Automatic Segmentation of Tissues in CT Images of the Pelvic Region

Martin Kardell
LiTH-IMT/BIT30-A-EX--14/521--SE

Supervisors: Alexandr Malusek, Maria Magnusson
Examiner: Örjan Smedby

November 2014
Abstract

In brachytherapy, radiation therapy is performed by placing the radiation source into or very close to the tumour. When calculating the absorbed dose, water is often used as the radiation transport and dose scoring medium for soft tissues and this leads to inaccuracies. The iterative reconstruction algorithm DIRA is under development at the Center for Medical Imaging Science and Visualization, Linköping University. DIRA uses dual-energy CT to decompose tissues into different doublets and triplets of base components for a better absorbed dose estimation. To accurately determine mass fractions of these base components for different tissues, the tissues needs to be identified in the image. The aims of this master thesis are: (i) Find an automated segmentation algorithm in CT that best segments the male pelvis. (ii) Implement a segmentation algorithm that can be used in DIRA. (iii) Implement a fully automatic segmentation algorithm.

Seven segmentation methods were tested in Matlab using images obtained from Linköping University Hospital. The methods were: active contours, atlas based registration, graph cuts, level set, region growing, thresholding and watershed. Four segmentation algorithms were selected for further analysis: phase based atlas registration, region growing, thresholding and active contours without edges. The four algorithms were combined and supplemented with other image analysis methods to form a fully automated segmentation algorithm that was implemented in DIRA.

The newly developed algorithm (named MK2014) was sufficiently stable for pelvic image segmentation with a mean computational time of 45.3 s and a mean Dice similarity coefficient of 0.925 per 512×512 image. The performance of MK2014 tested on a simplified anthropomorphic phantom in DIRA gave promising result. Additional tests with more realistic phantoms are needed to confirm the general applicability of MK2014 in DIRA.
Acknowledgments

I want to thank my supervisors Alexandr Malusek and Maria Magnusson for their input, support and guidance during this master thesis.
**Contents**

1 **Introduction** .......................................................................................................................... 9
   1.1 Computed Tomography ........................................................................................................ 9
   1.2 DIRA ................................................................................................................................. 10
   1.3 Image Segmentation ........................................................................................................... 10
   1.4 Aims ..................................................................................................................................... 11
   1.5 Thesis Structure .................................................................................................................. 12

2 **Preliminary Study: Overview of Segmentation Methods** .................................................... 13
   2.1 Thresholding ....................................................................................................................... 13
      2.1.1 Fuzzy C-means thresholding ....................................................................................... 15
   2.2 Region Growing .................................................................................................................. 17
      2.2.1 Method Description ..................................................................................................... 17
      2.2.2 Method Evaluation ..................................................................................................... 18
   2.3 Watershed ........................................................................................................................... 20
      2.3.1 Method Description ..................................................................................................... 20
      2.3.2 Method Evaluation ..................................................................................................... 20
   2.4 Graph Cuts ........................................................................................................................ 21
      2.4.1 Method Description ..................................................................................................... 21
      2.4.2 Grab Cut ...................................................................................................................... 23
      2.4.3 Grow Cut ..................................................................................................................... 26
   2.5 Level Set ............................................................................................................................. 27
      2.5.1 Method Description ..................................................................................................... 27
      2.5.2 Method Evaluation ..................................................................................................... 27
   2.6 Active Contours, Snakes ...................................................................................................... 29
      2.6.1 Method Description ..................................................................................................... 29
      2.6.2 Method Evaluation ..................................................................................................... 29
   2.7 Atlas Based Image Segmentation ......................................................................................... 33
      2.7.1 Method Description ..................................................................................................... 33
      2.7.2 Method Evaluation ..................................................................................................... 34
   2.8 Conclusions .......................................................................................................................... 36

3 **Methods** ............................................................................................................................... 37
   3.1 Histogram Matching ............................................................................................................ 37
   3.2 Removal of CT table ........................................................................................................... 39
   3.3 Thresholding ....................................................................................................................... 39
   3.4 Region Growing .................................................................................................................. 40
      3.4.1 Seed Generation .......................................................................................................... 40
      3.4.2 Region Growing Using Multi-Pixel Seeds .................................................................... 40
   3.5 Combined Thresholding and Region Growing ..................................................................... 40
   3.6 Atlas Based Image Segmentation ....................................................................................... 42
      3.6.1 Affine Registration ...................................................................................................... 42
      3.6.2 Non-Rigid Registration ............................................................................................... 42
      3.6.3 Implementation .......................................................................................................... 43
   3.7 Deformable Model .............................................................................................................. 44
   3.8 Dice Similarity Coefficient ................................................................................................. 45
   3.9 The MK2014 algorithm ...................................................................................................... 46
      3.9.1 Implementation in DIRA ........................................................................................... 48

4 **Results and Discussion** ........................................................................................................ 49
   4.1 Histogram Matching .......................................................................................................... 49
   4.2 Thresholding ....................................................................................................................... 49
1 Introduction

Cancer is group of more than 100 distinct diseases characterized by the uncontrolled growth of abnormal cells in the body (cancer, 2014). It affects not only the life expectancy of patients but also their quality of life. One way of treating cancer is the usage of radiation therapy—a method that uses ionizing radiation to destroy cancer cells (radiation therapy, 2014). To effectively target those cells with ionizing radiation, computed tomography (CT), magnetic resonance imaging (MRI) or ultrasound is used to determine the tumor location. An oncologist determines what radiation dose should be delivered to the tumor, and hospital physicists prepare an irradiation plan. Owing to the complexity of absorbed dose calculations, current radiation treatment planning (RTP) systems use simplified models. For instance in brachytherapy, where radiation is implanted directly into the tumor or tumor-bearing tissue (radiation therapy 2014), water is often used as the radiation transport and dose scoring medium, see e.g. (Beaulieu et al, 2012). These simplifications lead to inaccuracies in calculated absorbed doses. As these uncertainties may negatively affect the patients' quality of life or recurrence of the disease, the trend is to increase the accuracy of RTP methods. For instance to better characterize the transport medium, students and researchers associated with the Center for Medical Imaging Science and Visualization (CMIV) at the Linköping University (LiU) develop a software package cmiv-dira (https://code.google.com/p/cmiv-dira/), which uses dual-energy CT and image segmentation to estimate elemental composition of tissues. More information about these topics is in following sections.

1.1 Computed Tomography

Computed tomography is a technique that uses x-rays to create a tomographic image of an object. A radiation source and a detector are spinning around the object on opposite sides. The detector collects x-rays that have passed through the objects and the result is a projection. The projection depends on the rotation angle. When the detector and the radiation source have made a complete rotation around the object, the projections from the different rotation angles (a sinogram) can be used to calculate one slice of the object using an image reconstruction technique, for instance the filtered back projection. By repeating the procedure for different slices, a 3D volume of the object can be obtained. (datortomografi, 2014)

The attenuation of x-rays can be quantitatively calculated using the linear attenuation coefficient, \( \mu \) (ISO, 2009). To suppress the large dependence of the linear attenuation coefficient on photon energy, the Hounsfield value, \( H \), (CT number) is often used instead (Kalender, 2005). It is defined as

\[
H = 1000 \frac{\mu_{\text{object}} - \mu_{\text{water}}}{\mu_{\text{water}}}
\]  

(1.1)

where \( \mu_{\text{object}} \) and \( \mu_{\text{water}} \) are the linear attenuation coefficients of the object and water, respectively. The Hounsfield value of water is then 0 HU and the Hounsfield value of air is -1000 HU.
Beside the above described step-and-shoot technique called axial scanning, modern CT scanners also implement helical scanning (Kalender, 2005), where the object is moved with a constant speed during the scanning, and some of them can operate in dual-energy mode, where two sets of projections—one for the low and the other for the high x-ray tube voltage—are obtained simultaneously (Heismann, et al 2012). The classical filtered back projection is being replaced by iterative image reconstruction algorithms that permit the patient dose to be lowered and better suppress image artifacts (Hsieh, 2009). Of special interest are model based iterative image reconstruction algorithms, where the imaged object is approximated by a model.

1.2 DIRA

DIRA is a model based iterative reconstruction algorithm that uses dual-energy CT to decompose imaged tissues into base components of material doublets and triplets (Magnusson, 2011). DIRA is an automated algorithm that, after initialization, does not need any user input. The decomposed tissues can be used to calculate the absorbed dose more accurately. DIRA is a process that iterates several times to give a better result. In the current version the iteration starts with a filtered back projection followed by the tissue segmentation and then the classification. At present, DIRA only works with 2D images but it will be upgraded to work in 3D. The red arrow in figure 1 shows where in the iterative process the segmentation will be performed.

![Diagram of the DIRA algorithm](image)

Figure 1: Schematic diagram of the DIRA algorithm. The red arrow shows where in the iterative process the segmentation will be performed. Source: (Malusek et al., 2014)

More information about methods implemented in DIRA can be found in the master's thesis by Arif Muhammad (2010), Oscar Grandell (2012) and Robin Westin (2013).

1.3 Image Segmentation

Segmentation of a medical image is the separation of different structures. It can for instance be the separation of bones and soft tissues (Bankman, 2000). The separation can be done manually by a person or in an automated fashion. During the manual segmentation the person segmenting (usually a trained medical professional) uses both anatomical knowledge as well as information
from looking at the image. When the segmentation is done automatically, the methods and algorithms usually use either the image intensity or the image gradient. The automated methods are usually semi-automated since an operator needs to set initial conditions and boundaries or give feedback between iterations.

