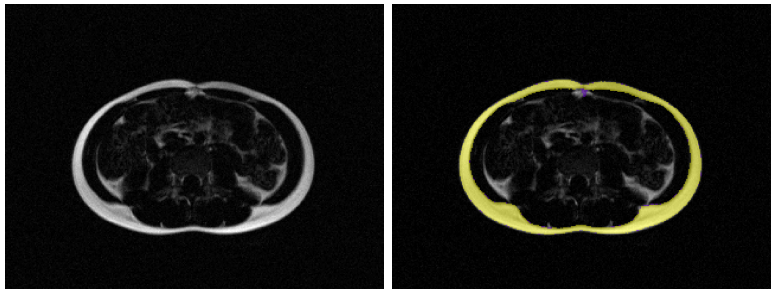


Slice 56

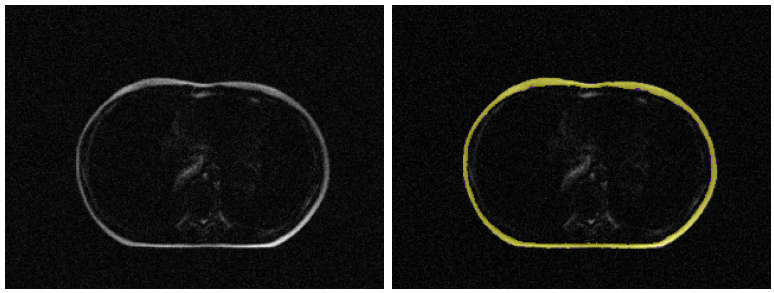
Figure B.8: SAT Segmentation for Dataset #8 in the presence of noise



Slice 1

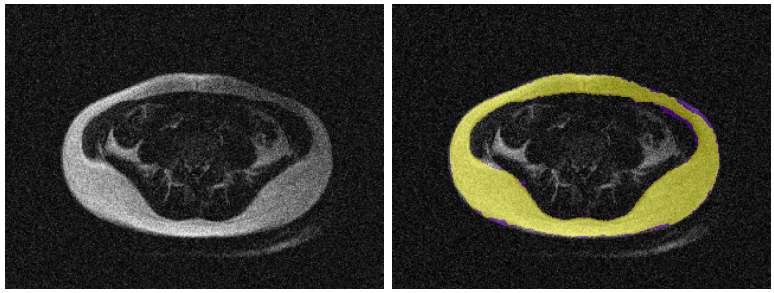


Slice 19

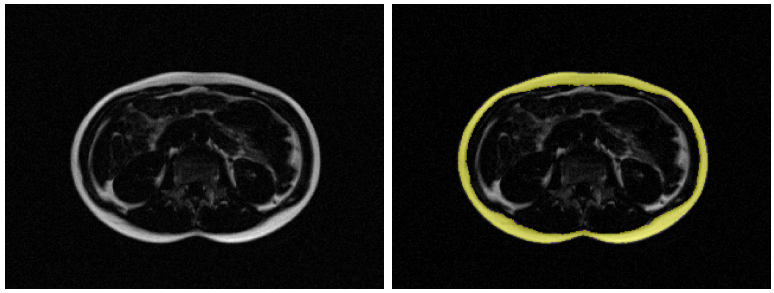


Slice 56

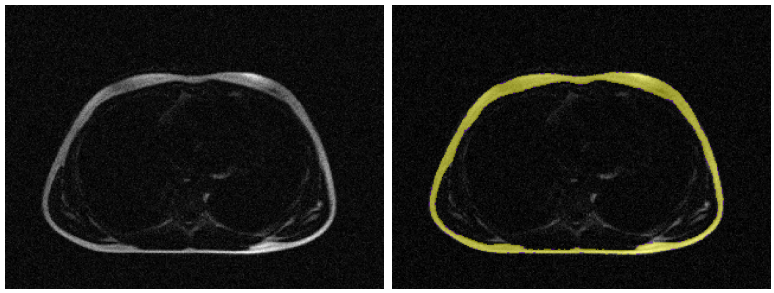
Figure B.9: SAT Segmentation for Dataset #10 in the presence of noise



