

A FULLY AUTOMATED APPROACH TO  
SEGMENTATION AND REGISTRATION OF  
MEDICAL IMAGE DATA FOR PULMONARY  
DIAGNOSIS

By  
Alvin Ihsani

A Thesis Submitted to the  
School of Graduate Studies  
in Fulfillment of the Requirements  
for the Degree  
Master of Applied Science

McMaster University

© Copyright by Alvin Ihsani, August 2009

# COMPUTER AIDED PULMONARY DIAGNOSIS

*Automatic Region of Interest  
Based Registration*

Master of Applied Science (2009)    McMaster University  
(Computing and Software)            Hamilton, Ontario

TITLE:                                    A Fully Automated Approach  
    to Segmentation and Registra-  
    tion of Medical Image Data for  
    Pulmonary Diagnosis

AUTHOR:                                 Alvin Ihsani, B.A.Sc. (Univer-  
    sity of Toronto)

SUPERVISOR:                            Jan Modersitzki

NUMBER OF PAGES:                    vii, 72

# Abstract

Molecular imaging is an exciting and relatively new technology that has found widespread use in the diagnosis and observation of various diseases. More recently, molecular imaging has penetrated areas such as drug development in order to facilitate the observation and analysis of the effects of newly developed drugs. The amounts of data in drug development experiments may be very large due to the fact that they contain both spatial and temporal information of medical images. Imaging techniques can facilitate the analysis of large amounts of data by automating information extraction and providing meaningful results.

The focus of the project concerning this thesis is to create a temporal and spatial atlas of an animal by utilizing and integrating data from images of different modalities. More specifically, the application treated in the thesis makes use of ventilation and perfusion data from CT and SPECT scans in order to aid in the observation of the effects of newly developed drugs in the treatment of lung diseases. This thesis describes the segmentation and registration techniques used in detail and how these were utilized to align and combine ventilation and perfusion data from both CT and SPECT scans.

# Acknowledgements

I wish to first thank my supervisor Dr. Jan Modersitzki for giving me the opportunity to work in the field of image processing. It is thanks to his continuous support, his confidence in me, his exceptional teaching skills and his positive attitude towards exploring new research fields that I was able to make my modest contribution in engineering.

A special thanks goes to Dr. Troy Farncombe and the McMaster Centre for Preclinical Imaging for providing a very challenging project as well as the advice and materials needed. I hope that the methods proposed in this thesis will aid him and his staff in their research.

I also want to thank Dr. Christopher Anand for first introducing me to the field of optimization and later advising me on courses where I could consolidate my knowledge and suggesting areas of research.

I want to thank Dr. Doug Down and Dr. George Karakostas for facilitating my degree transfer. Without their kindness and consideration I would not be in a field of research that I enjoy so much.

Last but not least, I want to thank my family, my fiancée and my peers at McMaster for advising me during times of doubt and never questioning my abilities. I also want to thank my manager Glenn Buchanan for the understanding, support and his ability to make anyone feel comfortable at work and in life.

# Table of Contents

Descriptive Note	iii
Abstract	iv
Acknowledgements	v
Table of Contents	vii
List of Figures	vii
<b>1 Automated Alignment of Pulmonary SPECT Data: An Introduction</b>	<b>1</b>
1.1 Problem Description and Motivation	1
1.2 Materials	3
1.3 Thesis Structure	4
1.4 Introduction to CT and SPECT	5
1.4.1 Introduction to CT	5
1.4.2 Introduction to SPECT	5
1.5 Alignment Process Breakdown	7
1.5.1 Finding an Initial Guess for the Region-of-Interest	7
1.5.2 Refining the Region-of-Interest	7
1.5.3 Registration and Alignment	10
1.6 Alternative Approaches	10
1.6.1 Segmentation	11
1.6.2 Registration	12
<b>2 Background on Existing Methods of Segmentation and Registration</b>	<b>14</b>
2.1 Segmentation Methods	14
2.1.1 The Basic Concept of Segmentation	14
2.1.2 The Mumford-Shah Approach to Segmentation	15
2.1.3 Chan-Vese Approach to Solving the Mumford-Shah Problem	16
2.1.4 Chan-Vese Approach to Segmentation	23
2.2 Registration Methods	28
2.2.1 The Basic Concept of Registration	28
2.2.2 Multi-Level Registration	29
<b>3 Project-Specific Methods for Segmentation and Registration</b>	<b>33</b>
3.1 The Modification of the Chan-Vese Functional for Segmentation	33
3.1.1 Problem Statement and Model	33
3.1.2 Discretization, Implementation and Termination Criteria for Segmentation	35
3.1.3 Alternative Objective Functional Trials for Segmentation	38
3.2 Objective Functions for Ellipsoid Fitting and Registration	39
3.2.1 Ellipsoid Fitting	39
3.2.2 Region-of-Interest Based Registration	41

<b>4</b>	<b>Testing the Segmentation and Alignment Processes</b>	<b>43</b>
4.1	Segmentation Experiments . . . . .	43
4.2	Testing the Alignment Process . . . . .	49
<b>5</b>	<b>Discussion on Current and Future Research</b>	<b>54</b>
5.1	Issues in Segmentation and Registration . . . . .	54
5.2	Conclusions . . . . .	56
<b>A</b>	<b>Matlab Code Extracts</b>	<b>61</b>
A.1	Matlab Code for Ellipsoid Fitting . . . . .	61
A.2	Matlab Code for Lung Segmentation . . . . .	63
A.3	Matlab Code for Image Registration . . . . .	68
A.4	Main Routine . . . . .	70

# List of Figures

1.1	Diffusion of Ventilation and Perfusion Contrast Agents in the Lungs . . .	2
1.2	Breakdown of the Objects in the CT Data . . . . .	3
1.3	Typical Coronal Slices of CT and SPECT Data . . . . .	4
1.4	Diagram of a CT Scanner . . . . .	5
1.5	Diagram of a SPECT Scanner . . . . .	6
1.6	Cross-Section of SPECT Detector Surface . . . . .	6
1.7	Diagram of the perfusion/ventilation alignment process. . . . .	8
1.8	Initial Guess of the Region of Interest . . . . .	9
1.9	Evolution of the Region of Interest . . . . .	10
1.10	Data Registration Based on the Region of Interest . . . . .	11
2.1	Conceptual Diagram of Level-Set Function and Curve . . . . .	17
2.2	Typical Shock Seen in the Level-Set Function . . . . .	22
2.3	Example of Points Lying Outside the Curve $\Gamma$ . . . . .	22
2.4	Typical Cases of Data-Fitting Terms in the Chan-Vese Functional . . . . .	24
2.5	Visualization of Staggered Gradient . . . . .	25
2.6	Visualization of the Averaging Operators on the Staggered Gradient . . . . .	26
2.7	Functionals at Different Levels of Coarseness . . . . .	29
2.8	One-Dimensional Spline Interpolation Example . . . . .	30
2.9	Examples of Image in Multiple Scales and Multiple Levels . . . . .	32
3.1	Typical Initial Guesses and Results of the Chan-Vese Segmentation Method . . . . .	34
3.2	Visualization of the Ellipsoid Function . . . . .	40
4.1	Images Used to Test the Segmentation Process . . . . .	44
4.2	Typical Initial Guesses and Results on the Test Image 4.1(a) . . . . .	45
4.3	Typical Initial Guesses and Results on the Test Image 4.1(a) with Added Noise . . . . .	46
4.4	Typical Initial Guesses and Results on the Test Image 4.1(b) . . . . .	47
4.5	Typical Initial Guesses and Results on the Slices of Laboratory Data . . . . .	48
4.6	Rotation Error Plot . . . . .	51
4.7	Displacement Error Plot . . . . .	52
5.1	Cloud Seen in $P_{SP}$ Data and Its Chan-Vese Segmentation . . . . .	55
5.2	Effect of Foreign Objects in Data Alignment . . . . .	56



## Chapter 1

# Automated Alignment of Pulmonary SPECT Data: An Introduction

Molecular Imaging is a relatively new technology that has found widespread use in the medical and pharmaceutical industries. This type of technology can provide very useful information on a subject and can help in delivering a more precise diagnosis. A precise diagnosis, however, does not always imply that the subject will always be provided the best treatment as there may exist more than one drug or therapy for the same symptom. In order to select the best treatment in terms of medication, where ideally the healing process is “short” and the side effects are minimal, it is indispensable that one should observe the short and long term effects of various drugs or therapies, especially when these are in the development stages. The observation of the effects of a specific treatment involves, however, a periodic “health check” which in some cases may involve scanning the subject over the period of the treatment using different devices such as a CT, SPECT or PET scanner as necessary. Periodic scans of the subject can provide temporal as well as spatial information about the subject thereby creating a better idea of the progress of the treatment.

When dealing with data acquired over a period of time then it is inevitable that corrections should be made to subsequent scans in order to facilitate the observation of differences in the subject. There are a number of reasons why corrections need to be made:

- It is not possible to place the subject in the same exact position in every scan
- There may be different machine artifacts over time
- Other artifacts are introduced by motion such as breathing, heart-beats or involuntary motions
- The subject may exhibit physiological differences over the period of treatment

This thesis, as will be seen in the following sections, deals with certain aspects of motion artifacts created by the placement of the subject in the scanning machine.

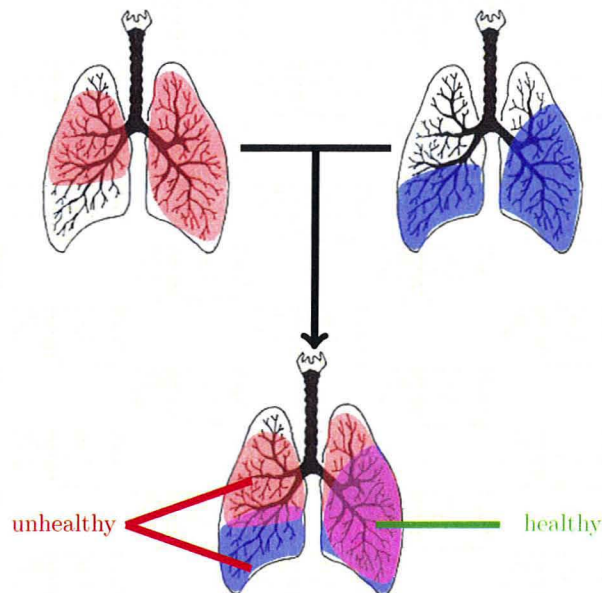
### 1.1 Problem Description and Motivation

The overall objective of the project this thesis is a part of is to enable the observation of the effects of a treatment on a subject over a period of time. As mentioned before, periodic scans of a subject during treatment can help in observing the effects of the

treatment over time. In this specific project the choice was to use a combined CT-SPECT scanner. Computed Tomography (CT) and Single Photon Emission Computed Tomography (SPECT) are described in more detail in later sections, but for now it is not imperative to have thorough knowledge of these.