The most common approach to automatically segment the pelvis is done by only segmenting one tissue at the time, most usually only the prostate and sometimes the rectum or bladder. Recent segmentation methods use atlases or masks. The atlas is registered or deformed to match the image to be segmented. Although many articles deal with CT image segmentation, an alternative approach based on MR images is also used.

Examples on atlas registration methods are: a rigid registration in 3D on CT-images (Boydev et al., 2013), multi atlas non-rigid registration, active appearance model, probabilistic active shape model (Litjens et al., 2014), and deformable models (Chen et al., 2011), (Liu et al., 2009). The multi atlas non-rigid registration uses atlases from 50 training sets that were manually segmented. The atlases were non-rigidly registered to the prostate and formed into a single segmentation (Litjens et al., 2014). One of the deformable models was applied to the prostate using an ellipse that was deformed to best match the prostate and on this result the level set segmentation method was applied (Liu et al., 2009). The other deformable model used manually segmented training sets that were deformed to match the prostate (Chen et al., 2011).

1.4 Aims

Though many different segmentation algorithms exist, a generally applicable algorithm for segmentation of the pelvic region has not been developed yet. The aim of the thesis work is to develop a fully automated 2D segmentation method for DIRA. The approach of the thesis is to first test several semi-automated segmentation algorithms and evaluate how well they perform for the fully automated segmentation method. The input for the automated segmentation method is a 2D CT-image slice. The quality of segmentation is evaluated using the Dice Similarity Coefficient, a measurement of spatial overlap between two areas. The radiologist's work is associated with uncertainties and therefore anatomical knowledge and experience gives a better result than the segmentation result performed by an untrained person.

To reach the aims, three tasks were formulated:

- Find an automated segmentation algorithm in CT that best segments the male pelvis.
- Implement a segmentation algorithm that can be used in DIRA.
- Implement a fully automatic segmentation algorithm.

The segmented tissues will be adipose tissue, bones, muscles (gluteus maximus) and prostate. An example slice of the pelvic region of a male shows the different tissues (figure 2).
1.5 Thesis Structure

This thesis is consisting of two main parts. The first part is chapter 2, evaluates seven different segmentation methods. The methods' principles are briefly explained and implementations of these methods are tested and evaluated. Several of these seven segmentation methods are then selected for the second part of the thesis which consists of chapter 3 to chapter 5 and has a more traditional scientific report structure. The selected segmentation methods are further developed and combined to a fully automatic segmentation algorithm (MK2014) and is explained in chapter 3, Methods. Chapter 4 shows and discusses the result of the individual parts as well as the combination of MK2014. Chapter 5 is the conclusion of this thesis.

Figure 2: Segmentation of tissues. The red curves outline bone, the yellow curves outline adipose tissue, the green curve outlines the prostate and the blue curves outline the gluteus maximus muscles.
2 Preliminary Study: Overview of Segmentation Methods

This chapter briefly explains seven different segmentation methods. Some of the methods use the image intensity to segment images and others use the image gradient. The evaluation in this chapter is mostly qualitative and segmentation result as well as automation is considered. This chapter does not follow the typical methods and results structure of a scientific report and will not cover the segmentation methods in depth.

In gradient based segmentation methods some of the image attributes are called lines and edges. These lines and edges are intensity changes in certain directions. Figure 3 shows an edge (a) and a line (b) and their intensity plots ((c) and (d) respectively) in the x direction.

![Figure 3: (a) Edge and (b) line and their respective intensity plots (c), (d).](image)

2.1 Thresholding

Traditional thresholding selects pixels in a certain intensity range. The pixels with other intensities are removed. The result is a binary image. (Rogowska, 2000)

![Figure 4: Thresholding with a pre-defined range for soft tissue. (a) Original CT-image. (b) Thresholded image. The range was selected from the image intensity in the prostate. This segmentation result is useless.](image)
In figure 4 the intensity range [131,157] was determined from one pixel intensity in the prostate and an arbitrarily chosen interval length. (The intensities were normalized to the interval [0,255].) The result was far from satisfactory. The selected pixels were spread out and the result was noisy. For a more uniform and less noisy solution, Otsu's method was tested. This method segments the image into a background and a foreground by setting one threshold (Otsu thresholding, 2014). The threshold value is chosen so that the spread (variance) in the background and foreground are as small as possible, thus creating a less noisy binary image (Morse, 2000). However, Otsu's method was not very well suited for CT-images since the contrasts of several tissues were very similar and almost all the tissues were segmented as foreground, the result is seen in figure 5. For instance, compare the spread in figure 4(b) with Otsu's method in figure 5.

![Figure 5](image)

*Figure 5: Otsu's method. The Otsu method minimizes the spread in the binary image.*

For a better thresholding estimation, the histogram of the image can be analyzed and divided into more than two different regions. The histogram in figure 6 shows peaks for the air (1), adipose tissue (2), soft tissue (3) and bone (4). The four peaks in the histogram were located in four different intensity ranges: [0,9] for air (black in figure 6), [9,71] for adipose tissue (red in figure 6), [71,229] for soft tissues (yellow in figure 6) and [229,255] for bone (white in figure 6). The intensity was normalized to [0,255] from the original Hounsfield values.

Since the Hounsfield values should be approximately the same for CT-images obtained using the same x-ray tube voltage, this approach can be automated by using the same ranges for thresholding. Another possibility is to automatically find the peaks and valleys in the histogram and choose the intensity ranges from one valley to the next. A drawback with the histogram thresholding is that the different pixels do not take the neighboring pixel into account and therefore the result is noisy.
The use of a histogram for the entire image is called global thresholding (Rogowska, 2000). A different approach is to use local (adaptive) thresholding. The image is divided into overlapping sub images where the intensity is examined (Rogowska, 2000). To suppress the effect of noise on the segmentation, a method called fuzzy segmentation or fuzzy c-means can be used.

2.1.1 Fuzzy C-means thresholding

The fuzzy c-means thresholding algorithm uses a membership function that has vague membership properties: A pixel may be a member of a cluster to a certain degree, binary membership as with normal thresholding is not used, see figure 7 (Sutton et al., 2000).

![Figure 6: Histogram for CT-image with threshold ranges (here re-scaled to [0,255], left) and corresponding segmented image (right): Bone (white), adipose tissue (red) and other soft tissues (yellow).](image)

![Figure 7: Principles of fuzzy thresholding. (a) Normal thresholding where all pixels with intensity between 6 and 8 are included. (b) Fuzzy thresholding where the closer the pixels intensity is to the mean value of the range, the higher the probability is that the pixel is added to the cluster.](image)
When using fuzzy c-means, the number of ranges is selected by the user. In this particular case (figure 8) of the pelvic region, 5 ranges were used. The Fuzzy C-means thresholding algorithm clusters similar parts of the image similarly to the Otsu's method but with several clusters. To further cluster the image, median smoothing can be applied before thresholding (Sutton et al., 2000). This is similar to low-pass filtering an image and therefore some of the small details are lost.

![Figure 8: Fuzzy thresholding using (a) no smoothing and (b) median smoothing. This method is less prone to noise than regular thresholding.](image)

Fuzzy c-means thresholding algorithm according to (Aja-Fernandez 2014) is not fully automatic since the number of ranges needs to be selected. The CT-images analyzed in this work had similar properties. The number of ranges was therefore chosen once and then the algorithm was fully automated. The ranges were set automatically by the algorithm unlike the manual histogram analysis, cf. figure 6.

Thresholding is a simple segmentation algorithm that works well. The method was more complex when using the fuzzy c-means algorithm but it added increased automation. The best result from thresholding was for the segmentation of bone. The result in figure 9 was obtained through fuzzy c-means thresholding and then filling the holes of the segmented bones.
2.2 Region Growing

2.2.1 Method Description
Region growing is an intensity based segmentation method in which the region can only grow. The region starts with one or several seeds, each consisting of one or several pixels. The seeds are manually or automatically positioned inside the tissue of interest (TOI). The intensity of the pixels neighboring the seed is compared to the seed intensity and if close enough, the neighboring pixels are added to the growing region. The growing and comparison continues with the only change that the growing region mean intensity (instead of only the seed intensity) is compared to the neighboring pixel intensity. When no pixels neighboring the growing region have an intensity close enough to the mean intensity of the region, the growing stops and the algorithm is completed. (Rogowska, 2000)

The region growing method uses a parameter called variance and positions of seeds as input variables. The variance is the maximum intensity difference between the growing region and a neighboring pixel allowed for the pixel to be added. If the difference is larger than the variance, the pixel is not added to the region. The variance is manually selected. There are several ways of initiating seeds. The easiest approach is to only initiate one single seed (per TOI) consisting of one single pixel. The disadvantage of this approach is: If the selected pixel intensity is far from the mean intensity of the TOI, the seed will not grow. (Rogowska, 2000)

The advantage of using the region growing method, where the mean intensity must match the neighboring pixels, is that the borders of the TOI do not need to have well defined edges. The disadvantage is that if the image has an intensity change inside the TOI—for instance owing to image artifacts—the algorithm will not segment all parts of this region satisfactorily. (Rogowska, 2000)
2.2.2 Method Evaluation

A semi-automatic, single-seed, single-pixel region growing method with manually chosen seed and variance was tested (simple single-seeded region growing, 2014). Every segmented TOI needed its own seed. For example, all disconnected parts of bones needed their own seeds. Otherwise some parts of bone were missed, see figure 10. The bone seeds are shown in figure 11.