Slice 1

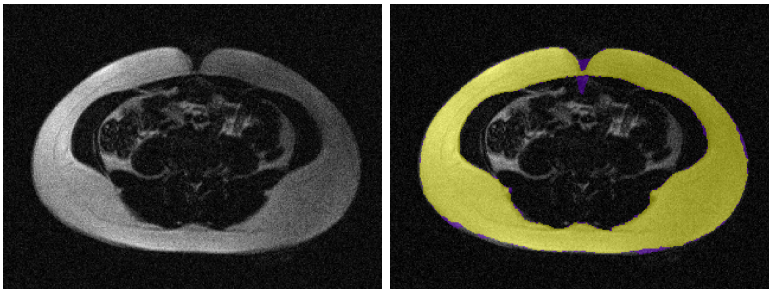


Slice 19

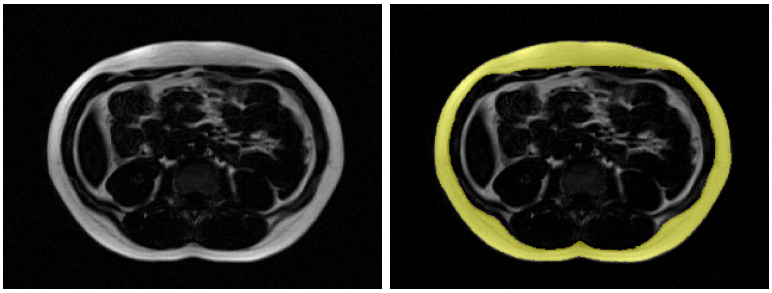


Slice 56

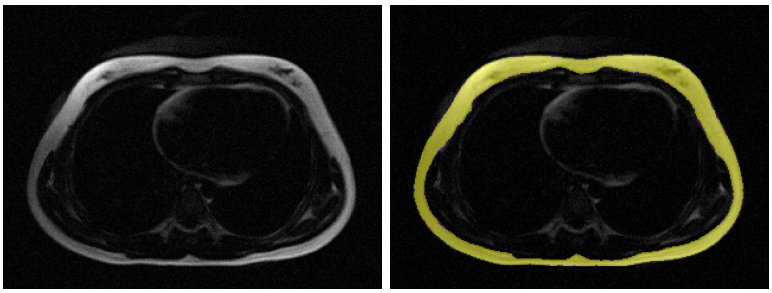
Figure B.10: SAT Segmentation for Dataset #12 in the presence of noise



Slice 1



Slice 19



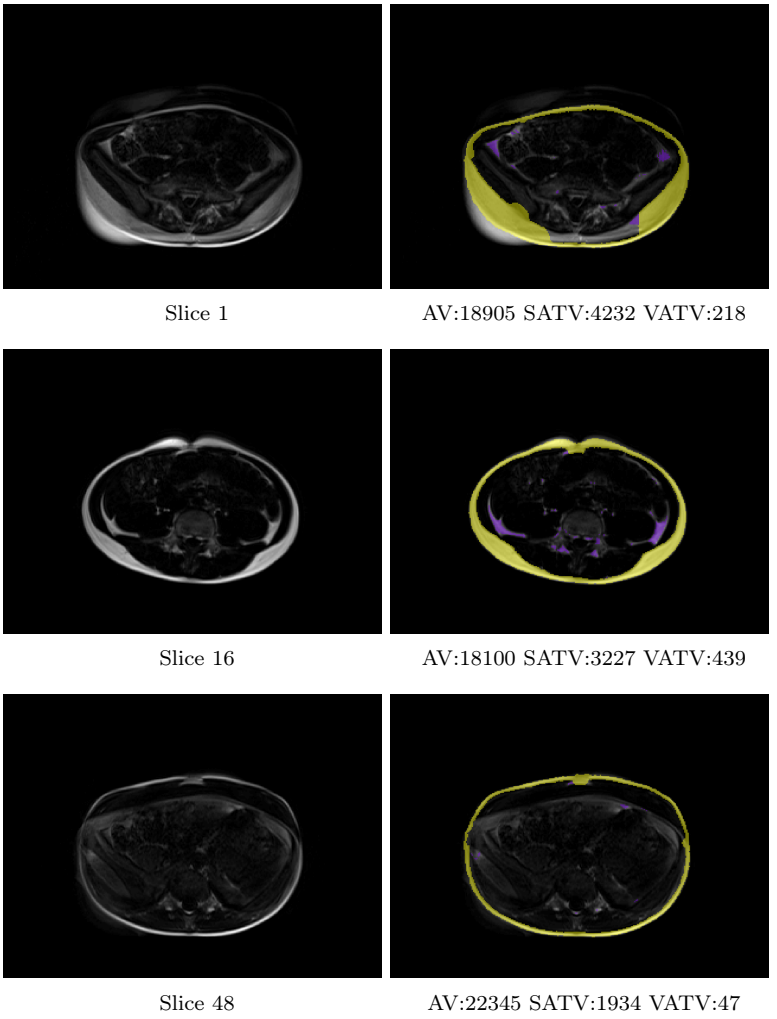
Slice 56

Figure B.11: SAT Segmentation for Dataset #13 in the presence of noise

B.2 SAT and VAT estimated volumes

The following pages present results for all the datasets using what are considered to be the optimal parameter setting by visual analysis of the results both for the subcutaneous adipose tissue segmentation and the visceral adipose tissue segmentation. Also the values for the total estimated volume of subcutaneous and visceral adipose tissue are presented for this parameter setting. For each of the datasets small videos containing all the slices in the dataset and the superimposed segmented adipose tissue are available on the attached CD. The SAT and VAT percentage is calculated by dividing the SAT (VAT) volume to the abdomen volume.

Dataset #1			
Abdomen volume:	926607		
SAT volume:	126837	**SAT %:**	13.7
VAT volume:	22484	**VAT %:**	2.4