The project this thesis is a part of deals with the observation of the effects of treatments on the lungs of a subject and specifically abnormal densities of air and blood within the lungs. These can provide information on the symptom being experienced and observe its development during the period of treatment.

In order to track air flow in the lungs a ventilation contrast agent is inhaled and its diffusion is tracked using a SPECT scanner. In a similar way, a perfusion contrast agent is injected in the blood stream then its diffusion through the lungs is observed using the same scanning machine [23, 29]. The air and blood diffusion data, fused together, provides information about healthy and unhealthy volumes in the lungs [23, 29, 33]. Figure 1.1 provides a 2D visualization of the aforementioned.

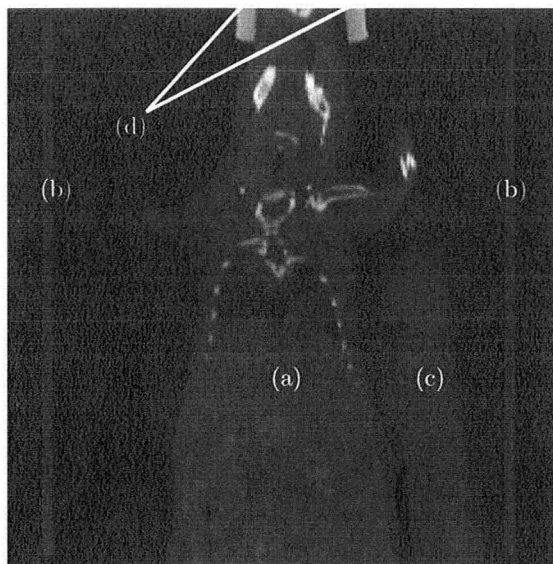


**Figure 1.1:** Visualization of the diffusion of the perfusion contrast agent (top-left), ventilation contrast agent (top-right) and fused data (bottom).

The fusion of the ventilation and perfusion SPECT data and its interpretation is outside the scope of the project treated in this thesis. The problem treated in this thesis deals with aligning the ventilation and perfusion data so that it is ready for fusion and interpretation.

At first, the alignment of the two types of imaging data may seem like a trivial task, however, there are great challenges when trying to automatically align ventilation and perfusion SPECT data. Since SPECT data provides information only on the diffusion of a contrast agent it may not be possible to fuse the data due to large structural differences between data samples. In Figure 1.1 the areas highlighted in red and blue are the only areas that appear in the separate SPECT data and they do not provide any information related to the real shape of the lungs. This means that if these two data samples were to be registered based only on the image intensities (in this case detected counts) the result may be meaningless for diagnosis. In order to register SPECT data, CT data taken at the same time using a combined SPECT-CT scanner can be used. The SPECT-CT scanner pre-aligns the SPECT and CT data since these are acquired only

2 minutes apart. CT data, unlike SPECT, is not based on the diffusion of a contrast agent but rather on the densities of the tissues of the subject and therefore has more structural consistency (see Figures 1.2, 1.3). By registering the Ventilation CT ( $V_{CT}$ ) data to the Perfusion CT data ( $P_{CT}$ ) a transformation can be found which aligns the  $P_{CT}$  to the  $V_{CT}$ . The transformation found is then applied to the Ventilation SPECT ( $V_{SP}$ ) to correctly align it to and Perfusion SPECT ( $P_{SP}$ ).



**Figure 1.2:** Typical coronal slice of CT showing (a) body of subject, (b) couch, (c) water cylinder and (d) muzzle holder.

If the CT data samples corresponding to the ventilation and perfusion SPECT data are registered using the entire data then the results may still be meaningless from a clinical perspective due to the fact that the CT data may also have structural differences such as the different position of the limbs or head in between scans. It is possible, however, to use parts of the CT data in order to achieve the desired alignment. CT data can be registered based only on information including and surrounding the lungs, heart, ribs and chest and back muscles. This area of the body will be referred to here as the Region-of-Interest (ROI). The ROI is important because it removes focus from other areas of the body that may indeed obstruct proper registration between  $V_{CT}$  and  $P_{CT}$ .

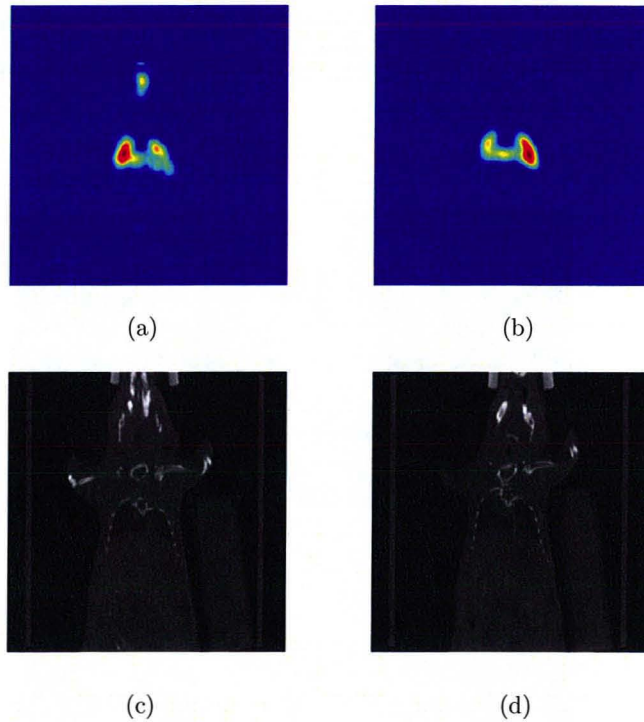
In summary, the process for the alignment of the SPECT data is the following:

- Given pre-aligned  $V_{CT}$ - $V_{SP}$  and  $P_{CT}$ - $P_{SP}$  pairs
- Find the region of interest in the  $P_{CT}$
- Register the region of interest of  $P_{CT}$  to  $V_{CT}$
- Take the resulting transformation and apply it to  $P_{SP}$  to align it to  $V_{SP}$

To the best knowledge of the author, this process is unique, however, there exist alternatives to completing each task in the process. The alternative methods are treated in more detail in a later section.

## 1.2 Materials

The material for this project was provided by Dr. Troy Farncombe and the McMaster



**Figure 1.3:** Coronal slice of (a)  $V_{SP}$ , (b)  $P_{SP}$ , (c)  $V_{CT}$  and (d)  $P_{CT}$ .

Centre for Preclinical Imaging. The samples provided were those of a mouse taken over a period of six weeks with a total of 24 samples. For each week two pairs of pre-aligned data were provided, namely  $V_{CT}$ ,  $V_{SP}$  and  $P_{CT}$ ,  $P_{SP}$  at a resolution of  $256 \times 256 \times 256$  voxels. The total metric space included at this resolution in each sample is  $(58.88 \times 58.88 \times 58.88)mm^3$ .

### 1.3 Thesis Structure

The rest of the thesis is structured as follows.

In Section 1.4, a brief introduction to Computed Tomography (CT) and Single Photon Emission Computed Tomography (SPECT) is presented in order to provide some background on the data used.

Section 1.5 will introduce an alignment process by which the desired alignment between the  $P_{SP}$  and  $V_{SP}$  data is achieved. Each step of the proposed alignment process is then discussed in more detail.

In Chapter 2, the theoretical background on existing methodologies of registration and segmentation is presented. The actual implementation of one of the segmentation methods discussed will also be presented.

Chapter 3 will describe why not all the methods in Chapter 2 could not be used “out of the box” for the project concerning this thesis and how they were modified to fit the project-specific problem.

In Chapter 4, experimental results will be presented in detail for synthetic as well as real-life examples.

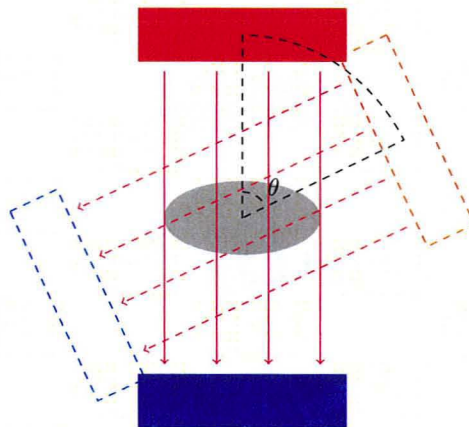
Chapter 5 will discuss the segmentation and registration methods used in terms of efficiency, advantages and disadvantages and suggest some further improvements and draw some conclusions on the work completed in this thesis.

## 1.4 A Brief Introduction to CT and SPECT

In order to have a clearer understanding of some parts of the alignment process presented in Section 1.5, it is beneficial to have some very basic background knowledge of Computed Tomography (CT) and Single Photon Emission Computed Tomography (SPECT). In the following sections a brief introduction to the CT and SPECT scanners is presented along with some information about the data produced.

### 1.4.1 A Brief Introduction to CT

Computed Tomography (CT) is a medical imaging method that uses a series of 2D X-ray projections in order to create 3D data [13, 55]. A CT scanner, as shown in Figure 1.4, has two main components, the detector and the X-ray source situated opposite to the detector. These are connected to a computer which controls the operation, the acquisition and the storage of data.



**Figure 1.4:** Diagram of a CT scanner showing the detector (blue), the X-ray source (red) and the subject (gray) in the center.

The source and detector rotate around a central axis by a step of  $\theta$  degrees every  $\tau$  seconds acquiring an X-ray projection of the subject inside the scanner at each step. The detector consists of a sensing surface composed of Cesium Iodide which scintillates when hit by the radiation thereby creating a flash of light which can be detected by the underlying sensor.

The process by which the 2D projections are combined into 3D data is known as *tomographic reconstruction*. Based on the theory of tomographic reconstruction, initially developed by Johann Radon, if there were an infinite amount of 2D projections the original object can be perfectly reconstructed in 3D [26].