![Figure 10: Region growing on bones. (a) Original CT-image. (b) Resulting regions (red). One small bone (blue arrow) was missed since it did not contain a seed.](image)

![Figure 11: Seed points for region growing in adipose tissue (yellow), bone (red), prostate (blue), rectum (cyan) and muscles (magenta).](image)

As seen in figure 10 the interiors of bones were not filled by the region growing method. The solution was to fill the segmented areas using the Matlab imfill function. The Matlab imfill function fills enclosed holes in a binary image. There are two drawbacks when filling a segmented area. The first one is that if something that should not be filled has a connected contour, it is filled anyway. A good example is the segmentation of adipose tissue in figure 12:
When the holes were filled, the entire body cavity was filled. This problem was easily avoided by setting a maximum size of the holes that the algorithm filled.

The second drawback is when the segmented region does not enclose a hole. Then the hole can not be filled. An example of this can be seen in figure 13, where the interior of the pubic bone is not filled.

The result of region growing can be seen in figure 13. This result was obtained with manually chosen variances and seed points. The variances were: 20 for adipose tissue, between 20 and 35 for bones, 18 for prostate, 14 and 20 for muscles and 30 for rectum. The seed points are in figure 11. This segmentation result was good. To make region growing fully automatic, the variances and seed points have to be automatically set. It seems that the same set of variances may be used for CT-images obtained using a wide range of scanning parameters.
2.3 Watershed

2.3.1 Method Description
The watershed segmentation method is a nonparametric intensity based method. The watershed algorithm can be explained using the concept of mountains, valleys and water. The mountains represent pixels of high intensity and the valleys represent pixels of low intensity. The algorithm is initiated by filling the valleys with water. Where water from different valleys meet, a dam is built so that the valleys are separated. The dam represents a segmentation line. (Rogowska, 2000)

2.3.2 Method Evaluation
The implemented segmentation method is based on Meyers (1994) approach, where “drops” is falling on the image and the drops flow to their nearest valley. When segmenting an image such as a CT-image of the pelvic region the image was severely over-segmented, see figure 14. This was because of the large intra-variability of the intensity of the CT-image. Watershed is a nonparametric method and to get a different segmentation result with larger segmented areas, marker-controlled watershed segmentation was tested. The “marker-control” is a pre-processing method of the image where the TOI's pixels were changed to be of similar intensity and segmentation lines are added to non-TOI parts (Marker-Controlled Watershed Segmentation, 2014.).

![Figure 14: Over-segmented watershed. The result is not useful since the not all of the bones are correctly segmented.](image)

To make the different TOIs' intensities more uniform, a type of morphological opening (Mathematical morphology, 2014) followed by morphological closing (Mathematical morphology, 2014) was performed. The more uniform the intensity, the less over-segmented the image was. The uniformity of the intensity was determined by the size of a user chosen disc
shaped morphological structuring element (Mathematical morphology, 2014); a small structuring element gave an over-segmented image.

Figure 15(a) shows an under-segmented image using marker-controlled watershed with a large morphological structuring element. Only the bones were identified, but even they contained parts of soft tissues. Figure 15(b) shows an over-segmented image with a small morphological structuring element. The result was good in some places, but for instance gluteus maximus and adipose tissue were incorrectly segmented, see the lower arrow in figure 15(b). For these reasons the watershed algorithm alone was not well suited for the automated algorithm. A combination of watershed and graph cuts could provide better results but this combination was not tested. The watershed algorithm was not suited for segmenting the entire pelvic region because of the many different intensities and tissues. The size of the morphological structuring element was manually selected and for a good result a visual inspection was needed to deduct the specific size's usefulness.

2.4 Graph Cuts

2.4.1 Method Description

Graph cuts is an intensity based global segmentation algorithm. It is based on the idea that an image can be represented by a mathematical structure called graph (graph theory, 2014), where vertices represent image pixels and weights of edges (lines connecting vertices) represent differences between pixel intensities. In case of the graph cut algorithm (Boykov and Jolly, 2001), two additional vertices called the source and sink terminals are added. Vertices representing pixels are connected to these two terminals and to vertices representing neighboring

Figure 15: Marker-controlled watershed segmentation. (a) Under-segmented watershed. The result is not useful since the not all of the bones are correctly segmented. (b) Sensitive watershed segmentation with smaller areas. Even though the areas are smaller the segmentation is bad. One segmented part can contain several different types of tissue. See the red arrows in the figure.
pixels only, see figure 16. In graph theory, a graph cut is a partition of the vertices of a graph into two disjoint subsets (Weisstein, 2014). Each graph cut has a weight that equals the sum of cut edge weights. It can be shown that under quite general assumptions about the weights, the cut with the minimum weight (called minimum cut) can separate the source and sink terminals. Vertices connected to the source terminal then define the segmented object.

The initiation of the algorithm starts with a user selecting pixels in the object (TOI) and the background. The selected pixels define the hard constraints for the segmentation. The soft constraints of the algorithm can be calculated using the minimum cut. (Boykov and Jolly, 2001)

**Figure 16: Principle of graph cuts. The minimum cut is calculated using image intensity differences. The thickness of the edges from the terminals (S and T) and pixels represents the cost. T is the background terminal (seed) and S is the object terminal (seed).**

The algorithm is best illustrated using a simple 3×3 pixel image, see figure 16. One object seed and one background seed are manually chosen from the image. This selection of seeds influences the weight of each edge in the graph representing the image. The weight of an edge is dependent on the difference between the intensity of the seed (terminal) and the pixels connected through the edge. The weight also depends on a regional term $\lambda$, which multiplies the terminal-pixel weights. The regional term helps the algorithm to easier segment isolated parts with similar intensity to object seeds intensity. The thicker the edge, the bigger the weight. The image is (minimum) cut and the two terminals are separated. Graph cut relies on that the selected seeds are correctly chosen due to the intensity based properties of the algorithm.

When segmenting an image larger than 3×3 pixels, more than one object seed and one background seed may need to be selected to enhance the segmentation result (Boykov and Jolly, 2001). The seed selection makes graph cut an algorithm that needs rather much user interaction, an example of seed selection is seen in figure 17.
2.4.2 Grab Cut

A variant of the graph cuts algorithm is the grab cut algorithm. This is an iterative algorithm which uses graph cuts in intermediate steps (Vezhnevets and Konouchine, 2005). The algorithm is initialized using a rectangle as the contour. When the grab cut algorithm is finished executing its first step, a type of reinitialization (initialization based on the previous segmentation result) can be proposed and an execution very similar to the common graph cuts is performed. The grab cut algorithms final step is a border matting that smooths the borders between the object and the background. (Rother et al., 2004)

If the approximate location of the object is known, the grab cut algorithm can be automated fairly easily since the initialization contour is a rectangle (grab cut does not use the regional term $\lambda$). However, the additional reinitialization can be harder to automate since the seeds are manually selected. In some cases the reinitialization may be needed to get a good segmentation. In figure 18(b) the soft tissue directly under the bone was incorrectly classified as bone. This error was corrected using the reinitialization seen in figure 18(c).
The grab cut algorithm uses color information from the pictures to segment objects (Rother et al., 2004). Since all CT-images are in gray scale the grab cut algorithm is not well suited for this application. The drawbacks of grab cut was evident during the prostate segmentation seen in figure 19.

Figure 18: Grab cut on bones. (a) Initialization using a rectangle. (b) The result was correctly segmented bones with a small incorrectly segmented soft tissue region. (c) Reinitialization using manually selected background seeds (blue curves). (d) The final result was the correctly segmented bones. The final result was good but the algorithm needed user interaction. The background in (b)-(d) is grayed out.

The grab cut algorithm uses color information from the pictures to segment objects (Rother et al., 2004). Since all CT-images are in gray scale the grab cut algorithm is not well suited for this application. The drawbacks of grab cut was evident during the prostate segmentation seen in figure 19.
Figure 19: Grab cut on prostate. (a) Initialization using a rectangle. (b) The result was the same rectangular region. (c) The first reinitialization using manually selected background seeds (blue curve). (d) The result was an empty region. (e) The second reinitialization using manually selected background (blue curve) and object (yellow curve) seeds. (f) The final result was approximately the same as the region outlined by the yellow curve in (e). The background in (b)-(f) is grayed out.
2.4.3 Grow Cut

Another variant of the graph cuts algorithm is the grow cut algorithm. Grow cut uses like graph cuts, seeds in both the object and the background to segment objects (but unlike graph cuts, no regional term $\lambda$). From the seeds, the object and background grows outwards. When two different regions meet (background region and object region) the neighboring pixels “battle”. The winner is decided according to two criteria: (i) the distance between the pixel and corresponding seed location, and (ii) the difference between the pixel and corresponding region intensity. The winning pixel adds the other pixel to its region. These battles are continued until the borders stop moving. (Vezhnevets and Konouchine, 2005)

The grow cut was initialized by manually selected seeds. The grow cut algorithm strongly depended on the number and placement of the seeds. This dependence is seen in figures 20 and 21. The more seeds were selected, the better the segmentation result was. The reason could be because of the inhomogeneity in the background and object. Since a lot of seeds included a larger number of intensities the result was better.

![Figure 20](image)

Figure 20: (a) Grow cut initialized with a large number of object (magenta) and background (cyan) seeds. (b) The result was a correctly segmented bone.

![Figure 21](image)

Figure 21: (a) Grow cut initialized with a small number of object (magenta) and background (cyan) seeds. (b) The bone was not correctly segmented as the region contained some additional soft tissues.
2.5 Level Set

2.5.1 Method Description
Level set is a gradient based segmentation methods which uses contours. Level set segments tissues using a level set function, which is a higher dimensional function (Sethian, 1996). In the case of 2D images the level set function is a function of pixel coordinates and time. The algorithm uses the zero level of this function as a contour, which moves as the iteration time progresses. The values inside the contour are represented by negative values which approach zero for a point approaching the contour edge and change to positive values when the point moves further outside the contour (Aguiar et al., 2004). An example of the level set method is shown in figure 22.