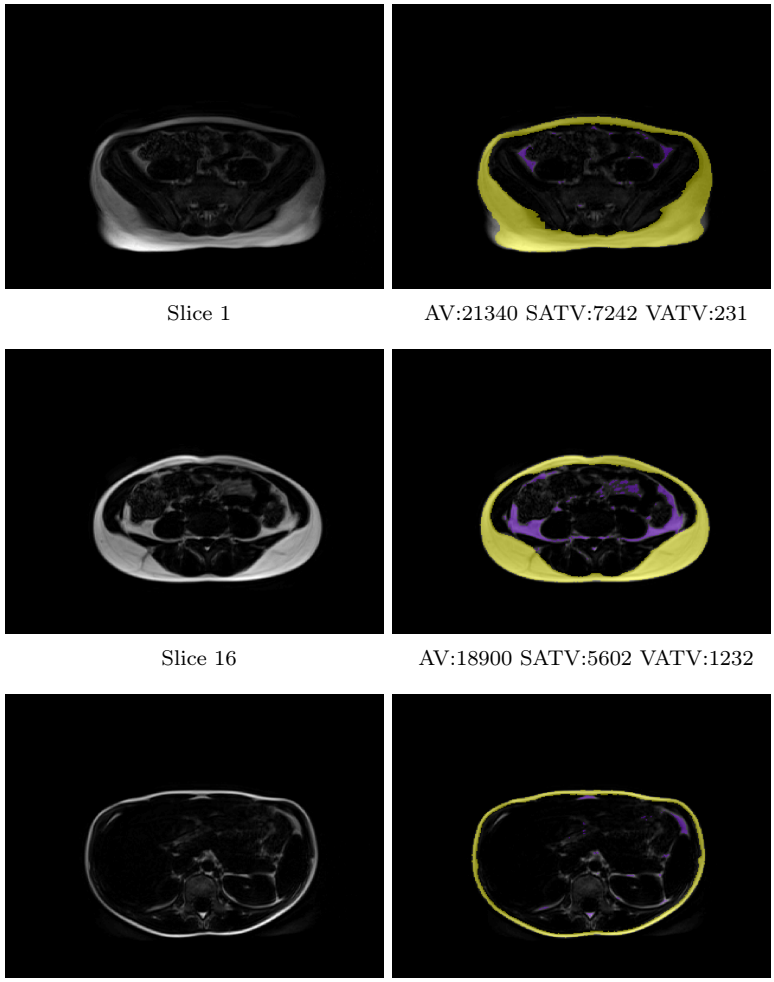
Slice 1 AV:18905 SATV:4232 VATV:218

Slice 16 AV:18100 SATV:3227 VATV:439

Slice 48 AV:22345 SATV:1934 VATV:47

Figure B.12: Final segmentation for Dataset #1

Dataset #2			
Abdomen volume:	951862		
SAT volume:	198108	**SAT %:**	21
VAT volume:	33385	**VAT %:**	4



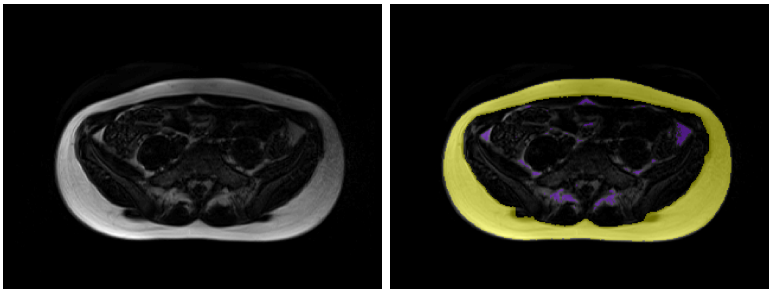
Slice 1 AV:21340 SATV:7242 VATV:231

Slice 16 AV:18900 SATV:5602 VATV:1232

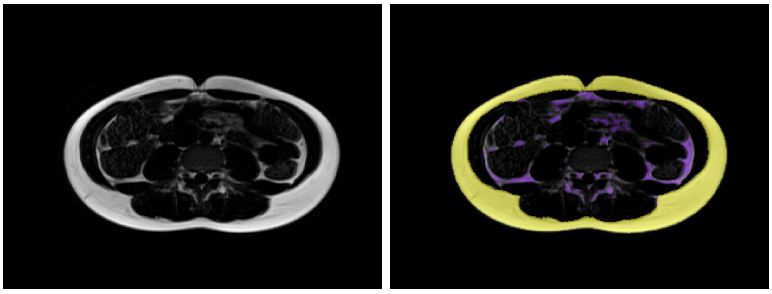
Slice 48 AV:22728 SATV:1941 VATV:191

Figure B.13: Final segmentation for Dataset #2

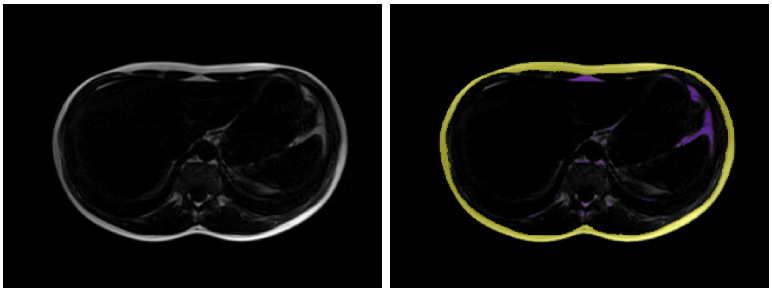
Dataset #3			
Abdomen volume:	1567413		
SAT volume:	354501	**SAT %:**	22.6
VAT volume:	54844	**VAT %:**	3.5