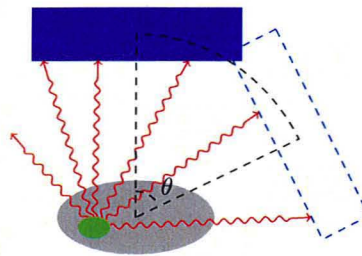
The important point about the data produced by the CT scanner is that it measures tissue density expressed in Hounsfield units (HU). Each tissue in the body exhibits a particular range of intensity values which may or may not overlap with other tissues. *Windowing* is one quick method to narrow focus to a specific range of intensity values. For instance, by windowing values ranging between -500HU and -100HU one is able to narrow focus on the lungs, however, this runs the risk of showing other tissues which lie in the same intensity range.

### 1.4.2 A Brief Introduction to SPECT

In some cases CT data does not provide enough information about the condition of the subject. For instance, CT data does not emphasize on patterns of air flow and blood

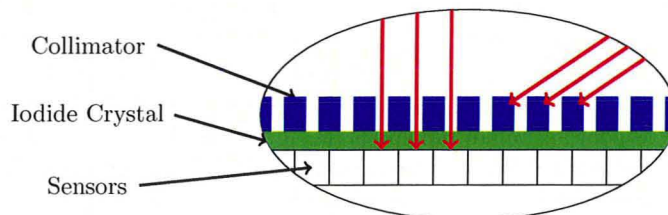
flow in the lungs which can be indicative of diseases. For such cases, different imaging techniques need to be employed.

Single Photon Emission Computed Tomography (SPECT) is another imaging technique used in nuclear medicine. The idea of SPECT is the same as that of CT where a detector is rotated on a central axis around the body of a subject, but instead of measuring the attenuation of rays coming from an X-ray source, it detects radiation levels from a radio-labeled contrast agent injected in the body of the subject by counting the number of photons hitting the sensor. Different from CT, SPECT does not measure tissue density but rather the diffusion of a contrast agent through the body and therefore SPECT data has more structural variation than CT data. The choice of the contrast agent depends on the organ of focus. For instance, for a lung perfusion scan of a mouse, the contrast agent used is Tc-99m labeled macro-aggregate albumin. This contrast agent “gets stuck” in small arterioles in the lungs thereby depicting blood flow. On the other hand, for a ventilation scan the contrast agent used is Tc-99m Technegas. When the compound is heated, the fumes of this contrast agent are inhaled by the subject and the smoke sticks to the lungs thereby showing ventilation.



**Figure 1.5:** Diagram of a SPECT scanner showing the contrast agent (green) in the subject (gray) in the center.

The detector of the SPECT scanner is similar to that of the CT scanner. However, the detector is composed of two main components, the collimator and the sensing surface. The collimator is a layer which is situated on top of the sensing surface. The purpose of the collimator is to block the sensing surface from detecting rays that are not perpendicular to the detector. Without the collimator the detector would detect rays that may not come from the source and through the body of the subject thereby providing inaccurate projections. Figure 1.6 shows the layers of the detector. The data is reconstructed by combining the 2D projections acquired using tomographic reconstruction and the intensity values are represented as reconstructed counts per unit time.



**Figure 1.6:** Cross-section of SPECT detector surface. Rays approximately perpendicular to the detector are sensed while others are blocked by the collimator.

## 1.5 Alignment Process Breakdown

The objective of the project concerning this thesis is to find a way of aligning the  $P_{SP}$  and  $V_{SP}$  data by using the corresponding  $P_{CT}$  and  $V_{CT}$  data since CT data has more structural consistency than SPECT data. However, objects and body parts in the CT data such as the couch, the water cylinder (used to calibrate Hounsfield Units for CT), the muzzle holder, the position of the arms, legs and lower body (see Figure 1.2) may obstruct the proper registration of the chest between the  $P_{CT}$  and  $V_{CT}$  data. In order to register the CT data correctly the focus must be narrowed down to the chest in the  $P_{CT}$  data which in the context of this thesis is known as the *region of interest*. Therefore, proper registration of the ventilation and perfusion CT is achieved by aligning the region of interest of the  $P_{CT}$  data with the  $V_{CT}$  data.

An alignment process that aligns the  $P_{CT}$  and  $V_{CT}$  data as required is described by following three steps:

1. Find an initial guess for the ROI on the  $P_{CT}$  data based on the  $P_{SP}$  data
2. Refine the initial guess based on the  $P_{CT}$  data alone
3. Register the ROI of  $P_{CT}$  data to the  $V_{CT}$  data

Figure 1.7 provides a conceptual map of the overall process. In the following sections each step is described in detail.

### 1.5.1 Finding an Initial Guess for the Region-of-Interest

At this stage an initial guess for the region of interest is needed. In order to obtain a “good” initial guess the  $P_{SP}$  data can be used. The  $P_{SP}$  data typically shows the highest concentration of the contrast agent inside the lungs. There may be cases where the contrast agent used for lung perfusion SPECT scans spreads to the bladder, however, this is not very common and can be easily masked off. Given the above information, the assumption is that the largest concentration of the contrast agent in perfusion SPECT is within the lungs (Figure 1.3-b). Based on this assumption unless there is a prior misalignment between the  $P_{SP}$ - $P_{CT}$  data pair, the initial ROI found from the  $P_{SP}$  data will lead to a volume that is within or very close to the lungs in the  $P_{CT}$  data.

For the first step of the alignment process, the approach is to fit an ellipsoid to a windowed version of the  $P_{SP}$  data. The windowing is chosen to remove small non-zero values outside of the lungs. The ellipsoid-fitting is done in such a way that the smallest ellipsoid enclosing the windowed  $P_{SP}$  data is found. This ellipsoid is indicative of the approximate region where the lungs are situated in the  $P_{CT}$  data. Figure 1.8 shows a typical result of the initial ROI.

The initial guess is described by an ellipsoid because the shape of the ROI is not important at this stage and an ellipsoid is easy to handle.

In the next step, presented in the following section, the ROI must be refined as it may still include regions in the  $P_{CT}$  data that may obstruct correct alignment.

### 1.5.2 Refining the Region-of-Interest

The initial guess obtained in the step described in the previous section may still contain organs or objects which might obstruct proper alignment. The reason for the initial guess containing these “foreign” objects is because it may be larger than the lungs. For instance, in subjects with no lung perfusion problems, the contrast agent spreads throughout the entire volume of the lungs and therefore the ellipse enclosing the windowed data in the  $P_{SP}$  might be larger than the lungs. This means that the initial region of interest in the  $P_{CT}$  data might contain part of the arms or other objects that are in close proximity to the chest. On the other hand, if the initial region of interest is too small then there may not be enough information in the region of interest to register

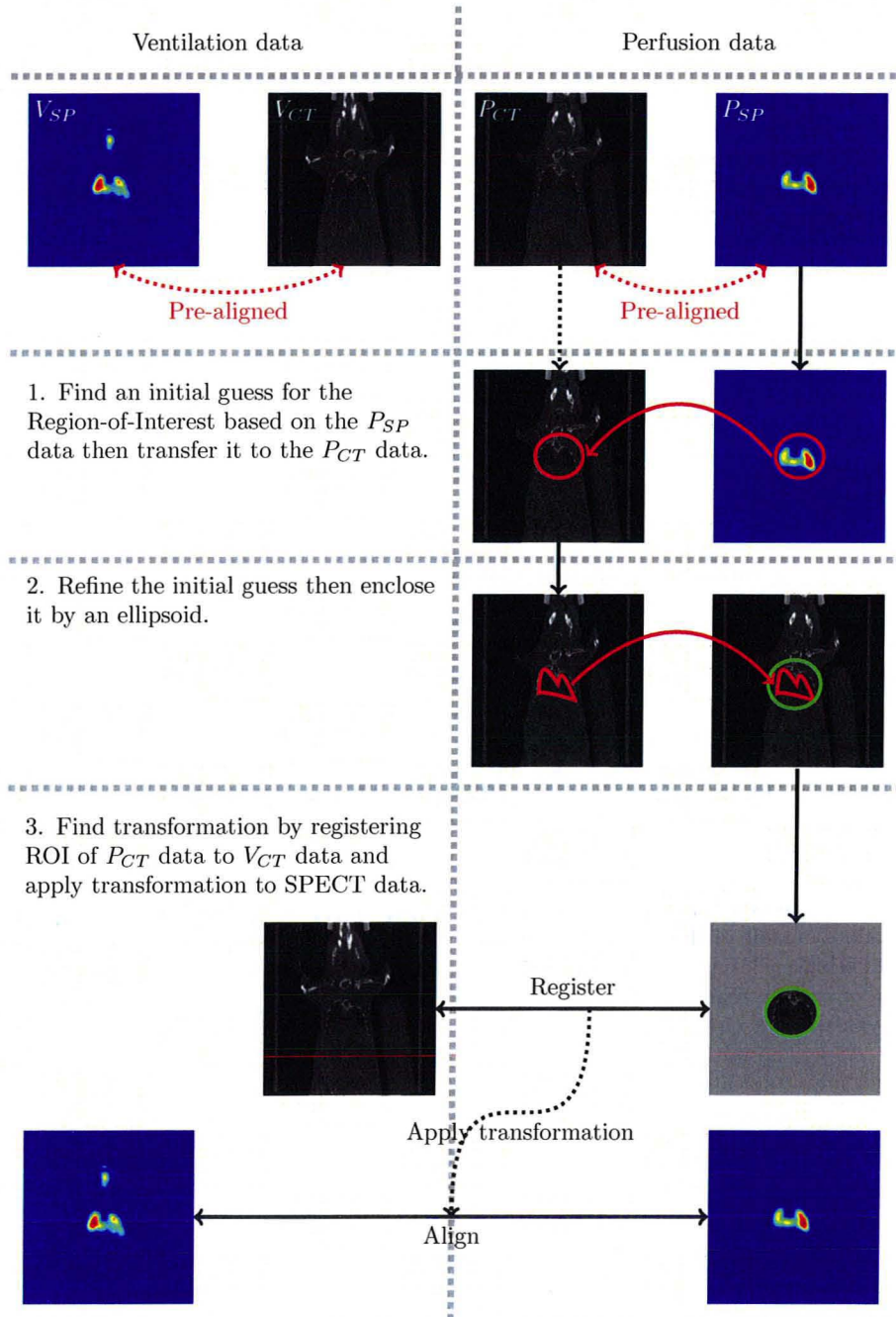
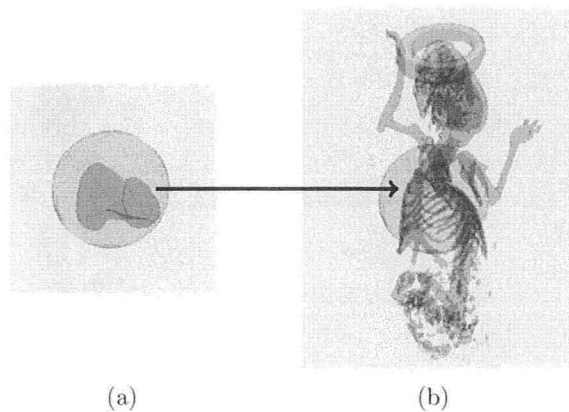


Figure 1.7: Diagram of the perfusion/ventilation alignment process.