![Level set algorithm. (a) Initialization contour (red curve). (b) Final contour (red curve). (c) Final contour (red curve) superimposed on the CT-image.](image)

The level set contour propagates in its normal (positive or negative) direction. The level set method uses the gradient of the input image to stop the contour (similar to the edge based external energy in section 2.6). The method has an advantage over the active contour method (section 2.6) as it can both split and merge its contour(s) when segmenting areas. The contour can however develop irregularities and become unstable. This is usually circumvented by a reinitialization of the algorithm using the contour result from the previous iteration as initial contour for the reinitialization. This reinitialization should be used as rarely as possible since it can lead to incorrect movement of the zero level contour. (Li et al., 2010)

The level set method has several parameters: the weighted length term $\lambda$ and the weighted area term $\alpha$, which influences the shrinking or growing of the contour. A Dirac delta function with a variable width $\varepsilon$ was used, where the width acted as a step size for the contour.

2.5.2 Method Evaluation
The Distance Regularized Level Set Evaluation algorithm was tested. The distance regularization term made the level set algorithm stable and no reinitialization during execution was needed (Li et al., 2010). The implemented algorithm used a filter to smooth the image where the filter size $n$
and the standard deviation of the filter $\sigma$ were used. The level set function was set to 
\[(1 + |\nabla I|^2)^{-1}\] where $I$ is the smoothed image intensity.

The algorithm uses a contour as initialization and this can be hard to automate. However, as long
as the entire initial contour was inside the TOI, the algorithm worked well, see figures 23 and 24. When the initialization contour was closer to the TOI contour, the result got better. Figure 24(a)
shows an initial contour derived from the atlas based registration.

![Figure 23](image1.png)  
**Figure 23:** (a) Level set using a square as initialization. (b) The resulting region (magenta) approximately covers the left gluteus maximus muscle in the CT-image. The result is good for this simplified initialization.

![Figure 24](image2.png)  
**Figure 24:** (a) Level set using an initialization contour derived from an atlas registered to the CT-image. (b) The resulting region (magenta) approximately covers the left gluteus maximus muscle in the CT-image. This more accurately calculated initialization gives a good segmentation result for an object such as the gluteus maximus.

Even if the level set algorithm was initialized automatically by using a contour from atlas based registration, the algorithm still had some drawbacks. One drawback was that the implemented
version of the algorithm could either grow or shrink. It could not “slither” to find the correct edges unlike some level set implementations. When the initialization contour was not completely inside (for the growing level set) or outside (for the shrinking level set) the TOI, the TOI was not correctly segmented. Another challenge to automate the algorithm was the large number of parameters needed to be set since each considered tissue was segmented separately. The parameters used for the result in figure 24 were $\lambda=5$, $\alpha=-1.5$, $\varepsilon=1.1$, $\sigma=2$ and $n=5$.

2.6 Active Contours, Snakes

2.6.1 Method Description

The principle behind snakes is the minimization of total energy associated to the current contour. The total energy consists of internal energy, which takes into account the shape of the contour, and external energy, which takes into account information in the image. In the original formulation by Kass et al. (1988), the total energy could also contain user-imposed constraints like springs and additional repulsion forces. These are however of interest in the interactive usage of the algorithm only and thus they are omitted in the further description. The name snake comes from the active contours movement when the contour is “searching” for the minimal energy, the contour then slithers. (Kass et al., 1988)

The internal energy comes from itself and it tries to minimize the arc length and the curvature of the contour(Kass et al., 1988). In practical implementations the internal energy is estimated by a sum of contributions from individual points approximating the contour. Two parameters $\alpha$, and $\beta$ weights the importance of the arc length and the curvature, respectively.

The external energy of an active contour comes from information in the image. To get the contour attracted to edges and lines in the image, the external energy is defined to have its minimum at high-intensity changes (high gradients) (McIntosh and Hamarneh, 2013). The external energy is comprised of three weighted parts. The parts are: $E_{\text{line}}$ representing lines, $E_{\text{edge}}$ representing edges and $E_{\text{term}}$ representing line or edge terminations; their respective weights are $w_{\text{line}}$, $w_{\text{edge}}$ and $w_{\text{term}}$ (Kass et al., 1988). $E_{\text{line}}$ is calculated from the image intensity and $E_{\text{edge}}$ from the image gradient. $E_{\text{term}}$ is calculated from a formula that reaches high values at the terminations of edges and lines. It is possible to make the snake move faster by multiplying the internal energy with a step size $\eta$. (Stragnefeldt, 2013).

2.6.2 Method Evaluation

A simplified version of the original algorithm was tested. The parameters $\beta$, $w_{\text{line}}$ and $w_{\text{term}}$ were set to zero and the new position of the contour was calculated according to figure 25 in every iteration. This algorithm was developed by the author during an image processing course at Linköping University. The algorithm was initialized using a manually drawn contour. An example can be seen in figure 26.
The external force—calculated as the gradient of the external energy—was derived from figure 26(b) or figure 26(c). For this example both images picturing the external force gave a similar result, seen in figure 26(d). The same algorithm was also used to make the contour grow instead of shrink. The only difference was that the internal force—calculated as the gradient of the internal energy—was directed outwards instead of inwards of the contour. No other changes to the algorithm were made. However, the coins previously segmented could not be segmented from the inside using the gradient in figure 27(b). This was because this image was not low-pass filtered and some of the lines on the inside the coins are detected and the contour stops at these lines. When using the low-pass filtered image (figure 27(c)) the result is more satisfactory.

Figure 26: Active contours. (a) Manual initialization. (b,c) Gradient of the image, see the text for more information. (d) Resulting contour. Source: Matlab R2014a example image

Figure 25: Minimizing of arc length when using active contours. A, B and C are points of the contour. B’ is the new location of point B. This movement is performed on all points on the contour.

Figure 27: Growing active contours. (a) Manual initialization. (b,c) Gradient of the image, see the text for more information. (d) Resulting contour. Source: Matlab R2014a example image
When this method was used on a CT-image (figure 28) the result was far from satisfactory. The gradient images of the CT-image contained a lot of lines and edges, see figure 28(c,d). This was not only due to the noise in the image but also because the pelvic area contained tissues that formed natural lines and edges in the image. Even when the algorithm segmented the bone, the tissue that had the highest and most prominent intensity, the active contour got stuck on edges that was not bone. Also some of the bones' edges were missing in the low pass filtered gradient image, as can be seen in figure 28(d).

![Figure 28: The (a) initial contour and (b) final contour for a shrinking snake with a less than satisfactory result and (c) corresponding gradient image and (d) low pass filtered gradient image.](image)

When the bone was segmented from the inside the result was similar (figure 29). The bone inhomogeneity resulted in a lot of lines and edges in the gradient images. Also some low intensity edges were missing in the low pass filtered gradient image. The elasticity parameters $\alpha$, $\beta$ and the step size $\eta$ were all set to 1 in both cases (figures 28 and 29). The algorithm was also tested with different values of the parameters but the quality of segmentation did not improve compared to figures 28 and 29 for the same reason as before.
Since the result of the tested algorithm was not very good, two additional algorithms were tested. The first additional algorithm was similar to the previous algorithm. The internal energy was calculated by minimum curvature interpolation as suggested in (Jacob et al., 2001) and not as expressed in figure 25. The external energy is a combination of gradient and region-based image energy. This contour slithered back and forth and did not strive to only shrink (or grow) as in the already tested method. The result was not much better than the result for the previously tested method, even though the contour can both grow and shrink at the same time (figure 30).

Figure 29: The (a) initial and (b) final contours for a growing snake algorithm. The final contour is not satisfactory.

Figure 30: Active contour with a “slithering” contour. The blue line shows the initial contour and the red line shows the final contour. This method gives a bad segmentation result.

This was primarily due to the fact that the external energy was calculated using similar principles as before. The gradient image and the image itself had a lot of unwanted lines and edges and the
curve did not “stick” to the correct edges. If the initialization curve could have been more exactly calculated, the result would have been better since the edges of the TOI, e.g. the bone, would have been close to the initial curve.

The second additional algorithm was a combination of snakes and level set based on (Chan and Vese, 2001). This method is promising since it not only depends on the edges in the gradient map but also uses the level set method (described in section 2.5). The result in figure 31 was not satisfactory since the initialization contour was a rectangle. A better initialization curve could give a better segmentation result.

![Figure 31: Active contours without edges on bone. (a) The initialization curve is applied to (b) the input image and the algorithm is run for 250 iterations. (c) The result is not satisfactory because some of the edges in the gradient image were not prominent enough to stop the contour from growing too far.](image)

The snakes algorithms gave a good result when the edges of the TOI were well defined. However, if the image had edges that were not part of the TOI contour the snake easily “got stuck” and stopped at these edges. A good initialization curve gave a good result since the chance of the contour getting stuck on the wrong edges was small. The “Active contours without edges”-method was also promising since it was not as dependent on the edge detection.

### 2.7 Atlas Based Image Segmentation

#### 2.7.1 Method Description

Atlas based segmentation is a gradient based segmentation method. An atlas is a reference image where the objects of interest are already segmented. The atlas is transformed to represent the tissues of interest in a CT-image. An approach to segmentation using atlases can be based on single, multi or statistical atlases. For the multi-atlas registration the atlases are usually first aligned using rigid registration (Fritscher et al., 2014). The Atlases are globally transformed by using a non-rigid or affine registration and this is done to get the locations of the atlas-objects to correspond to the images objects (Ding et al., 2005).
Due to the lack of suitable pelvic data-sets and atlases, this report only works with single atlases. The proposed implementation expresses a vector field containing information about how every point in the atlas image is going to be moved to best match the image of interest. The registration transforms the image using translation, rotation, scaling and local stretching with the help of two quadrature filters (Svensson et al., 2008). The quadrature filters phases are used to describe similarities between lines and edges in the atlas and the image (Granlund and Knutsson, 1995). Figure 32 shows how the quadrature filters calculate the phase in the image.