Slice 1 AV:28991 SATV:10323 VATV:478



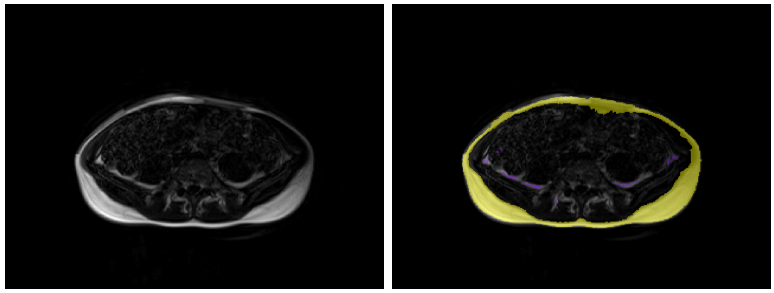
Slice 19 AV:25913 SATV:7620 VATV:1144



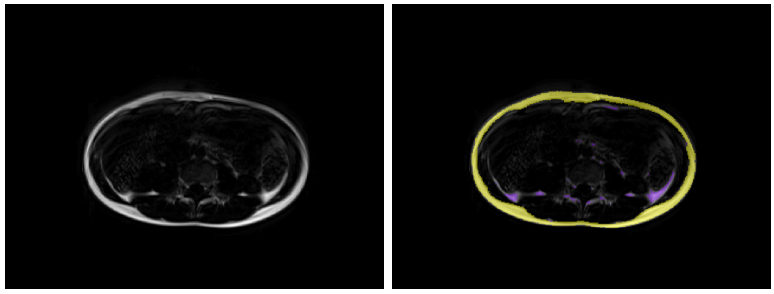
Slice 56 AV:33266 SATV:3642 VATV:481

Figure B.14: Final segmentation for Dataset #3

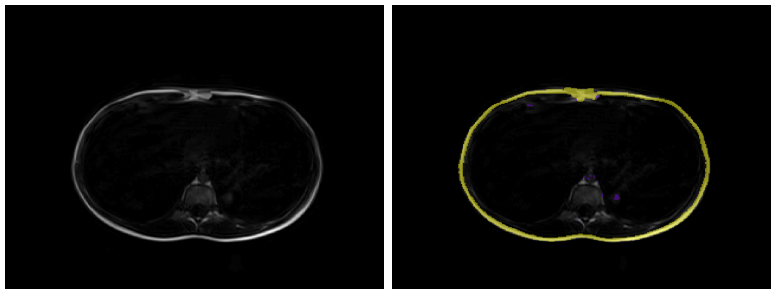
Dataset #4Abdomen volume:	1042024		
SAT volume:	130262	**SAT %:**	12.5
VAT volume:	11601	**VAT %:**	1.1



Slice 1 AV:20098 SATV:4328 VATV:140



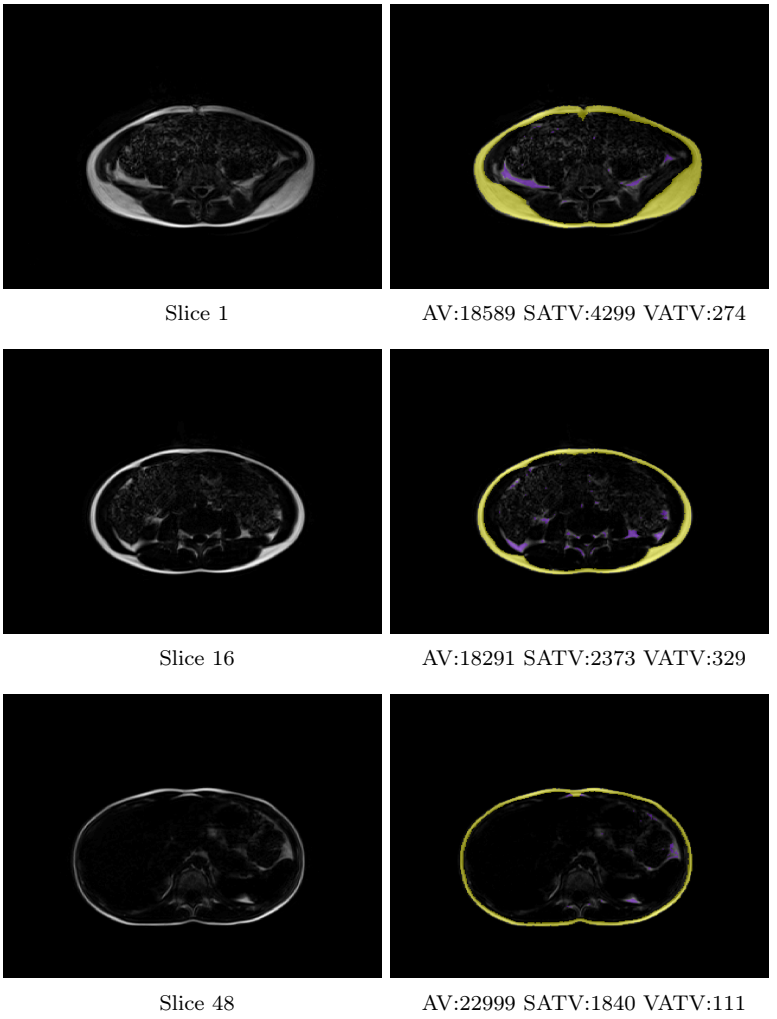
Slice 16 AV:19710 SATV:2894 VATV:312



Slice 48 AV:25543 SATV:2152 VATV:59

Figure B.15: Final segmentation for Dataset #4

Dataset #5			
Abdomen volume:	952560		
SAT volume:	112934	**SAT %:**	11.9
VAT volume:	12345	**VAT %:**	1.3