**Figure 1.8:** A typical result of (a) the initial ROI on the  $P_{SP}$  data and (b) on the  $P_{CT}$  data.

the data reliably. An assumption that can be made about the initial region of interest is that the ellipsoid is very close to or within the lungs.

In order to remove any existing “foreign” objects from inside the region of interest the ellipsoid is evolved to the shape of the lungs using a segmentation method. The segmentation method used is known as active contours using a level-set formulation [31, 37, 38, 47–50]. The thorough mathematical details will be discussed in later chapters, but the underlying concept is presented here.

Given 3D data, a so-called level-set function  $\phi : \mathbb{R}^4 \rightarrow \mathbb{R}$  is used to describe the contour  $\Gamma$ , which in turn is used to describe the shape of the lungs, as its intersection with the 3D plane  $\Gamma = \{\phi = 0\}$ . Initially,  $\Gamma$  is the contour of the ellipsoid found in the initial guess presented in the previous section. The function  $\phi$  must then evolve according to some predefined criteria in order for  $\Gamma$  to enclose some volumes of interest in the body of the subject. There are many approaches to define how the function  $\phi$  evolves. Some of these approaches include:

- Directly defining partial differential equations, such as that of curvature flow using the heat equation [37]
- Using an optimization setting [2, 18, 31, 37, 38, 44, 47–50, 53, 54]

In the project treated in this thesis an optimization approach was chosen.

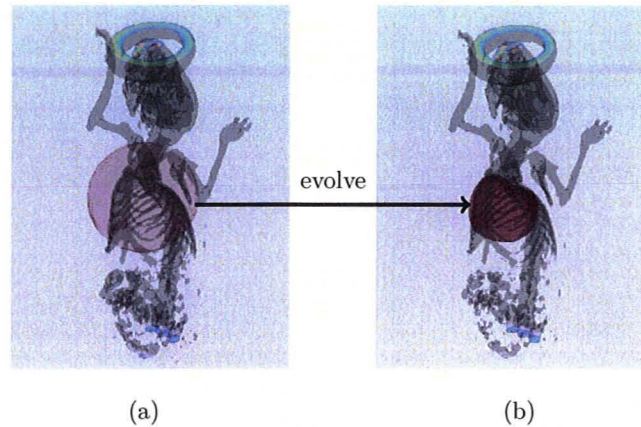
The purpose of this step is to evolve an initial guess in such a way that the volume it encloses approximately matches the volume of the lungs, see Figure 1.9.

Once this step is complete another ellipsoid is fit on the segmented volume. The reason for this is that, even if the lungs are perfectly segmented, still more information about the surrounding tissues, such as the ribs, is needed because they have a rigid structure. While other organs such as the heart and the lungs may move within the chest during a scan due to heartbeat and breathing, the ribs are more rigid and help when finding a rigid transformation to align the  $P_{CT}$  and  $V_{CT}$  data.

The purpose of this step is to find an ellipsoidal region that is more accurate than the initial. The desired result is to find a refined region of interest that is close to the surface of the lungs. For instance,

- If the initial ellipsoid is too large then after the segmentation step, the ellipsoid should be smaller according to the size of the lungs in the  $P_{CT}$  data.
- Analogically, if the initial guess is too small then after this step the ellipsoid should have grown.

In the next section, the incorporation of the result from this step as well as the registration process are described.



**Figure 1.9:** The evolution from (a) the initial guess in the  $P_{CT}$  data into (b) the segmented volume in the  $P_{CT}$  data.

### 1.5.3 Registration and Alignment

As a final step, the refined ellipsoidal ROI is used as a weight on the  $P_{CT}$  data. In other words, the  $P_{CT}$  data is spatially filtered in such a way that the only regions containing information about the subject are enclosed in the refined ROI.

Finally, the ROI of the  $P_{CT}$  data is registered to the  $V_{CT}$  data. The registration in this specific project is multilevel [11, 22]. Multi-level registration involves registering the template ( $P_{CT}$ -ROI) and reference ( $V_{CT}$ ) data at multiple resolutions starting with the lowest desired resolution and incrementing it to the highest desired level. At lower resolutions, the data retains such properties as shape, orientation and relative position but loses detail. This means that at lower resolutions the registration is faster and less prone to finding local minima. As the resolution is increased the minimizers found at lower resolutions are used as the initial guess and refined.

For the project concerning this thesis, the registration currently deals only with rigid transformations. Other transformations, such as affine and elastic, will be implemented in the future. Furthermore, it is assumed that the relative position and shape of the organs inside the rib-cage remains unchanged due to the fact that the rib-cage has a rigid structure. The registration finds the transformation that minimizes the  $L_2$ -norm of the distance between the ROI of the  $P_{CT}$  and  $V_{CT}$  data.

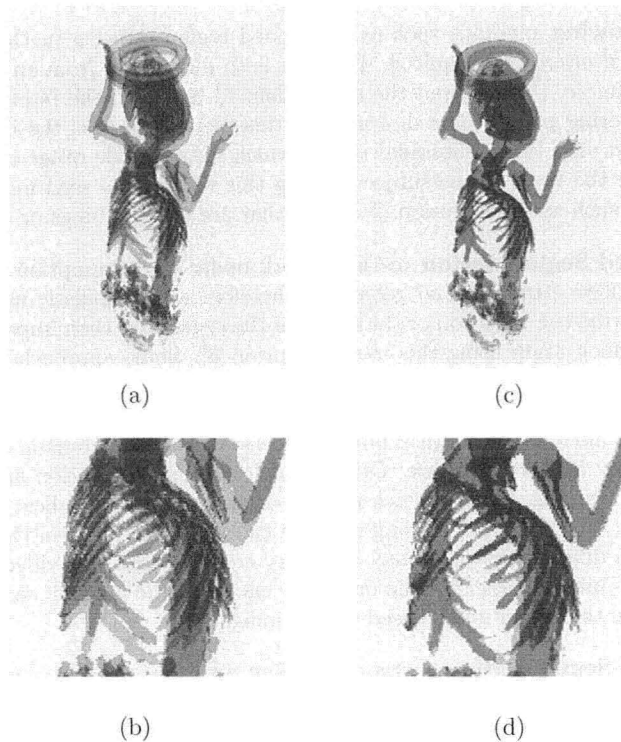
Figure 1.10 shows a typical pair of  $P_{CT}$ - $V_{CT}$  data before and after registration. Since the registration is based only on data inside the region of interest of the  $P_{CT}$ , it is expected that the two data are aligned correctly in the chest area.

As a final sub-step, the resulting transformation of the  $P_{CT}$  data to  $V_{CT}$  data is applied to the  $P_{SP}$  data in order to align this with the corresponding  $V_{SP}$  data.

## 1.6 Alternative Approaches

To the best knowledge of the author, there are no other fully automated processes that try to achieve alignment between different measurements of SPECT data in the same way. Each individual step of the aforementioned process, however, may be potentially substituted by alternative segmentation and registration methodologies.

In the following sections, some alternative approaches to segmentation and registration are presented and compared to the current approach.



**Figure 1.10:** Overlapped  $P_{CT}$  (red) and  $V_{CT}$  (gray) shown (a) before registration with (b) closeup of the ROI and (c) after the registration with (d) closeup of the ROI.

### 1.6.1 Segmentation

Segmentation methods aim to partition an image into separate regions according to some criteria. For instance, when building an atlas of the human body the aim may be to partition the image into separate regions showing different organs or tissues. In other situations, it may be necessary to focus on a specific organ or object therefore a single region containing the object of interest is needed.

There are many methods that can be used in image processing for segmentation [7] some of which are briefly described in the following paragraphs and are compared in terms of the objective of this part of the project.

**Histogram** based methods use a histogram of the image in order to find clusters in an image. The intensity of values or color can be used as the measure. Simple histogram methods may not be able to isolate an organ of interest such as the lungs due to the fact that the lungs and other parts of the body may have similar intensity values. However, more advanced methods using also active contours methods such as that proposed in [51] may provide a closer result to the one expected for the project of this thesis. Methods such as that proposed in [51] may be considered for future research in the segmentation of the lungs.

**Edge Detection** methods accentuate the edges in an image. The edges in the image are typically detected by the sharp transitions in color or intensity between different objects [19]. The edges detected can then be used in another segmentation method to find the boundaries of an object of interest. However, an object of interest can also be isolated without using edges therefore edge detection methods

were not used in the project of this thesis.

**Region Growing** methods such as the seeded region growing method can isolate an organ of interest as required. The idea is to pick pixels from an organ of interest, for example, then expand the region defined based on the intensity values of the neighboring pixels. The downside of this method is that the input provided by the first step is an ellipsoidal region which may include other organs or air from outside the body of the subject. Using this region as a seed may have undesired effects such as the inclusion of organs that are not the lungs or other objects.

**Model-based Segmentation** methods work under the assumption that organs have a repetitive structure and geometry, therefore a probabilistic model can be used to describe the variation of the shape of the organ and then impose constraints on the segmentation using this model as prior [8]. Some approaches in model-based segmentation require training samples. In the application of this thesis training is being avoided due to the fact that it is time consuming, in an off-line sense, costly and the models of statistical inference between the probabilistic model and actual data may be very complex. Other model-based segmentation approaches do not require training samples which is more desirable for the application in this thesis, however, automatically finding a statistical or deterministic model to describe the lungs in different subjects may be a very complex task. Provided the assumption that the initial guess is within or closely enclosing the lungs, it seems more natural to refine the initial guess based on the image.

**Multi-Scale Segmentation** methods segment an image in scale-space. This means that during the process of segmentation the scale of the image varies incrementally from coarse details to finer details. From a recent number of tests of the currently implemented segmentation method in multi-scale, it seems that the results are promising and this is definitely a path that will be followed in the very near future.