![Figure 32](image)

Figure 32: The phase returned by the quadrature filter describes lines and edges in the image. The lines represented by a peak and a valley in the intensity profile are at 0 and $\pi$ rad, respectively. The edges represented by an increase and decrease in the intensity profile are at $\pi/2$ and $3\pi/2$ rad, respectively.

The quadrature filters, which are several magnitudes smaller than the image (for instance 13×13 pixels compared to 127×127 pixels), are moved over the image and the lines and edges are found. From this a vector field is created. This vector field, where the corresponding vectors are representing the movements of the atlas's pixels, is used to translate the atlas to the image. (Svensson et al., 2008)

2.7.2 Method Evaluation

The atlas was created from MRI-data. The atlas was first registered to the image using an affine registration of the outer edge of the body, figure 33. The body contour was derived from the analyzed image using simple thresholding. The purpose of this was to align the image and the atlas to increase the probability that the quadrature filters would find corresponding edges and lines when performing a non-rigid registration.
To increase the speed of the affine registration, the atlas and image were reduced to a smaller size (127×127 pixels) and the smaller atlas was registered to the smaller image. The created vector field was then interpolated to the full size of the image and the atlas was transformed. The next step of the algorithm was to perform non-rigid transform using the local phase quadrature filter method described previously.

Figure 33: Image and Atlas (a) before and (c) after affine registration. The operation was performed on (b) an image and atlas with reduced sizes to speed up the calculations. The pink and green colors represent the atlas and image, respectively. The white parts in the image represent the overlap of the atlas and image.

To increase the speed of the affine registration, the atlas and image were reduced to a smaller size (127×127 pixels) and the smaller atlas was registered to the smaller image. The created vector field was then interpolated to the full size of the image and the atlas was transformed. The next step of the algorithm was to perform non-rigid transform using the local phase quadrature filter method described previously.

Figure 34 shows three stages of the registration. Figure 34(a) shows the original unchanged atlas (in pink) on top of the original image (in green). Figure 34(b) shows the same thing after the initial affine registration. Figure 34(c) shows the result after the non-rigid registration.

Atlas based registration gave good results not only for the presented images but also for other analyzed images of the pelvic region. However it was important to use an atlas that well

Figure 34: Image and atlas (a) before registration, (b) after affine registration and (c) after non-rigid registration. The quality of registration increases from left to right. The pink and green colors represent the atlas and the original CT-image, respectively.
represented tissue structures in the region. The atlas had to be selected manually, the rest of the algorithm was fully automatic. To further enhance the result of the atlas based registration, a deformable model can be applied on the image using the registered atlas as an initial contour (Ding et al., 2005). The methods that may be used are active contours and level set.

2.8 Conclusions
The region growing and thresholding methods gave the best result for segmentation of bones and adipose tissue. The thresholding method is easy to automate since the approximate intensity ranges for different tissues are the same for CT-images obtained using the same x-ray tube voltage. Region Growing uses seeds and variances as parameters. The variances of tissues depend on variations in CT scanning parameters and patient sizes to a small degree only and thus can be set once for each type of tissue. A multi pixel seed gave better result than a single pixel seed and is therefore better suited for the automated method.

The prostate and muscles required more complex segmentation algorithms since the intensity values of all soft tissues (except adipose tissue) are similar. The similarity of intensity values was a disadvantage to the intensity based segmentation methods and therefore only gradient based segmentation methods were selected for these tissues. Of the latter the atlas based image segmentation method was the best. This method, combined with a deformable model, such as level set and/or active contours was used in the next stage, see chapter 3.

In summary, the thresholding and region growing methods were selected for the bone and adipose tissue segmentation. The atlas based image segmentation and “active contours without edges” methods were selected for the prostate and muscle segmentation.
3 Methods

Five anonymized CT-images from three different patients were retrieved from the PACS-system and imported to Matlab in the DICOM file format. The CT-scanner parameters were: The tube voltage of 120 kV and slice thickness of 5 mm. The images depict the male pelvis at the approximate height of the hip joints and their size was 512×512 pixels. The image intensity was converted to 8 bit integers and thus ranged from 0 to 255.

Segmentation methods described in sections 3.1-3.7 were combined into the proposed MK2014 algorithm (section 3.9).

The quality of segmentation was assessed using the Dice similarity coefficient (section 3.8). Owing to the lack of ground truth segmented images only one image was analyzed. The ground truth is a manual segmentation performed by a radiologist and it is compared with the automated segmentation for a quantitative result.

The MK2014 algorithm was evaluated in DIRA using a simplified anthropomorphic phantom (The data processing was done in Matlab R2014a using a PC with Intel(R) Core(TM) i5-4200U CPU and 8 GB RAM.

3.1 Histogram Matching

The purpose of histogram matching is to adjust an input image's histogram so that the cumulative distribution functions (CDF) of the input and reference image intensities match (Histogram Adjustments in MATLAB, 2014),(Gonzalez and Woods, 2002). The reason for using this image analysis method is to give a better and more stable result to the thresholding and region growing methods. The output image pixel intensities can be calculated from the two CDFs, see figure 35.

![Figure 35: The CDFs for the input and reference images. For a given y-value, the x-value of the input image is transformed to the corresponding x-value of the reference image. This transformation is done for all y-values.](image-url)
Since the CDFs match, the pixel value of the image can be found using a look-up table. This look-up table is represented via a transfer curve, see figure 36.

![Transfer curve converting the intensities of the input image to the intensities of the output image.](image)

**Figure 36:** Transfer curve converting the intensities of the input image to the intensities of the output image.

This transfer curve is used to change the input image so that the histograms of the input and reference images are approximately the same, see figure 37. Figure 38 shows the reference image, an input image and the output image.

![Histograms for unmatched and matched images.](image)

**Figure 37:** (a): histogram for unmatched image (magenta) and reference image (blue). (b): histogram for matched image (magenta) and reference image (blue).
3.2 Removal of CT table

The CT table, seen as two curved lines in figure 38, needs to be removed to not give false results for the automated segmentation. The table was removed by first creating a binary image where pixel values of all tissues and the CT table were set to 1 and the pixel values of air were set to 0. The algorithm looked for “islands” in the image and if the “island” was small enough, it was removed.

3.3 Thresholding

The thresholding used in the implementation of the algorithm was regular thresholding with fixed ranges. The image intensities were normalized and ranged from 0 to 255 instead of using the Hounsfield scale where air is -1000 HU, adipose tissue is -100 to -50 HU and compact bone is about +1000 HU or more. Since the maximum Hounsfield value can differ for different images while the scale in the algorithm always goes from 0 to 255, the normalized tissue ranges may differ between CT-images. This problem was solved by using the histogram matching described earlier in this section. The normalized ranges for the reference image (figure 38(a)) are in table 1.

Table 1: Expected Hounsfield values and corresponding normalized ranges for selected tissues. The Hounsfield values were taken from selected DICOM-images.

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Expected Hounsfield value</th>
<th>Normalized range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Air</td>
<td>-1000</td>
<td>[0,6]</td>
</tr>
<tr>
<td>Adipose Tissue</td>
<td>[-100,-50]</td>
<td>[6,75]</td>
</tr>
<tr>
<td>Compact Bone</td>
<td>&gt;1000</td>
<td>[180,255]</td>
</tr>
</tbody>
</table>

The segmentation mask is defined as

\[
g(x, y) = \begin{cases} 
1 & \text{if } (T_i < f(x, y) < T_h) \\
0 & \text{else}
\end{cases}
\]  

(3.1)
where \( f \) is the input image, \( g \) is the output mask, \( T_l \) and \( T_h \) are low and high threshold values, respectively, and \( x \) and \( y \) are coordinates. Equation (3.1) is used for every threshold range. In the implemented method it is used for the segmentation of adipose tissue and bone.

### 3.4 Region Growing

The selected region growing algorithm is a multi-pixel multi-seed method with variable intensity range.

#### 3.4.1 Seed Generation

Seeds for image segmentation are usually manually set. To automate the process, the locations of tissues are needed. The information so far acquired by the automated algorithm is through thresholding. The result from thresholding (bones and adipose tissue) is used to generate seeds. The seed generation method first removes small noise artifacts from the thresholding results. Next, it erodes the images to only keep as few pixels as possible from the segmented areas. For every 4-connected component in the binary image, two pixels are selected as seeds. For instance for bones, every bone part that is not connected to another bone part gets two single pixel seeds. These seeds are dilated to create multi-pixel seeds. Because of the high stability when using region growing on adipose tissue and to minimize computational time, only one seed is selected for the adipose tissue.

#### 3.4.2 Region Growing Using Multi-Pixel Seeds

The region growing method is explained in section 2.5. The only difference here is that it has been modified to use multi-pixel seeds. To automate the algorithm, both the variances and the seeds need to be automatically set. The seed generation is automated as earlier explained in this chapter and the variances are set to one fixed value (per tissue type). The variances can be fixed since the input image intensity is matched to the reference image intensity using the histogram matching.

The region growing method uses two seeds per separate bone part and the seeds initialize each own region growing iteration. Each region grows independently and a union both regions is taken as the result. This is to increase the stability of the segmentation and increase the number of pixels segmented as bone.