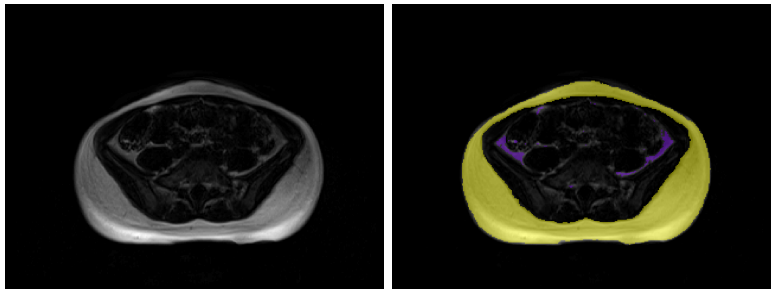
Slice 1 AV:18589 SATV:4299 VATV:274

Slice 16 AV:18291 SATV:2373 VATV:329

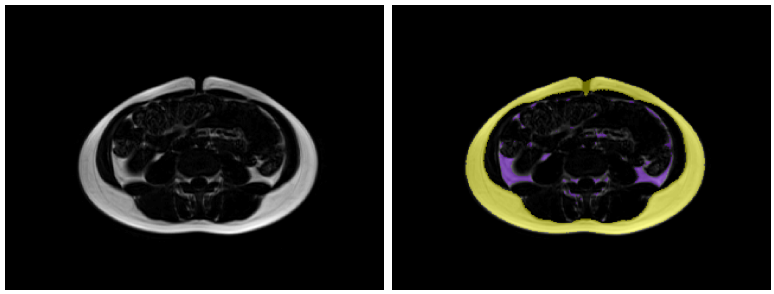
Slice 48 AV:22999 SATV:1840 VATV:111

Figure B.16: Final segmentation for Dataset #5

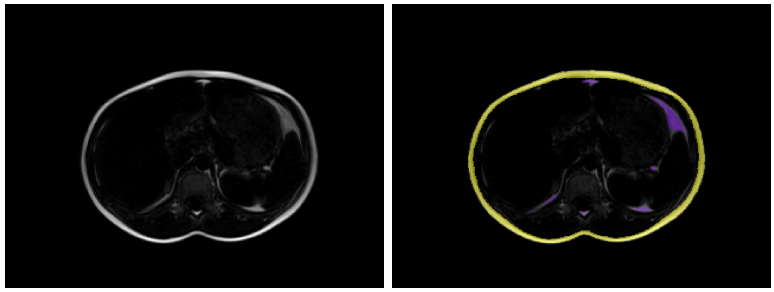
Dataset #6			
Abdomen volume:	1067144		
SAT volume:	235271	**SAT %:**	22
VAT volume:	29571	**VAT %:**	2.8



Slice 1 AV:24065 SATV:9559 VATV:381



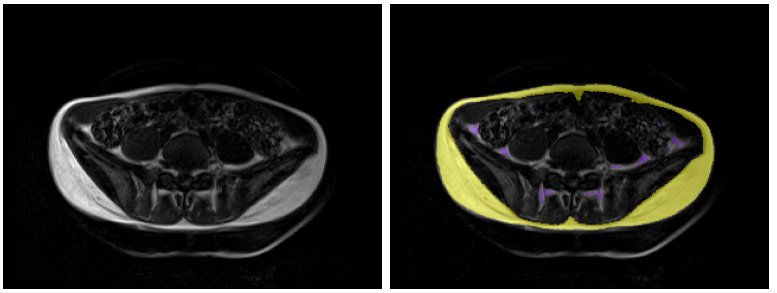
Slice 16 AV:21712 SATV:6427 VATV:768



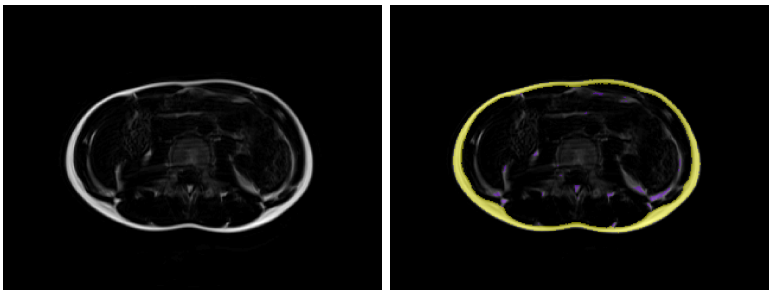
Slice 48 AV:24427 SATV:2554 VATV:345

Figure B.17: Final segmentation for Dataset #6

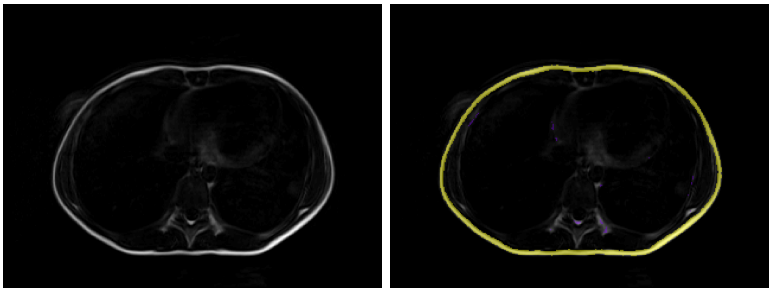
Dataset #7			
Abdomen volume:	1367895		
SAT volume:	159488	**SAT %:**	11.7
VAT volume:	12801	**VAT %:**	0.9