## 1.6.2 Registration

Image registration methods aim to find a reasonable transformation between two images in order to make the images more similar to one another. A reasonable transformation is problem specific and may include rigid, affine, elastic or other transformations. Likewise, the similarity measurement is also problem specific. For instance, the project in this thesis uses the energy of the difference image as a measurement of similarity between the  $V_{CT}$  and  $P_{CT}$  data. An alternative way to measure similarity is by using feature-based methods. Typically, feature-based methods either require user input to draw corresponding features (not desirable for the application treated in this thesis) or find corresponding features automatically. Since the  $V_{CT}$  and  $P_{CT}$  data are of the same modality, the intensity-based registration was chosen over the feature-based one.

The current approach of registration deals only with rigid transformations since it works under the assumption that the rib-cage has a fairly rigid structure. However, elastic and affine transformations are not being ruled out since there may be situations for which a rigid transformation may not be sufficient to properly align the data. So far, the restriction to rigid transformations has worked for the data used for the project treated in this thesis.

Multi-level and multi-scale registration are two registration methods that are effective in avoiding local minima. Multi-level registration consists of using multiple resolutions to align images, while multi-scale registration consists of using a single resolution at multiple scales. These methods will be explained in more detail in a later chapter.

In the application treated here, multi-level registration was the registration method of choice since it avoids local minima and improves computation time. Local

minima are avoided by using coarser resolutions which retain only the main features such as shape, location and orientation. The minimizers found at each stage are used as the starting guess at a finer level and refined until the maximum desired resolution is reached. Computation time is improved due to the usage of lower resolutions.

## Chapter 2

# Background on Existing Methods of Segmentation and Registration

This chapter will focus on existing methods of registration and segmentation. These methods have theoretical and practical importance since this project is heavily based on them. The notation and some of the functions presented here are inherited by later chapters where these methods have been extended in order to fit the specific needs of this project.

Due to the fact that segmentation has evolved significantly in the last twenty years the description of it will be laid out chronologically. First, the early work of Mumford and Shah [10] will be described due to its importance in posing the problem of segmentation and de-noising. Later, a regularization to the Mumford-Shah problem proposed by Osher and Sethian [47], known as the level-set formulation, will be introduced along with state of the art segmentation methods that use this formulation to solve special cases of the Mumford-Shah problem such as those proposed by Chan and Vese [49, 50]. The Chan and Vese methods proposed in [49, 50] will be discussed in finer detail as these are the basis of the segmentation method developed in the project of this thesis.

Section 2.2 will provide some background on existing registration methods. This section begins with an overview of image registration [3, 21, 35] using an optimization approach followed by a description of single, multi-scale and multi-level registration in order to provide an idea of the tools used in this thesis.

## 2.1 Segmentation Methods

### 2.1.1 The Basic Concept of Segmentation

The basic idea of segmentation is to be able to partition the domain  $\Omega \subset \mathbb{R}^d$  of an image  $u : \Omega \rightarrow \mathbb{R}$  into a finite number of open subsets  $\Sigma_i$ , for  $i \in \{1, 2, \dots, n\}$ , where the following conditions based on [10] hold

$$\begin{cases} \bar{\Omega} = \bigcup \bar{\Sigma}_i \\ \Sigma_i \cap \Sigma_j = \emptyset, \quad \forall i \neq j \\ \Sigma_i \neq \emptyset \\ \Sigma_i \subset \Omega \end{cases} \quad (2.1)$$

The partitioning of the domain into subsets is based on some predefined criteria which are problem dependent. For instance, in some cases it may be necessary to partition the domain of an image into separate objects with sharp boundaries between

them. In other cases, it may be necessary to extract a single object from an image depending on its shape, location, size or measured value.

A boundary  $\Gamma$  divides the image into open subsets  $\Sigma_i$  and is defined as

$$\Gamma = \bigcap \bar{\Sigma}_i \cap \Omega. \quad (2.2)$$

In other words,  $\Gamma$  is the union of the boundaries of subsets  $\Sigma_i$  inside  $\Omega$ .

With this definition of  $\Gamma$  it is possible to enforce constraints on it based on internal factors, such as length or curvature, and external factors, such as the underlying data [31, 39, 50, 53, 54]. One way to impose a certain shape on  $\Gamma$  is to use an optimization approach where an objective functional can be constructed that regularizes and penalizes its behaviour. An objective functional can have internal energies  $E_{in}$  and external energies  $E_{out}$  as shown below.

$$E(\Gamma) = E_{in}(\Gamma) + E_{out}(\Gamma). \quad (2.3)$$

In order to find the curve  $\Gamma^*$  that minimizes (2.3) the steepest descent method can be used where the evolution of  $\Gamma$  is defined by  $-\nabla E$  at each step.

In the following sections only most influential methodologies pertinent to the project in this thesis will be discussed.

### 2.1.2 The Mumford-Shah Approach to Segmentation

In this section some of the work of Mumford and Shah in [10] will be discussed. The functionals described in this section are of importance due to the fact that they are introductory to the work described in [47, 49, 50] which are treated in the following sections.

In [10], an image is considered to be a projection of a 3D world into an observer point  $P$  where  $g(\rho)$  represents the intensity of light reaching  $P$  in the direction  $\rho$ . When these rays of light are projected onto a plane  $\Omega$ , then the intensity of light incidental to the plane can be represented by  $g(x, y)$ . The light reflected from the surfaces  $S_i$  of objects  $O_i$  is projected on the plane  $\Omega$  and divides the plane  $\Omega$  into open subsets  $\Sigma_i$ . Objects that overlap will create subsets  $\Sigma_i$  that share a common boundary  $\Gamma$ , also known as an edge, where the image is considered to be discontinuous. Other discontinuities may be introduced by surface orientation, such as the surfaces of a cube, markings on the surface of the object and shadows. The assumption is that the image  $g(x, y)$  is piece-wise smooth and it can be modelled by a set of smooth functions  $f_i$  defined on disjoint regions  $\Sigma_i$  covering  $\Omega$ . However, this model is not entirely accurate due to the following:

- Textured objects, such as carpets, or fragmented objects, such as a canopy of leaves, define more complicated images
- Shadows create discontinuities which are not real, due to the penumbra
- Surface markings may come in misleading forms
- Partially transparent objects and reflecting objects also define more complicated images
- The measurement of  $g$  may contain noise

In summary, the domain of the image  $g$  is to be decomposed in such a way that

- the image  $g$  varies smoothly within each  $\Sigma_i$ , and
- is discontinuous over the boundary  $\Gamma$  between different  $\Sigma_i$ .

From an approximation theory point of view, the problem seeks ways to define and compute *optimal approximations* of a general function  $g(x, y)$  by a piece-wise smooth function  $f(x, y)$ , where each  $f_i$  defined on  $\Sigma_i$  is differentiable. Given this definition then the optimal approximation is defined as the minimizer of the following functional

$$E(f, \Gamma) = \mu^2 \int_{\Omega} (f - g)^2 dx dy + \int_{\Omega \setminus \Gamma} \|\nabla f\|^2 dx dy + \nu |\Gamma| \quad (2.4)$$

Here,  $f$  is a differentiable function on  $\bigcup \Sigma_i$ , which is allowed to be discontinuous over  $\Gamma$ , and  $|\Gamma|$  represents the total length of the arcs making up  $\Gamma$ . In this functional, the first term asks that  $f$  approximate  $g$ , the second term asks that  $f$  vary as little as possible in each  $\Sigma_i$  and the third term asks that the boundaries between different regions that make up  $\Gamma$  be as small as possible.

Dropping any of the three terms in (2.4) would lead to the following results. Without the first term then  $f = 0$  and  $\Gamma = \emptyset$ ; without the second term then  $f = g$  and  $\Gamma = \emptyset$ ; without the third term  $\Gamma$  can be a fine grid of  $N$  horizontal and vertical lines, composing  $N^2$  square regions, where  $f$  is the average value of  $g$  in each  $\Sigma_i$ . If  $\Gamma$  covers the entire set  $\Omega$ ,  $\Gamma = \Omega$ , the second term drops and  $f = g$ .

Loosely speaking, a minimizer  $(f^*, \Gamma^*)$  of the functional in (2.4) is a cartoon of the actual image  $g$ . In  $f^*$  edges are drawn sharply and precisely and the objects surrounded by the edges are smooth and without texture. It is obvious that  $f^*$  no longer retains the amount of detail of  $g$ , but still contains many of its essential features.

It was not known whether the problem of minimizing (2.4) was well-posed, but Mumford and Shah conjectured this to be true. The problem was proven to be well-posed only later by Maso, Morel and Solimini in [15]. Mumford and Shah conjectured that for all continuous functions  $g$ ,  $E$  has a minimum in the set of all pairs  $(f, \Gamma)$ , with  $f$  differentiable on each  $\Sigma_i$  and  $\Gamma$  a finite set of singular points joined by a finite set of  $C^1$ -arcs.

Functional (2.4) had been previously introduced in [45] for functions  $g$  and  $f$  on a lattice and studied in [1, 25], however, Mumford and Shah introduced this for a plane domain  $\Omega$ .

A restriction on the functional in (2.4) is to use piece-wise constant functions  $f$  such that  $f = a_i$  on each open set  $\Sigma_i$ . This means that the second term in (2.4) drops out since the gradient is zero. Dividing both sides by  $\mu$  the following functional is obtained

$$E_0(f, \Gamma) = \sum_i \int_{\Sigma_i} (g - a_i)^2 dx dy + \nu_0 |\Gamma|. \quad (2.5)$$

This functional is minimized in the variables  $a_i$  by setting

$$a_i = \text{mean}_{\Sigma_i}(g) = \int_{\Sigma_i} g dx dy / \text{area}(\Sigma_i). \quad (2.6)$$

Therefore, the functional to be minimized is in fact the following

$$E_0(\Gamma) = \sum_i \int_{\Omega_i} (g - \text{mean}_{\Sigma_i} g)^2 dx dy + \nu_0 |\Gamma|. \quad (2.7)$$

As shown in [10] by Mumford and Shah, if  $\Gamma$  is fixed then the  $f$  which minimizes  $E$  in (2.4) tends to a piece-wise constant limit when  $\mu \rightarrow 0$  and therefore  $E_0$  in (2.7) is the natural limit functional of  $E$ . In addition, the problem of minimizing  $E_0$  is proven to be well-posed: for any continuous function  $g$ , there exists a  $\Gamma$  made up of a finite number of singular points joined by a finite set of  $C^2$ -arcs in which  $E_0$  attains a minimum [10].