### 3.5 Combined Thresholding and Region Growing

To further enhance the thresholding and region growing segmentation results, the methods can be combined. The combination of thresholding and region growing was only implemented for the bones. When segmenting bones with the two methods, the bone marrow was usually not included in the segmentation result. The bone marrow is part of the bones and in general it should be included. The cavities containing bone marrow need to be encircled by bone for the algorithm to be able to differentiate between bone marrow and soft tissues. When segmenting in
2D using region growing and thresholding, some of the edges of the bone were missing due to vein openings, noise or other artifacts, see figure 39.

![Figure 39: Thesholding of bones. The arrows point at some of the holes that should be closed so that the bone cavities can be filled.](image)

When uniting the region growing and thresholding results, some of the cavities were encircled and filled. However, not all the cavities were encircled and so a different image analysis approach was used: morphological skeleton and morphological closing. A morphological skeleton is the result from eroding a binary image and only leaving the medial axis. The remaining pixels leave an unbroken skeleton of every component. Morphological closing is applied to the image skeleton to encircle the holes. Morphological closing is a dilation followed by erosion. These operations encircle most of the skeleton's cavities. The closed skeleton is united with the region growing-thresholding union and the bone cavities are filled.

![Figure 40: Morphological skeleton and morphological closing. (a) Morphological skeleton of bones. (b) Morphological closing of skeleton. (c) Result when combining morphological skeleton, morphological closing, region growing and thresholding.](image)
For adipose tissue, the region growing result was used unaltered and all the smallest holes were filled as explained in section 2.5.

3.6 Atlas Based Image Segmentation

Since the soft tissues intensities were close to each other, the region growing and thresholding was not used for the segmentation of muscles and prostate. Gradient based methods were instead used. The first implemented gradient based method was the atlas based image registration. The method started with affine registration and continued with non-rigid registration.

3.6.1 Affine Registration

A modified non-rigid registration that resembles affine registration was used. Affine transformation is a one-to-one point mapping, where for instance straight and parallel lines remain straight and parallel after transformation (Affine transformation, 2014). Affine transformation allows for global translation, rotation, scaling and shearing of an image. Local stretching and deformation is not allowed. Non-rigid transformation allows local translation, rotation, scaling and stretching. The implemented affine registration used the same principles as the non-rigid registration, except that local transformations did not occur inside the body. For the sake of brevity, this modified non-rigid registration is referred to as affine registration in this report.

3.6.2 Non-Rigid Registration

The non-rigid registration was briefly explained in section 2.7.1 and this chapter describes it in greater detail. The description is based on (Granlund and Knutsson, 1995) and (Svensson et al., 2008).

The transformation is calculated by adding a vector matrix $v$ to the atlas image matrix so that the difference between the transformed atlas image and the analyzed image defined as

$$
\varepsilon^2 = \|I_2(x + v(x)) - I_1(x)\|^2
$$

is minimized. When looking for similarities in the two images it is possible to analyze the phases of two quadrature filters' responses (Granlund and Knutsson, 1995). The filter responses are imaginary when the filter hits a line and real when it hits an edge in the image, see figure 32.

The phase gradient for the image is estimated as

$$
\nabla \varphi = \begin{bmatrix} \nabla_x \varphi \\ \nabla_y \varphi \end{bmatrix} = \begin{bmatrix} \arg[q_1(x + 1, y)q_1^*(x - 1, y) + q_2(x + 1, y)q_2^*(x - 1, y)] \\ \arg[q_1(x, y + 1)q_1^*(x, y - 1) + q_2(x, y + 1)q_2^*(x, y - 1)] \end{bmatrix}
$$

(3.3)

where $q_1$ and $q_2$ are the responses of the two quadrature filters, $x$ and $y$ are coordinates in the image, “*” stands for the complex conjugate and “arg” stands for the argument.

A quadrature filter can only identify lines and edges in certain directions and therefore more than one filter needs to be used. This method uses two quadrature filters that cover different directions.
in the image. Since the filters cover different directions, they are not equally good at describing the signals in different directions. To know which filter to trust for each image pixel, the security measure \( c \) is used. It is defined as

\[
 c = \sqrt{q_1 q_2} \cos \left( \frac{\Delta_{t\varphi}}{2} \right)^2 \tag{3.4}
\]

where \( \Delta_{t\varphi} \) is the change in quadrature filter phase over one time instance:

\[
\Delta_{t\varphi} = \varphi(x, t + 1) - \varphi(x, t) = \varphi_1 - \varphi_2 = \arg(q_2 q_1^*) \tag{3.5}
\]

In equation (3.5) \( x \) is the position of the pixel and \( t \) defines the time instance.

The equation

\[
\nabla \varphi_2^T B(x)p \approx \varphi_1(x) - \varphi_2(x) \tag{3.6}
\]

gives a parameter \( p \) as a function of \( x \) which is further used to calculate a movement field for the image (Svensson et al., 2008). The symbol \( \nabla \varphi_2^T \) denotes a transposed gradient vector. The base matrix \( B \) (for 2D images) for the parameter vector \( p \) is defined as (Svensson et al., 2008)

\[
 B(x) = \begin{bmatrix} 1 & 0 & x & y & 0 & 0 \\ 0 & 1 & 0 & 0 & x & y \end{bmatrix} \tag{3.7}
\]

Equation (3.6) does not use the security measure and thus the two quadrature filter responses will incorrectly contribute to the transformation. To mitigate the problem the security measure can be added as

\[
\underbrace{c(x, k) \nabla \varphi_2^T B(x)}_{W} p \approx \underbrace{c(x, k) (\varphi_1(x) - \varphi_2(x))}_{\tilde{A}} b \tag{3.8}
\]

where \( W \) and \( \tilde{A} \) are matrices and \( b \) is a vector. This parameter vector \( p \) minimizes \( \epsilon \) and is given by the weighted least square solution

\[
p = (\tilde{A}^T W^T W \tilde{A})^{-1} \tilde{A}^T W^T W b \tag{3.9}
\]

The entire translation process is iterated until the difference between the two images is sufficiently small. How large a difference is allowed is a trade-off between accuracy and speed.

### 3.6.3 Implementation

The registration part of the implemented atlas based segmentation was performed the same way for both affine and non-rigid registration. This was possible due to alterations to the atlas and image for the affine registration. Before the algorithm registered the atlas, the atlas and the image were altered so the only edges and lines in the images were the borders of the bodies. This created an effect of non-rigid translation at the atlas borders, but inside the atlas only affine translation occurred. During the affine registration, the atlas and the image were aligned.
Since there is a big difference between different patients muscle mass and amount of fat, the non-rigid registration was only performed on the bones. The bones from the atlas were registered to the already segmented bones. When the atlas bones had been non-rigidly registered and translated, the created movement field was applied to the entire atlas, thus moving the muscles and prostate. For extra stability, the overlap of the segmented bones and the atlas bones from both the affine and non-rigid registration was compared and only the best result was used.

After the registration of the muscles and prostate, a deformable model was applied to the muscles and prostate with the atlas segmentation as initial contours.

3.7 Deformable Model

The deformable model is the last part of the automated algorithm and the deformable model is the “Active Contours without Edges” by Chan and Vese. This model uses a contour that can both grow and shrink according to an energy minimizing formula. The contour is however dependent on a level set formulation that will stop the contour at the boundaries of the TOI. The level set part of the algorithm makes the method less dependent on only the image gradient, as oppose to traditional snake algorithms. (Chan and Vese, 2001)

The method uses two forces that are opposing each other. One force tries to shrink the contour and the other tries to make the contour grow

$$F_{\text{tot}} = F_1(C) + F_2(C) = \int_{C_i} |u_0(x, y) - c_1|^2 \, dx\, dy + \int_{C_o} |u_0(x, y) - c_2|^2 \, dx\, dy$$

(3.10)

where $u_0$ is the image, $C_i$ and $C_o$ are the regions inside and outside, respectively, the $C$ contour and $c_1$ and $c_2$ are constants.

To minimize the forces, the contour should be the same as the object borders $C_0$

$$F_1(C_0) + F_2(C_0) \approx 0$$

(3.11)

This is explained using the four examples: (i) If the curve is outside the object, $F_1>0$ and $F_2<0$. (ii) If the curve is inside of the object it will be the other way around, $F_1<0$ and $F_2>0$. (iii) If the contour is both inside and outside the object, $F_1>0$ and $F_2>0$. (iv) The last example is of course the energy minimization when $C=C_0$. This results in $F_1\approx0$ and $F_2\approx0$ and therefore the total energy $F_{\text{tot}}\approx0$. A visualization of this example can be seen in figure 41. (Chan and Vese, 2001)
When the algorithm is running, the constants $c_1$ and $c_2$ are recalculated once per iteration. This is done using a Heaviside function $H$ and a Lipschitz function $\phi$

$$c_1(\phi) = \frac{\int_{\Omega} u_0(x, y) H(\phi(x, y)) \, dx \, dy}{\int_{\Omega} H(\phi(x, y)) \, dx \, dy}$$  \hspace{1cm} (3.12)$$

$$c_2(\phi) = \frac{\int_{\Omega} u_0(x, y) [1 - H(\phi(x, y))] \, dx \, dy}{\int_{\Omega} [1 - H(\phi(x, y))] \, dx \, dy}$$  \hspace{1cm} (3.13)$$

where $u_0$ is the image. For more information, see “Active Contour Without Edges” (Chan and Vese, 2001).

3.8 Dice Similarity Coefficient

To be able to measure the accuracy of the automated segmentation algorithm, a ground truth is needed. When the ground truth is obtained, the accuracy can be calculated using the Dice similarity coefficient. The DSC is commonly used when evaluating the performance of
segmentation methods. The DSC measures the spatial overlap between the ground truth and the automated segmentation, where the ground truth and automated segmentation are binary images. The value for the DSC is between 0 (no overlap) and 1 (perfect overlap). The Dice Similarity Coefficient is calculated as

$$D = \frac{2|A \cap B|}{|A \cap B| + |A \cup B|}$$ (3.14)

where \(A\) is the automatically segmented pixels and \(B\) is the ground truth pixels. (Babalola et al., 2008)

The Dice similarity coefficient can also be expressed as a Venn diagram, see figure 42.