Slice 1 AV:22579 SATV:5637 VATV:222



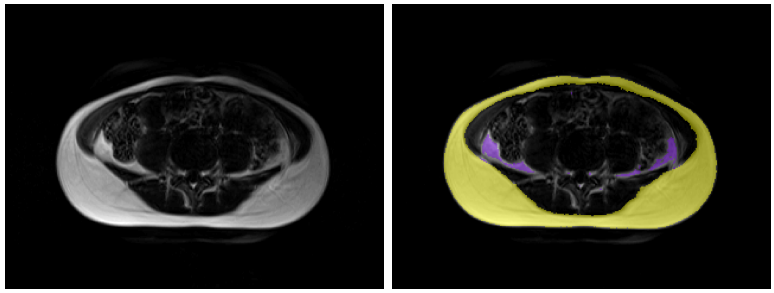
Slice 19 AV:20851 SATV:2668 VATV:186



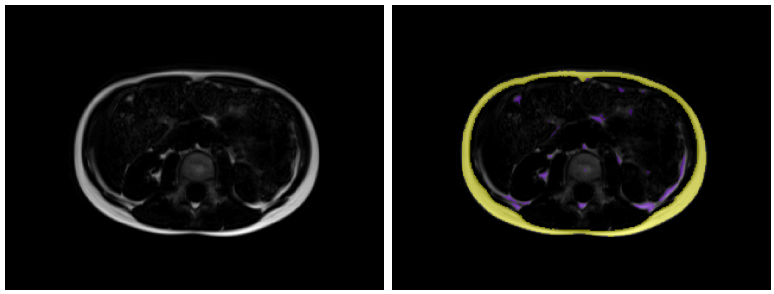
Slice 56 AV:29946 SATV:2295 VATV:59

Figure B.18: Final segmentation for Dataset #7

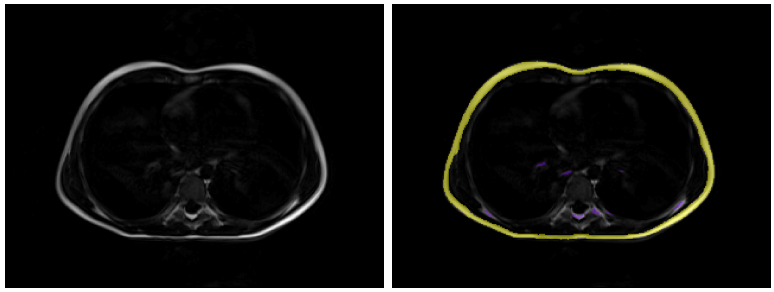
Dataset #8			
Abdomen volume:	1367427		
SAT volume:	199742	**SAT %:**	14.6
VAT volume:	19572	**VAT %:**	1.4



Slice 1 AV:25543 SATV:2152 VATV:59



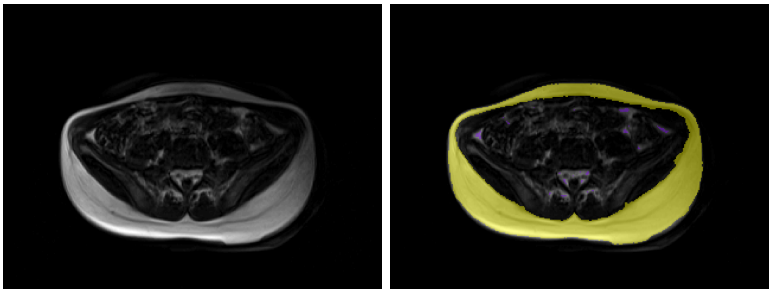
Slice 19 AV:25543 SATV:2152 VATV:59



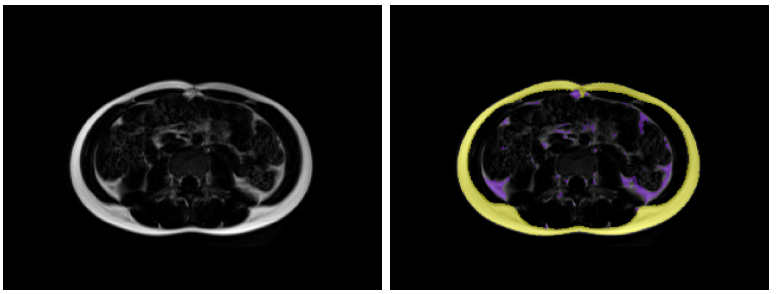
Slice 56 AV:25543 SATV:2152 VATV:59

Figure B.19: Final segmentation for Dataset #8

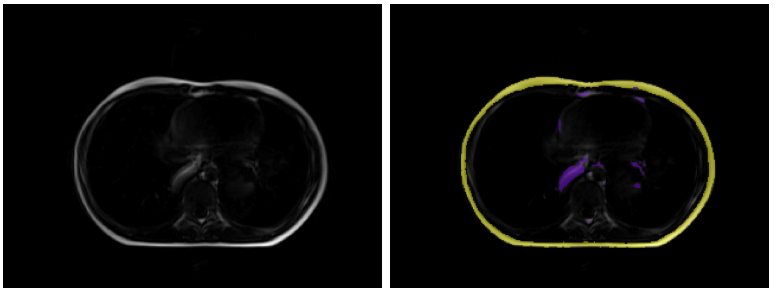
Dataset #10			
Abdomen volume:	1267415		
SAT volume:	216872	**SAT %:**	17.1
VAT volume:	23970	**VAT %:**	1.9