### 2.1.3 Chan-Vese Approach to Solving the Mumford-Shah Problem

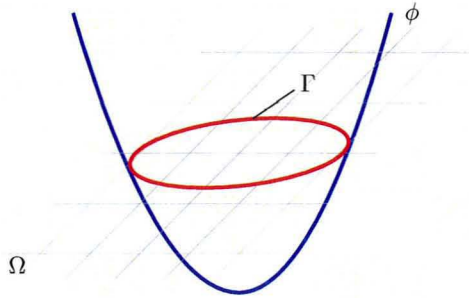
In the last section a brief description of the Mumford-Shah problem was laid out along with the functionals proposed by them in order to address the problem of segmentation.



Note that Mumford and Shah did not impose special conditions on the nature of  $\Gamma$  itself.  $\Gamma$  introduces discontinuities where it passes and singular points wherever it “crosses” with itself. The generalization of  $\Gamma$  complicates the problem.

A regularization on  $\Gamma$ , known as the level-set formulation was first introduced by Osher and Sethian in [47] and has been used quite extensively since [50]. This formulation simplifies a number of the special cases considered by Mumford and Shah [10]. In [47] the curve  $\Gamma \subset \Omega \subset \mathbb{R}^d$  is defined as the zero level-set of a function  $\phi : \mathbb{R}^{d+1} \rightarrow \mathbb{R}$  (see Figure 2.1), therefore

$$\Gamma = \{\mathbf{x} | \phi(\mathbf{x}) = 0\}. \quad (2.8)$$



**Figure 2.1:** Conceptual diagram showing the zero level-set  $\Gamma$  of the function  $\phi$  in 3D.

This formulation allows for cusps and corners as dictated by automatic topological changes.

Assuming a single level-set functional and  $\Gamma$  lying entirely within the domain  $\Omega$ , then a natural question arises with regards to indicating what lies “inside” and “outside” the curve  $\Gamma$ . It is then useful to define an indicator function that facilitates this.

One indicator function that can facilitate the identification of different regions is the Heaviside function defined as

$$H(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}. \quad (2.9)$$

The actual Heaviside function used in this thesis is based on [50, 56].

$$H(z) = \frac{1}{2} \left( 1 + \frac{2}{\pi} \arctan \left( \frac{z}{\epsilon} \right) \right) \quad (2.10)$$

where  $\epsilon$  is a small positive number. As will be seen in the following material, it is necessary to have a differentiable Heaviside function that is why (2.10) is used.

For simplicity, from this point on, all integrals over the domain  $\Omega$  will be assumed to be over  $x$  and  $y$ , therefore the following notation will be used unless otherwise stated:

$$\int_{\Omega} q = \int_{\Omega} q(x, y) dx dy$$

A simplified approach to the Mumford-Shah functional in (2.4) was proposed by Chan and Vese in [49]. The idea is to partition the image into two regions separated by a boundary  $\Gamma$ . The regions  $f_1, f_2$  are defined as

$$g = \begin{cases} f_1 & \text{if } \phi > 0 \\ f_2 & \text{if } \phi < 0 \end{cases} \quad (2.11)$$

where  $f_1$  and  $f_2$  are  $C^1$  functions on  $\{\phi \geq 0\}$  and  $\{\phi \leq 0\}$  respectively. The image  $g$  can be constructed by combining  $f_1$  and  $f_2$ . This is known as the *two phase model* [49].

Using the above setting a simplified version of (2.4) can be written as follows

$$\begin{aligned}
 E(f_1, f_2, \phi) &= \mu \int_{\Omega} |\nabla f_1|^2 H(\phi) + \int_{\Omega} |f_1 - g|^2 H(\phi) \\
 &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H(\phi)) + \int_{\Omega} |f_2 - g|^2 (1 - H(\phi)) \\
 &+ \nu \int_{\Omega} H'(\phi) |\nabla \phi|
 \end{aligned} \tag{2.12}$$

In this functional the first two terms ask that  $f_1$  approximate  $g$  inside the curve  $\Gamma$  while keeping as “flat” as possible; the second two terms ask the same of  $f_2$ , but for the region outside  $\Gamma$ ; the last term asks that  $\Gamma$  should have the smallest possible length. These conditions closely resemble the conditions stated by Mumford and Shah for (2.4) with an additional limitation imposed on the number of regions that compose the image.

The last term in equation (2.12) was obtained as follows. The length of the curve  $\Gamma$ , as defined in [27], is

$$|\Gamma| = \int_{\Omega} \|\nabla H(\phi)\|.$$

With some simple manipulations and considering that  $0 \leq H(z)$  in (2.10) then

$$\|\nabla H(\phi)\| = \|H'(\phi)\nabla\phi\| = H'(\phi)\|\nabla\phi\|.$$

Therefore,

$$|\Gamma| = \int_{\Omega} \|\nabla H(\phi)\| = \int_{\Omega} H'(\phi)\|\nabla\phi\|.$$

In order to minimize (2.12) the steepest descent method was used. In the following sections the computation of  $-\nabla E$  and the necessary conditions are described step by step. First, the first variation with respect to  $\phi$  is computed. This is followed by a first variation on  $f_1$  and  $f_2$ . Finally, a summary of the computed gradient and the necessary conditions is provided.

### Computing the First Variation of $E$ With Respect to $\phi$

In order to minimize functional (2.12) a variational approach is taken. Let  $f_1$  and  $f_2$  be fixed, then

$$\begin{aligned}
 \gamma(t) &= E(f_1, f_2, \phi + t\psi) = \mu \int_{\Omega} |\nabla f_1|^2 H(\phi + t\psi) + \int_{\Omega} |f_1 - g|^2 H(\phi + t\psi) \\
 &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H(\phi + t\psi)) + \int_{\Omega} |f_2 - g|^2 (1 - H(\phi + t\psi)) \\
 &+ \nu \int_{\Omega} H'(\phi + t\psi) |\nabla\phi + t\nabla\psi|
 \end{aligned}$$

and

$$\begin{aligned}
 \gamma'(t) &= \mu \int_{\Omega} |\nabla f_1|^2 H'(\phi + t\psi)\psi + \int_{\Omega} |f_1 - g|^2 H'(\phi + t\psi)\psi \\
 &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H'(\phi + t\psi))\psi + \int_{\Omega} |f_2 - g|^2 (1 - H'(\phi + t\psi))\psi \\
 &+ \nu \int_{\Omega} H''(\phi + t\psi) \|\nabla\phi + t\nabla\psi\| + \nu \int_{\Omega} H'(\phi) \frac{\nabla\phi + t\nabla\psi}{\|\nabla\phi + t\nabla\psi\|} \nabla\psi.
 \end{aligned}$$

In order to find the descent at  $\phi$  then set  $t = 0$

$$\begin{aligned}
 \gamma'(t) &= \mu \int_{\Omega} |\nabla f_1|^2 H'(\phi)\psi + \int_{\Omega} |f_1 - g|^2 H'(\phi)\psi \\
 &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H'(\phi))\psi + \int_{\Omega} |f_2 - g|^2 (1 - H'(\phi))\psi \\
 &+ \nu \int_{\Omega} H''(\phi) \|\nabla\phi\| + \nu \int_{\Omega} H'(\phi) \frac{\nabla\phi}{\|\nabla\phi\|} \nabla\psi
 \end{aligned}$$

The last term in this equation does not describe the variation in terms of the perturbation  $\psi$  directly, but rather in terms of the change in perturbation  $\nabla\psi$ . Using integration by parts [17] the following form can be reached.

$$\begin{aligned}\gamma'(0) &= \mu \int_{\Omega} |\nabla f_1|^2 H'(\phi)\psi + \int_{\Omega} |f_1 - g|^2 H'(\phi)\psi \\ &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H'(\phi)\psi) + \int_{\Omega} |f_2 - g|^2 (1 - H'(\phi)\psi) \\ &+ \int_{\partial\Omega} H'(\phi)\psi \left\langle \frac{\nabla\phi}{\|\nabla\phi\|}, \vec{n} \right\rangle - \int_{\Omega} H'(\phi)\nabla \cdot \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) \psi\end{aligned}\quad (2.13)$$

where  $\vec{n}$  is the normal to the boundary  $\partial\Omega$ . In order to find the solution  $\phi$  then  $\gamma'(0) = 0$ .

Chan and Vese have assumed in their formulation that  $\phi$  does not vary with respect to the normal of the boundary of the domain and therefore the boundary term

$$\int_{\partial\Omega} H'(\phi)\psi \left\langle \frac{\nabla\phi}{\|\nabla\phi\|}, \vec{n} \right\rangle$$

in equation (2.13) vanishes since

$$H'(\phi) \left\langle \frac{\nabla\phi}{\|\nabla\phi\|}, \vec{n} \right\rangle = 0 \quad (2.14)$$

This is known as the *natural boundary condition* [4, 28]. The natural boundary condition on  $\partial\Omega$  means that the curve  $\Gamma$  will intersect with the boundary of  $\Omega$  perpendicularly. The physical equivalent would be a soap bubble adhering to a surface. Once the soap bubble touches the surface, the adhesiveness caused by the intermolecular forces will make the bubble stick to the surface. The bubble will ultimately take the shape of a hemisphere with its boundaries perpendicular to the surface.