![Venn diagram showing spatial overlap when using the Dice similarity coefficient](image)

Figure 42: Venn diagram showing spatial overlap when using the Dice similarity coefficient

3.9 The MK2014 algorithm

The methods described in sections 3.1-3.7 were combined to form the automated algorithm MK2014. The inputs for MK2014 are: the analyzed image, one reference image for the histogram matching and one atlas for the image registration. The presented reference image can be used for most CT-slices in the pelvic region while the atlas image can only be used for several transverse slices in the pelvic region as the anatomy rapidly changes from slice to slice there.

MK2014 is executed in the following steps

1. Histogram matching
2. Thresholding image to segment bones and adipose tissue
3. Seed generation using thresholding result
4. Region growing initialized using seeds to segment bones and adipose tissue
5. Region growing and thresholding results are combined with morphological skeleton and morphological closing

6. Affine registration performed on image using atlas

7. Non-rigid registration performed on image using atlas

8. Registered contours deformed to fit prostate and muscles

Table 2 shows which methods were used in which tissues and figure 43 shows the flowcharts for the algorithm for the different tissues.

**Table 2: Methods used on different tissues.**

<table>
<thead>
<tr>
<th>Method</th>
<th>Bones</th>
<th>Adipose tissue</th>
<th>Prostate</th>
<th>Muscles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Histogram matching</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Removal of CT-table</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Thresholding</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Region Growing</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Combined TH and RG</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Affine registration</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Non-rigid registration</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Deformable model</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

*Figure 43: Flowcharts for the algorithms different parts*
3.9.1 Implementation in DIRA
The MK2014 algorithm was implemented in DIRA and evaluated using a phantom derived from slice number 113 of the ICRP 110 male voxel phantom (Malusek et al., 2014). The phantom consisted of: compact bone, bone marrow, muscle, adipose tissue and calcified prostate tissue. The segmentation was done to: (i) Bone which included compact bone and bone marrow. (ii) Soft tissue which included adipose tissue, muscles and calcified prostate.
4 Results and Discussion

The Dice similarity coefficients presented in this chapter were calculated from a single slice only. In other cases quality of segmentation was assessed visually.

Figures 44(b) and 44(c) show the CT-image from which the ground truth was segmented. The DSC was calculated from the segmentation result (based on the CT-image seen in figure 44(b)) and the ground truth.

4.1 Histogram Matching

The histogram matching took a short time to execute and gave a stable result. The result can be seen in figure 44.

In general the histogram matching creates a smaller interdiffernce between images' intensities and gives a more uniform and better result. Even though the histogram matching is rather insensitive to interdifferences in patients, the matching can however fail if the input image is not showing the pelvic region or the pelvic region in the image is composed in a very different way than in the reference image. For instance if the patient is obese, the input image will have a lot of adipose tissue which the reference image is lacking. The images will then not match in a preferable way, which in turn will negatively affect the thresholding and region growing results.

4.2 Thresholding

The thresholding method was fast, easy to implement and gave stable results. The latter was mainly because the histogram matching was used in the first step. The thresholded tissues were bones and adipose tissue because of the similar intensity of the other types of tissues in the pelvic region.

A pure thresholding method is sensitive to intensity differences between images since the thresholding ranges are set. The histogram matching mostly took care of this problem and

---

Figure 44: (a): reference image. (b): unmatched image. (c): matched image. (c) is a good matching of (a).
normalized the tissue intensities but if the histogram matching would fail, the probability that the thresholding also would fail is high. The thresholding method is insensitive to the location, shape and size of tissues.

The resulting DSC of this method was 0.9079 for bones, 0.8309 for adipose tissue and the mean computation time was 0.1 s. Figure 46(b) shows the result.

4.3 Region Growing

The seed positions in this implementation were automatically determined from the thresholded image, which in turn depended on the histogram matching. It was important that the seeds were positioned on the bones and inside the adipose tissue. If the seeds had been outside of these tissues, the region growing method would have segmented the wrong kind of tissue, or even air. When the seed generation was tested using the available CT-slices, the seeds always had the correct placement as long as the image was histogram matched. A different approach of automatic seed generation could have been to generate the seeds from an atlas registered to the image. The seed placement for three different slices can be seen in figure 45.

![Figure 45: Seed placement for bones (red dots) and adipose tissue (green dots) for slices from three different males.](image)

![Figure 46: (a): region growing of bones (red) and adipose tissue (blue). (b): thresholding of bones (red) and adipose tissue (blue).](image)
The region growing depended on the variance, whose value had to be set properly. If the variance was too low, the entire TOI would not be segmented because of possible intensity differences inside the TOI. A too large variance would make the region grow to tissues outside the TOI. Only the tissues (bones, adipose) with intensities far from other tissues intensities were chosen for the region growing algorithm. This made it possible to choose larger variances that would segment a larger part of the TOIs correctly.

The region growing operations took from 7.3 to 11.1 s, with a mean of 8.9 s. The resulting DSCs for the region growing result were 0.8988 for the bones and 0.9108 for adipose tissue. Figure 46(a) shows the segmented tissues.

4.4 Combining Thresholding and Region Growing

The combination of thresholding and region growing was used for bones only. For adipose tissue, this approach resulted in a lower Dice similarity coefficient. The best result for the adipose tissue was when only the region growing method was used for segmenting. However, the thresholding of adipose tissues was needed for automatic seed generation.

The union of thresholding and region growing for bone segmentation gave the DSC of 0.9218, which was a higher value than for either region growing or thresholding alone. To further enhance the result, the morphological skeleton and morphological closing was used in combination with the thresholding-region growing union. This solution segmented more of the bone and the DSC value rose. A drawback with this solution was that some non bone areas that originally were not encircled became encircled and filled. This added false positive area was usually smaller than the added true positive area and the DSC increased. The DSC value was then 0.9570 and gave the best result for bones. The results from the different segmentations of adipose tissue and bones are shown in figures 47 and 48, respectively.

Figure 47: Adipose tissue using (a) thresholding and (b) region growing. Red lines define the automated segmentation and blue lines define the ground truth. The segmentation result is better for the region growing method.
Figure 48(a, b) shows the bone segmentation from thresholding and region growing, respectively. The thresholding method segments the uppermost part of the bones better than region growing, but the region growing more accurately segments the tail bone (the lowermost part of the bones). The two methods (figure 48(c)) combine these two different advantages and the combined result with the morphological closing and morphological skeleton is shown in figure 48(d).

*Figure 48: (a) Segmented bones using thresholding, (b) region growing, thresholding united with region growing (c) without and (d) with morphological skeleton and morphological closing. Red lines define the automated segmentation and blue lines define the ground truth. The best result is (d).*
4.5 Atlas Based Segmentation

The atlas based segmentation was the part of the MK2014 algorithm that took the longest time: between 14.1 and 46.9 s. The large time variation was due to the difference in body placement in the CT-images and interbody differences. The atlas was optimized for only a small transverse range in the pelvic region and the algorithm took longer time to compute when the selected slice was not in this range. The mean computation time was 23.1 s. This time includes both the affine and non-rigid registrations.

The affine registration result was used for the muscles and the non-rigid registration result was used for the prostate. The reason was that the variations in the patients' fat and muscle composition changed the placement of the muscles a lot. The non-rigid result was not as good for muscles as for bones and prostate. The bones and prostate have similar look and location for different patients. The non-rigid registration therefore moves the atlas prostate to a more accurate position than when only using the affine registration. When an atlas was created to segment tissues, some assumptions had to be made. The tissue composition of the atlas had to be assumed and the registration probably would not give a good result if the patient's tissue composition differed a lot from that of the atlas.

The DSC values for the affine registration was 0.7886 for the prostate and 0.8899 for the muscles. For the non-rigid registration the DSC was 0.8065 for the prostate, which was an improvement compared to the affine registration.

4.6 Deformable Models

The simple non-rigidly translated atlas prostate was not a perfect fit so after this operation, a deformable model was applied. The computation time for the muscles and prostate segmentation was between 10.2 and 10.7 s with a mean of 10.6 s. Since the initial contour was derived from the atlas based segmentation, the deformable model result was dependent on the performance of the atlas based segmentation. The deformable model increased its accuracy when the initial contour was closer to the TOI contour. If the atlas based image segmentation fails and the contour for the deformable model is initialized at the wrong place in the image, the result would be bad. The DSC was 0.9324 and 0.9001 for the prostate and the muscles, respectively, after application of the deformable model.
The reason a small part of bone was segmented when segmenting the left muscle (see figure 49(b)) was that the automatically generated initial contour is unwantedly including a small part of the bone. The prostate segmentation can be seen in figure 50.

4.7 MK2014 Algorithm

The final automated algorithm took between 35.8 and 68.9 s to run, with a mean of 45.3 s. The manual ground truth segmentation performed by a radiologist took about 60 minutes to complete. The manual segmentation was performed in an environment not completely familiar to the radiologist, which may have slowed down the manual segmentation. The time difference was very large and the automated segmentation did not need human supervision when running.

The MK2014 algorithm is a combination of several segmentation and image analysis methods that uses several parameters. Some of the parameters needed to be set so that the combined
methods allowed for a large variation between images. This potentially made MK2014 more unstable with a higher risk of false positive results.