Slice 1 AV:25543 SATV:2152 VATV:59



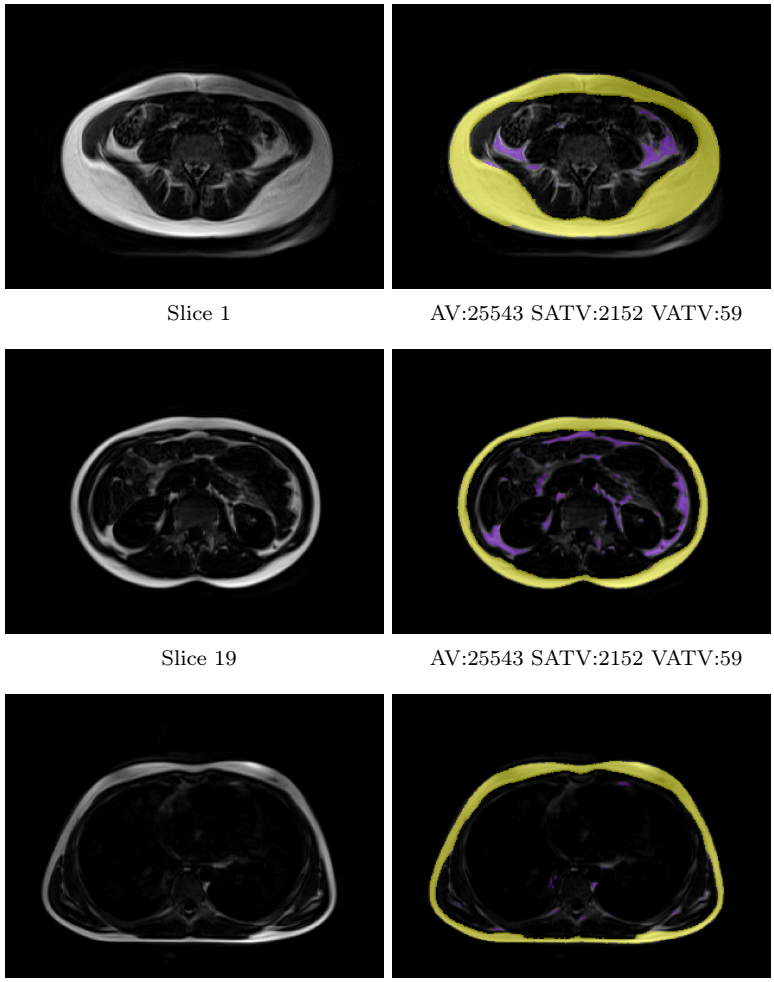
Slice 19 AV:25543 SATV:2152 VATV:59



Slice 56 AV:25543 SATV:2152 VATV:59

Figure B.20: Final segmentation for Dataset #10

Dataset #12			
Abdomen volume:	1466045		
SAT volume:	262223	**SAT %:**	17.9
VAT volume:	37788	**VAT %:**	2.6



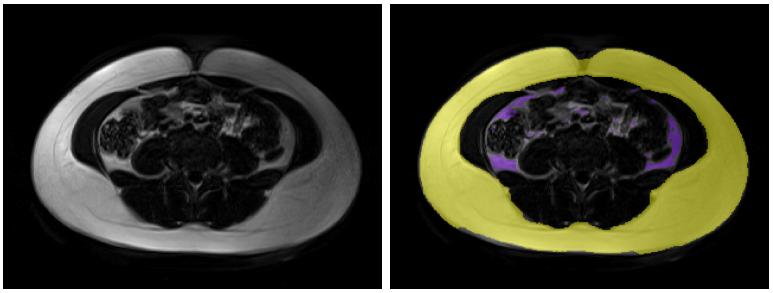
Slice 1 AV:25543 SATV:2152 VATV:59

Slice 19 AV:25543 SATV:2152 VATV:59

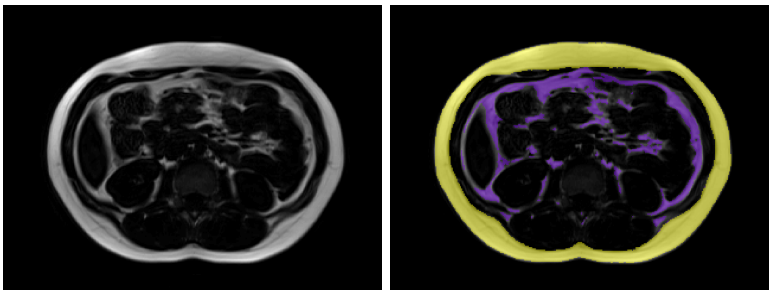
Slice 56 AV:25543 SATV:2152 VATV:59

Figure B.21: Final segmentation for Dataset #12

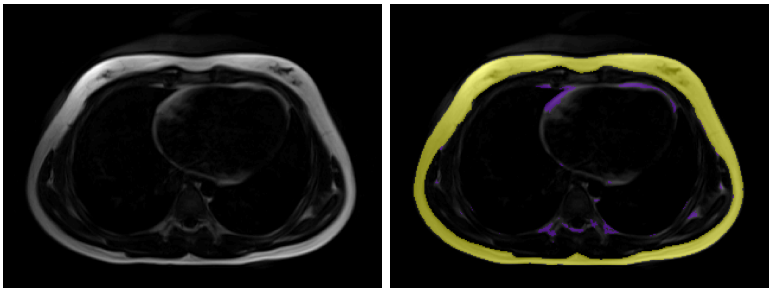
Dataset #13			
Abdomen volume:	2091722		
SAT volume:	507130	**SAT %:**	24.2
VAT volume:	96762	**VAT %:**	4.6