Using the above boundary conditions (2.13) can be written as

$$\begin{aligned}\gamma'(0) &= d_{\psi}E(f_1, f_2, \phi) = \mu \int_{\Omega} |\nabla f_1|^2 H'(\phi)\psi + \int_{\Omega} |f_2 - g|^2 H'(\phi)\psi \\ &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H'(\phi)\psi) + \int_{\Omega} |f_2 - g|^2 (1 - H'(\phi)\psi) \\ &- \int_{\Omega} H'(\phi)\nabla \cdot \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) \psi.\end{aligned}\quad (2.15)$$

In order to conclude that  $\gamma'(0) = 0$ , then for any perturbation  $\psi$

$$H'(\phi) \left[ \nu \nabla \cdot \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) - |f_1 - g|^2 - \mu |\nabla f_1|^2 + |f_2 - g|^2 + \mu |\nabla f_2|^2 \right] = 0.$$

The descent direction for  $\phi$  is parameterized using an artificial "time"  $t \geq 0$  as  $\phi(\mathbf{x}, t)$  where  $\phi(\mathbf{x}, 0) = \phi_0(\mathbf{x})$ .

$$\frac{\partial\phi}{\partial t} = H'(\phi) \left[ \nu \nabla \cdot \left( \frac{\nabla\phi}{\|\nabla\phi\|} \right) - |f_1 - g|^2 - \mu |\nabla f_1|^2 + |f_2 - g|^2 + \mu |\nabla f_2|^2 \right] \quad (2.16)$$

### Computing the First Variation of $E$ With Respect to $f_1$ and $f_2$

So far the flow of  $\phi$  was found, but in order to follow the steepest descent the flow of (2.12), the necessary conditions for  $f_1$  and  $f_2$  must be computed also. Fixing  $\phi$  and  $f_2$  and following a similar approach as above

$$\begin{aligned}\gamma(t) &= E(f_1 + ts_1, f_2, \phi) = \mu \int_{\Omega} |\nabla f_1 + t\nabla s_1|^2 H(\phi) + \int_{\Omega} |f_1 + ts_1 - g|^2 H(\phi) \\ &+ \mu \int_{\Omega} |\nabla f_2|^2 (1 - H(\phi)) + \int_{\Omega} |f_2 - g|^2 (1 - H(\phi)) \\ &+ \nu \int_{\Omega} H'(\phi) |\nabla\phi|.\end{aligned}$$

Then

$$\gamma'(t) = \mu \int_{\Omega} (\nabla f_1 + t \nabla s_1) \nabla s_1 H(\phi) + \int_{\Omega} (f_1 + t s_1 - g) s_1 H(\phi)$$

In order to find the descent at  $f_1$  then  $t = 0$ , therefore

$$\gamma'(0) = \mu \int_{\Omega} \langle \nabla f_1, \nabla s_1 \rangle H(\phi) + \int_{\Omega} (f_1 - g) s_1 H(\phi)$$

Let  $\chi = \{\phi > 0\}$  and  $\partial\chi = \{\phi = 0\}$ , then the above equation can be written as

$$\gamma'(0) = \mu \int_{\chi} \langle \nabla f_1, \nabla s_1 \rangle + \int_{\chi} (f_1 - g) s_1$$

The first term in the last equation does not describe the variation in terms of the perturbation  $s_1$ , but using integration by parts [17] with the following substitutions

$$\begin{aligned} u &= \nabla f_1 & v' &= \nabla s_1 \\ u' &= \Delta f_1 & v &= s_1 \end{aligned} \tag{2.17}$$

then  $\gamma'(0)$  can be written as

$$\gamma'(0) = \mu \int_{\partial\chi} s_1 \langle \nabla f_1, \vec{n} \rangle - \mu \int_{\chi} \Delta f_1 s_1 + \int_{\chi} (f_1 - g) s_1 \tag{2.18}$$

where  $\vec{n}$  is the normal to the boundary  $\partial\chi$ .

Let  $s_1 = 0$  in a neighborhood  $\epsilon$  at the boundary  $\Gamma$ . Then

$$\int_{\partial\chi} s_1 \langle \nabla f_1, \vec{n} \rangle = 0$$

Since  $\gamma'(0) = 0$  then

$$-\mu \int_{\chi_\epsilon} \Delta f_1 s_1 + \int_{\chi_\epsilon} (f_1 - g) s_1 = 0$$

which implies

$$-\mu \Delta f_1 s_1 + f_1 - g = 0$$

Letting  $\epsilon \rightarrow 0$  and relaxing  $s_1$  to be any smooth perturbation then

$$\lim_{\epsilon \rightarrow 0} \left( -\mu \int_{\chi_\epsilon} \Delta f_1 s_1 + \int_{\chi_\epsilon} (f_1 - g) s_1 \right) = -\mu \int_{\chi} \Delta f_1 s_1 + \int_{\chi} (f_1 - g) s_1 = 0$$

Since this is valid in the entire  $\chi$ , then it remains that

$$\int_{\partial\chi} s_1 \langle \nabla f_1, \vec{n} \rangle = 0$$

for any variation  $s_1$  and so

$$\langle \nabla f_1, \vec{n} \rangle = 0$$

Then  $\gamma'(0)$  can be written as

$$\gamma'(0) = d_{s_1} E(f_1, f_2, \phi) = \int_{\chi} (f_1 - g) s_1 - \mu \int_{\chi} \Delta f_1 s_1 = 0$$

Letting  $\gamma'(0) = 0$  then for any perturbation  $s_1$ ,  $f_1$  can be solved as follows.

$$(1 - \mu \Delta) f_1 = g \tag{2.19}$$

The variation and necessary condition for  $f_2$  is found by the same analogy as  $f_1$ .

$$(1 - \mu \Delta) f_2 = g \tag{2.20}$$

### Summary of the Gradient Flow and Necessary Conditions

The gradient of the functional in (2.12) is described by

$$\begin{aligned}
 d_\psi E(f_1, f_2, \phi) &= \mu \int_\Omega |\nabla f_1|^2 H'(\phi) \psi + \int_\Omega |f_2 - g|^2 H'(\phi) \psi \\
 &+ \mu \int_\Omega |\nabla f_2|^2 (1 - H'(\phi)) \psi + \int_\Omega |f_2 - g|^2 (1 - H'(\phi)) \psi \\
 &- \int_\Omega H'(\phi) \nabla \cdot \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) \psi. \tag{2.21} \\
 d_{s_1} E(f_1, f_2, \phi) &= \int_\chi (f_1 - g) s_1 - \mu \int_\chi \Delta f_1 s_1 \\
 d_{s_2} E(f_1, f_2, \phi) &= \int_{\Omega \setminus \chi} (f_2 - g) s_2 - \mu \int_{\Omega \setminus \chi} \Delta f_2 s_2
 \end{aligned}$$

where  $\chi = \{\phi \geq 0\}$  and  $s_1, s_2$  and  $\psi$  are the perturbations on  $f_1, f_2$  and  $\phi$  respectively. With  $-\nabla E = 0$ , the above equations lead to the following necessary conditions.

$$\begin{aligned}
 \frac{\partial \phi}{\partial t} &= H'(\phi) \left[ \nu \nabla \cdot \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) - |f_1 - g|^2 - \mu |\nabla f_1|^2 + |f_2 - g|^2 + \mu |\nabla f_2|^2 \right] \\
 f_1 &= g + \mu \Delta f_1 \\
 f_2 &= g + \mu \Delta f_2
 \end{aligned} \tag{2.22}$$

In the next section the algorithm by which the minimizer is found using the system in (2.22) is described.

### Algorithm

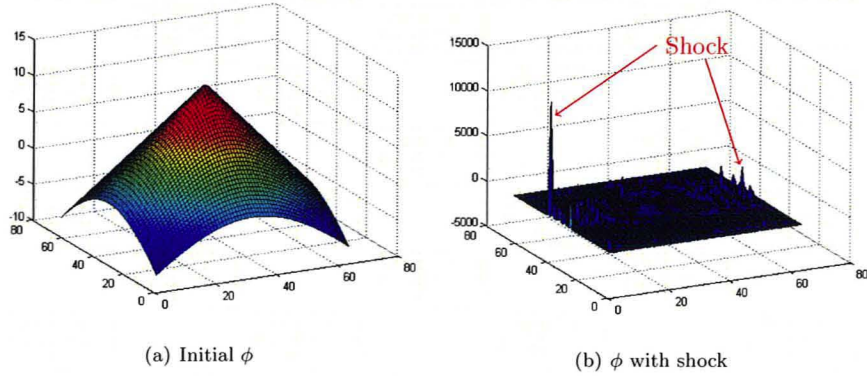
The standard steps to solve the Euler-Lagrange equations in (2.22) are described below. Assuming a time-discretization of  $\phi(t, \mathbf{x})$  as  $\phi(n\tau, \mathbf{x})$ , where  $\tau$  is the time-step and  $n$  a finite positive integer representing the iteration number then let  $(\phi)^n, (f_1)^n$  and  $(f_2)^n$  the respective function at iteration  $n$ . Some of the steps in the algorithm will be elaborated upon later.

1. Let  $n = 0$  be the initial level-set function  $\phi^0$
2. Calculate  $(f_1)^n$  on  $\{\phi \geq 0\}$  and  $(f_2)^n$  on  $\{\phi \leq 0\}$  using (2.19) and (2.20)
3. Extend by  $C^1$  functions  $(f_1)^n$  on  $\{\phi < 0\}$  and  $(f_2)^n$  on  $\{\phi > 0\}$  in a small neighborhood near the  $\{\phi = 0\}$
4. Solve (2.16) to obtain  $\phi^{n+1}$
5. Re-initialize  $\phi^n$  to the signed distance function to the curve. This is done only near the zero level-set [32]

The discretization scheme will be explained in a later section, but for now the algorithm is presented to give an idea of the typical steps in the evolution of  $\phi, f_1$  and  $f_2$ .

In the third step, the extension of  $f_1$  and  $f_2$  is necessary since, in practice, these have to be computed near a narrow band on the boundary, but neither is defined across the boundary. There are a number of options to perform this extension the details of which are described in [36, 40, 41, 43, 52].

In step five, the re-initialization of the function  $\phi$  is important due to the fact that the level-set function develops shocks, a very sharp or flat shape during evolution (see Figure 2.1.3), which can make further computations inaccurate [5]. The effect of



**Figure 2.2:** The (a) initial  $\phi$  develops (b) shocks during evolution.

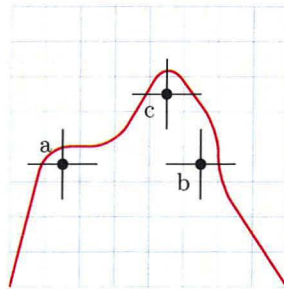
shocks can be removed using re-initialization. For instance,  $\phi$  can be re-initialized using the signed distance function at its zero level-set as described in [16, 32]:

$$\begin{aligned} \psi_\tau &= \text{sign}(\phi(t))(1 - |\nabla\psi|) \\ \psi(0, \cdot) &= \phi(0, \cdot) \end{aligned} \quad (2.23)$$

Here,  $\phi(t, \cdot)$  is  $\phi$  at time  $t$ . The new  $\phi(t, \cdot)$  will be  $\psi$ , such that  $\psi$  is obtained at the steady-state of (2.23). The solution of  $\psi(t, \cdot)$  will have the same zero level-set as  $\phi(t, \cdot)$  and away from the contour  $|\nabla\psi|$  converges to 1.