When the algorithm was developed, only a small number of CT-images in the relevant transverse range was available for testing and only one CT-image with a ground truth was available for a quantitative result. The algorithm was tested on patients that were entirely in the field of view, were not obese, did not have any tissues removed or did not have any materials implanted in the body (e.g. a titanium hip joint). For a general applicability, more the MK2014 needs to be tested using more CT-images.

The MK2014 algorithm is presently the only (to the authors knowledge) fully automatic segmentation that segments the male pelvic region for decomposition of tissues.

The DSC value when comparing the automatic segmentation result and the ground truth were: 0.957 for bones, 0.911 for adipose tissue, 0.900 for muscles and 0.932 for prostate with a mean Dice similarity coefficient of 0.925. The result is shown in figure 51. Figure 52 shows the individual parts of the automated segmentation result compared with the ground truth.

![Figure 51](image)

*Figure 51: Final result using the automated algorithm compared with the ground truth. The green and magenta areas are were the ground truth and automated algorithm results differ.*
4.7.1 Implementation in DIRA

Raw projections from patient CT-scans were not available for processing in DIRA and so a simple mathematical phantom was used, see figure 53. A more complex phantom was proposed but there was not enough time to implement this phantom in the limited time frame. The proposed phantom would have been derived from anonymized patient CT-images.

The computation time for the segmentation was approximately 30 s per one DIRA iteration. The DSCs were 0.992 and 0.989 for bones and soft tissues, respectively.

Figure 52: segmentation results compared with the ground truth. Red lines define the automated segmentation and blue lines define the ground truth.
Figure 53: Result using the DIRA phantom. (a) Linear attenuation coefficients of the phantom. (b) Segmented bone and bone marrow are displayed in green color; the underlying white color from the compact bone causes some of the correctly segmented pixels to appear white. (c) Segmented soft tissues are displayed in green color.
5 Conclusion

This report tested seven different types of algorithms for segmentation of CT-images in the male pelvic region. Four of the seven methods showed promising results and were selected for further development of the combined algorithm MK2014. The four different methods were: thresholding, region growing, atlas based registration and the deformable model active contours without edges. These methods were applied to different tissues of the analyzed image with satisfactory results.

The methods that worked best were the atlas based registration and the deformable model (applied in this order). The resulting Dice similarity coefficients, when comparing the automatic segmentation result and the ground truth, were: 0.957 for bones, 0.911 for adipose tissue, 0.900 for muscles and 0.932 for prostate. The MK2014 algorithm was very stable and gave a satisfactory result for every analyzed CT-image (five images were analyzed and only one image had the information about ground truth available). Clearly more work needs to be done to test the algorithm on images with known ground truth segmentation. A drawback of the algorithm was the long computation time of the atlas based registration algorithm.

The MK2014 algorithm was implemented in DIKA using a simplified anthropomorphic phantom. The result was promising. Additional tests with more realistic phantoms and real CT-data are needed.
6 Future Work

At present DIRA can only reconstruct and segment images in 2D. The next step is to extend DIRA's functionality to 3D. This will remove some of the problems with the male pelvic region segmentation since the anatomy in adjacent slices will be known. For instance when a part of the outermost bone in the hip joint is missing in the 2D slice (due to noise, image artifacts, or a vein passage), the segmentation of this bone does not give a good result. This was explained in section 3.6 and figure 39. This problem can be solved in 3D by using information from the adjacent slices.

When applying the deformable model on the muscles and bone is included in the initial contour, it will be possible to remove the bone part from this contour by using the already segmented bone.

The current implementation of DIRA can use any number of doublets and triplets for material decomposition. The usage of one doublet for bones and one triplet for soft tissues is, however, the only configuration that has been tested so far. The MK2014 algorithm has the capability to segment a CT image to four different tissues (adipose tissue, bone, muscles and prostate). This feature can be used to extend the number of soft tissue triplets in DIRA from one to three.

In the future DIRA will process data from multiple modalities, for instance CT and MR imaging. Figures 54 and 55 show MR images registered to CT images. These figures show the CT image at 120 kV and the MR image with T1 weighting of the same patient. The combination of different modalities will give additional information that can be used for tissue segmentation. For instance the well visible prostate region in the MR image shown in figure 54(b) can be transferred to the CT image shown in figure 54(a). Some problems that could arise with the combination of the different modalities are the differences in the anatomy of the patients due to time passed between CT and MR scans (e.g. bowel gas distorting the intestines) and also because of the position and geometry of the scanning bed.

![Figure 54: MR to CT registration. (a) is the CT-image, (b) is the unregistered MR-image and (c) is the registered MR-image.](image-url)
Figure 55: (a) Registered and (b) unregistered MR-image (magenta) overlaid the CT-image (green).
7 References


URL: http://www.diva-portal.org/smash/record.jsf?searchId=1&pid=diva2:549562


Heismann B J, Schmidt B T, Flohr T and SPIE (Society), 2012. Spectral computed tomography, Bellingham, Wash. , 1000 20th St. Bellingham WA 98225-6705 USA, SPIE.


Jacob, M., 2003. SplineSnake


Marker-Controlled Watershed Segmentation - MATLAB & Simulink Example - MathWorks Nordic [WWW Document], 2014

Mathematical morphology [WWW Document], 2014.


URL: http://www.diva-portal.org/smash/record.jsf?searchId=2&pid=diva2:397399

Otsu thresholding, The Lab Book Pages.


URL: https://www.imt.liu.se/edu/courses/TBMI02/pdfs/registration.pdf (accessed 2014-10-07)


Appendix A

function [bones, adipose, prostate, muscles] = segmentation(image, atlas, histogramReference)
% This is an automated segmentation algorithm.
% Inputs:
%     image: the image that are to be segmented
%     atlas: the atlas used for the segmentation
%     histogramReference: the image used for the histogram matching
% Outputs:
%     bones: the bones of the CT-slice represented by a binary image
%     adipose: the adipose tissue of the CT-slice represented by a binary image
%     prostate: the prostate of the CT-slice represented by a binary image
%     muscles: the muscles of the CT-slice represented by a binary image

% HISTOGRAM-MATCHING-----------------------------------------------
% match image to reference image
[imageHistMatch] = histogramMatching(image, histogramReference);

% REMOVE-CT-TABLE-----------------------------------------------------
% remove CT-table
imageHistMatch = removeTable(imageHistMatch);

% THRESHOLDING-------------------------------------------------------
% thresholding & labeling
[bonesThreshold, adiposeThreshold, bonesThresholdFilled] =
    automatedThresholding(imageHistMatch);

% SEED-GENERATION-----------------------------------------------------
% get bone seeds
[boneSeeds] = getBoneSeeds(bonesThreshold);

% get adipose tissue seed
[adiposeSeed] = getAdiposeSeed(adiposeThreshold);

% REGION-GROWING-----------------------------------------------------
% region growing on adipose tissue
adiposeRegionGrowing = regionGrowing(20, imageHistMatch, ...)
    boneSeeds(1,1), boneSeeds(2));
adiposeRegionGrowingFilled = fillSmallHoles(adiposeRegionGrowing,180);
se = strel('disk',7);
adiposeRegionGrowingFilled = imclose(adiposeRegionGrowingFilled,se);
adiposeRegionGrowingFilled = fillSmallHoles(adiposeRegionGrowingFilled,1000);
adipose = adiposeRegionGrowingFilled;

% region growing on bones
bonesRegionGrowing = zeros(size(imageHistMatch));
for i=1:length(boneSeeds)
    bonesTemp{i} = regionGrowing(35, imageHistMatch, ...)
        boneSeeds(i,1), boneSeeds(i,2)); % Bone right hip joint
    bonesRegionGrowing = bonesRegionGrowing | bonesTemp{i};
end
bonesRegionGrowingFilled = imfill(bonesRegionGrowing, 'holes');

% COMBINE TH & RG-----------------------------------------------------
% bones
bonesAll = bonesRegionGrowingFilled | bonesThresholdFilled;

% CLOSE HOLES----------------------------------------------------------
%closing region growing || thresholding segmentation for bones
bonesSkeleton = bwmorph(bonesAll,'skel',Inf);
se = strel('disk',9);
bonesSkeletonClosed = imclose(bonesSkeleton,se);
bones=bonesSkeletonClosed | bonesAll;
bones=imfill(bones,'holes');

%--------------------------------------------------------------------------
%ATLAS-BASED-IMAGE-REGISTRATION--------------------------------------------
atlas=rgb2gray(atlas);
warning('off', 'MATLAB:unknownElementsNowStruc');
[atlasRegistered,atlasRegisteredAffine]=registration(imageHistMatch, atlas, bones);
warning('on', 'MATLAB:unknownElementsNowStruc');

%--------------------------------------------------------------------------
%DEFORMABLE MODEL PROSTATE-------------------------------------------------
prostate=deformableModelProstate(imageHistMatch, atlasRegistered);

%--------------------------------------------------------------------------
%SEGMENTATION OF MUSCLES---------------------------------------------------
[~, atlasMuscles]=histc(atlasRegisteredAffine,[150 180]);
atlasMuscles=bwareaopen(atlasMuscles, 50, 4);
atlasMusclesLabeled=bwlabel(atlasMuscles,4);
muscleLeft=atlasMusclesLabeled;
muscleLeft(muscleLeft==1)=0;
muscleRight=atlasMusclesLabeled;
muscleRight(muscleRight==2)=0;
muscleRight(muscleRight==2)=1;
muscleLeftFinal=deformableModelMuscle(imageHistMatch.*((uint8(bones)*(-1))+1), muscleLeft);
muscleRightFinal=deformableModelMuscle(imageHistMatch.*uint8((int8(bones)*(-1))+1), muscleRight);
muscles=muscleLeftFinal+muscleRightFinal;
end