Slice 1 AV:25543 SATV:2152 VATV:59



Slice 19 AV:25543 SATV:2152 VATV:59



Slice 56 AV:25543 SATV:2152 VATV:59

Figure B.22: Final segmentation for Dataset #13

List of Figures

List of Tables

Bibliography

[1] M. A. Bernstein, K. F. King, and X. J. Zhou. *Handbook of MRI Pulse Sequences*. Academic Press, 1 edition, Sept. 2004.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1124–1137, Sept. 2004.

[4] E. M. Delfaut, , J. Beltran, , G. Johnson, , J. Rousseau, , X. Marchandise, , and A. a. Cotten. Fat suppression in mr imaging: Techniques and pitfalls. *Radiographics*, 19(2):373–382, 1999.

[5] W. T. Dixon. Simple proton spectroscopic imaging. *Radiology*, 153(1):189–194, Oct. 1984.

[6] G. H. Glover. Multipoint dixon technique for water and fat proton and susceptibility imaging. *Journal of Magnetic Resonance Imaging*, 1(5):521–530, 1991.

[7] G. H. Glover and E. Schneider. Three-point Dixon technique for true water/fat decomposition with B0 inhomogeneity correction. *Magnetic resonance in medicine*, 18(2):371–383, Apr. 1991.

[8] D. Greig, B. Porteous, and A. Seheult. Exact Maximum A Posteriori Estimation for Binary Images. *Royal Journal on Statistical Society*, 51(2):271–279, 1989.

[9] H. Gudbjartsson and S. Patz. The Rician distribution of noisy MRI data. *Magnetic resonance in medicine*, 34(6):910–914, Dec. 1995.

[10] IMAIOS. Signal magnitude of water and fat, 2012. http://www.imaios.com/en/e-Courses/e-MRI/Image-quality-and-artifacts/chemical-shift.

[11] J. Jensen and C. B. Anker. *Segmentation of Abdominal Adipose Tissue in MRI in a Clinical Study of Growth and Diet*. DTU Informatik, 2011.

[12] O. D. Leinhard, A. Johansson, J. Rydell, O. Smedby, F. Nyström, P. Lundberg, and M. Borga. Quantitative abdominal fat estimation using mri. In *ICPR*, pages 1–4. IEEE, 2008.

[13] K. Li, X. Wu, D. Z. Chen, and M. Sonka. Optimal surface segmentation in volumetric images – a graph-theoretic approach. *IEEE TRANS. PATTERN ANAL. MACHINE INTELL*, 28(1):119–134, 2006.

[14] T. H. Mosbech. *Fat Segmentation in Abdominal MR-scans*. DTU Informatik, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2008. Supervised by Rasmus Larsen, IMM, DTU.

[15] C. D. of Growth and Reproduction. Research project : Puberty. Website, 2009. http://www.reproduction.dk/index-filer/Page1433.htm.

[16] J. Petersen, M. Nielsen, P. Lo, Z. Saghir, A. Dirksen, and M. de Bruijne. Optimal graph based segmentation using flow lines with application to airway wall segmentation. In G. Szekely and H. K. Hahn, editors, *IPMI*, volume 6801 of *Lecture Notes in Computer Science*, pages 49–60. Springer, 2011.

[17] J. Petersen, M. Nielsen, P. Lo, Z. Saghir, A. Dirksen, and M. De Bruijne. Optimal graph based segmentation using flow lines with application to airway wall segmentation. In *Proceedings of the 22nd international conference on Information processing in medical imaging*, IPMI'11, pages 49–60, Berlin, Heidelberg, 2011. Springer-Verlag.

[18] J. Sijbers, A. den Dekker, J. V. Audekerke, M. Verhoye, and D. V. Dyck. Estimation of the noise in magnitude mr images. *Magnetic Resonance Imaging*, 16:87–90, 1998.

[19] T. E. Skinner and G. H. Glover. An extended two-point Dixon algorithm for calculating separate water, fat, and B0 images. *Magn Reson Med*, 37:628–630, 1997.

[20] Wikipedia. Rice distribution, 2006. http://en.wikipedia.org/wiki/Rice_distribution.

[21] Wikipedia. Vertebral column figure — Wikipedia, the free encyclopedia, 2006. http://en.wikipedia.org/wiki/File:Illu_vertebral_column.jpg.

[22] A. Wrangsjö, J. Pettersson, and H. Knutsson. Non-rigid registration using morphons. In H. Kalviainen, J. Parkkinen, and A. Kaarna, editors, *Image Analysis*, volume 3540 of *Lecture Notes in Computer Science*, pages 501–510. Springer Berlin / Heidelberg, 2005.

[23] X. Wu and D. Z. Chen. Optimal net surface problems with applications. In *ICALP*, pages 1029–1042, 2002.