The method proposed by Chan and Vese [49] on the first functional proposed by Mumford and Shah [10] is effective for de-noising signals as well as segmentation. The drawback of this model is the continuity conditions over the boundary  $\Gamma$ . This adds some difficulty to implementation due to the fact that some special cases have to be considered for the calculation of the gradient on the boundary. As shown in Figure 2.3 not all neighboring points of a point are inside the same region therefore assumptions must be made when solving the problem in a discrete setting.

This approach to segmentation, however, is more complex than needed for the purpose of the project of this thesis since it both segments and de-noises the image. The purpose of the segmentation step for this project is to extract a single organ, namely the lungs, from the body of a subject without retaining additional information about the measured values in the image so no de-noising is required.



**Figure 2.3:** Visualization of the curve  $\Gamma$  passing in a grid showing points  $a$ ,  $b$  and  $c$  on the boundary with a different number of neighboring points lying outside their region.

A simplified approach proposed in [50] by Chan and Vese partitions the domain of the image according to the average values inside and outside the curve  $\Gamma$ . This is a restriction on the approach just discussed where  $f_1 = a_1$  and  $f_2 = a_2$ .

### 2.1.4 Chan-Vese Approach to Segmentation

In this section a reduced case of the functional treated so far proposed by Chan and Vese [50] will be discussed. The basic idea of this approach is to separate components in an image into background and foreground. Specifically, all regions enclosed by the curve  $\Gamma$  are part of the foreground and the rest is background.

Many approaches to segmentation make use of the edges in the image [31, 37, 53, 54]. In the Mumford-Shah model the image  $g$  is assumed to be smooth due to the fact that direct use of  $|\nabla g|$  is made. Other methods try to relax this assumption by introducing an edge detector that makes use of a smoothing operator, such as the following

$$\eta(|\nabla g(x, y)|) = \frac{1}{1 + |\nabla_{\sigma} g(x, y)|^p}, \quad p \geq 1 \quad (2.24)$$

where  $\nabla_{\sigma} g(x, y) = \nabla[G_{\sigma}(x, y) * g(x, y)]$ .  $G_{\sigma} * g$  produces a smoother version of  $g$  since this is typically the convolution of an image  $g$  with a Gaussian function  $G_{\sigma}(x, y) = \sigma^{-1/2} e^{-|x^2+y^2|/4\sigma}$  where  $\sigma > 0$  is the spread. The resulting image  $\eta(|\nabla g|)$  is positive in homogeneous regions and close to zero at the edges.

The purpose of an edge detector is to have the curve evolution stop at the points where  $\eta$  is zero. However, this is true only for the highest edges in  $g$ . Other edges in  $g$  that may indeed be of interest to the problem may have values away from zero in  $\eta$  and therefore the curve might not stop there. Furthermore, when an image contains a lot of noise the Gaussian smoothing applied to the image should be stronger and this means that sharp edges are smoothed as well.

#### Problem Statement

The functional constructed by Chan and Vese in [50] will be the focus of this section. This functional closely resembles a reduced case of the functional (2.7) where the domain  $\Omega$  is partitioned only in two regions  $\Sigma_{out}$  and  $\Sigma_{in}$  outside and inside  $\Gamma$  respectively.

The model is the following. Assume that the image  $g$  is formed by two regions of approximately piece-wise constant intensities with values  $g_{out}$  and  $g_{in}$  and that the object to be detected is represented by the value  $g_{in}$  with a boundary  $\Gamma_0$  separating it from the other value. Then  $g$  is approximately the same to  $g_{in}$  inside  $\Gamma_0$  and to  $g_{out}$  outside  $\Gamma_0$ . Letting the region inside  $\Gamma$  be denoted by  $\chi$  then the fitting term then can be expressed as

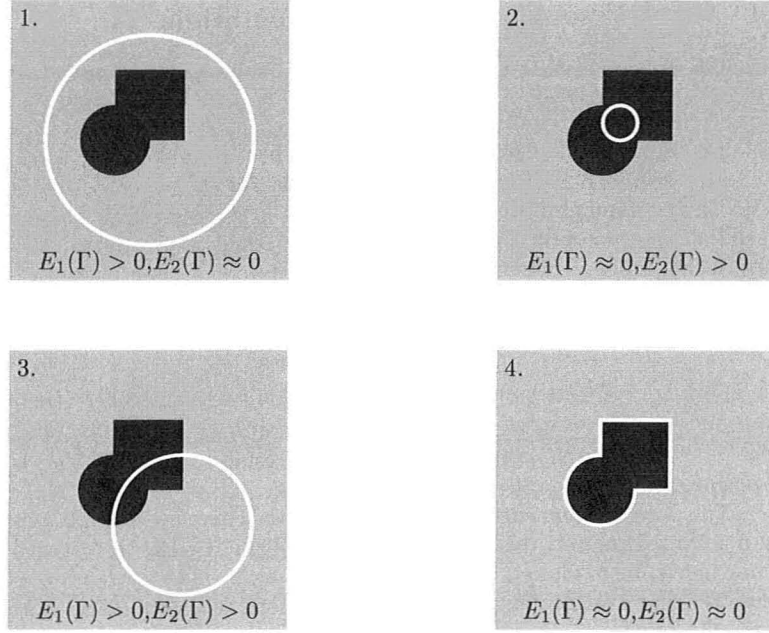
$$E_1(\Gamma) + E_2(\Gamma) = \int_{\Sigma_{in}} |g - c_1|^2 + \int_{\Sigma_{out}} |g - c_2|^2 \quad (2.25)$$

where  $\Gamma$  is any variable curve, and the constants  $c_1$  and  $c_2$  that depend on  $\Gamma$  represent the average value of  $g$  inside and outside  $\Gamma$  respectively. For the image just described the curve  $\Gamma_0$  is the minimizer. This explanation is depicted below in Figure 2.4.

Using the level-set formulation, assuming a single level-set function  $\phi$  then the full form of the Chan-Vese functional is the following

$$\begin{aligned} E(c_1, c_2, \phi) &= \mu \int_{\Omega} H'(\phi) |\nabla \phi| \\ &+ \nu \int_{\Omega} H(\phi) \\ &+ \lambda_1 \int_{\Omega} |u - c_1|^2 H(\phi) \\ &+ \lambda_2 \int_{\Omega} |u - c_2|^2 (1 - H(\phi)) \end{aligned} \quad (2.26)$$

where the first term represents the length of the curve, the second term represents the area inside the curve or  $\{\phi \geq 0\}$  and the last two are the data-fitting terms with parameters  $\mu, \nu, \lambda_1$  and  $\lambda_2$  greater than zero.



**Figure 2.4:** Typical cases of the data-fitting term depending on the position of the curve, see [50].

The minimizer of the functional in (2.27) can be found using a variational approach as done before. By fixing  $\phi$  the analytical solutions to  $c_1$  and  $c_2$  are found as

$$c_1(\phi) = \frac{\int_{\Omega} H(\phi)g}{\int_{\Omega} H(\phi)} \quad (2.27)$$

$$c_2(\phi) = \frac{\int_{\Omega} (1 - H(\phi))g}{\int_{\Omega} 1 - H(\phi)} \quad (2.28)$$

Here  $c_1$  turns out to be the average value of the image  $g$  inside the curve  $\Gamma$  and  $c_2$  the average value outside the curve  $\Gamma$ . Analogically, a first variation on  $\phi$  can be taken, obtaining the following flow.

$$\frac{\partial \phi}{\partial t} = H'(\phi) \left[ \mu \nabla \cdot \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - \nu - \lambda_1 |u - c_1|^2 + \lambda_2 |u - c_2|^2 \right] \quad (2.29)$$

The minimizer  $(c_1^*, c_2^*, \phi^*)$  can be found by solving the Euler-Lagrange equations in (2.27), (2.28) and (2.29).

### Discretization

In this section, the discretization of the Chan-Vese approach in [50] will be treated in detail. The discretization of this approach is of importance since the same discretization is used in the segmentation approach developed in the project of this thesis.

The system can be discretized and solved numerically as proposed in [12, 14, 46]. Given that the data in this project is 3D then the following discretization will assume 3D data, however, the visualizations will be in 2D for simplicity.

First, let the discretization in "time" be  $\phi(n\tau, \mathbf{x}) = \phi^n$  and

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\tau} \quad (2.30)$$



where  $\tau$  is the time step and  $n$  is a finite positive integer representing the iteration number. The constants defined in (2.27) and (2.28) can then be written as

$$c_1^n = c_1(\phi^n), \quad c_2^n = c_2(\phi^n) \quad (2.31)$$

and (2.29) can be written as

$$\frac{\phi^{n+1} - \phi^n}{\tau} = H'(\phi^n) \left[ \mu \nabla \cdot \left( \frac{\nabla \phi^{n+1}}{|\nabla \phi^n|} \right) - \nu - \lambda_1 |g - c_1^n|^2 + \lambda_2 |g - c_2^n|^2 \right] \quad (2.32)$$

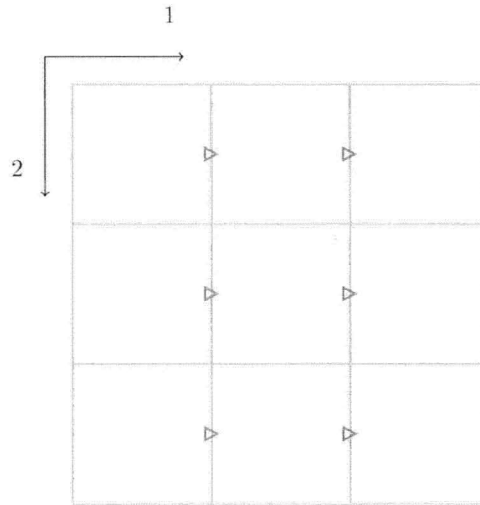
The next step is to discretize the image, the level-set function and the gradient and divergence operators in (2.32). In this context, the values in the image and level-set function are cell-centered [22], therefore cell  $g_{i,j,k}$  centered at  $(x_i, y_j, z_k)$  has value  $g(x_i, y_j, z_k)$ . Analogously,  $\phi_{i,j,k} = \phi(x_i, y_j, z_k)$  with centre  $(x_i, y_j, z_k)$ .

In order to discretize the gradient and divergence operators in (2.32) the following difference operators are used.

$$\partial^+ = \frac{1}{h} \begin{pmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -1 \end{pmatrix} \in \mathbb{R}^{m-1,m} \quad (2.33)$$

$$\partial^- = -(\partial^+)^T \in \mathbb{R}^{m,m-1} \quad (2.34)$$

where  $h$  is the discretization of the domain  $\Omega$  according to the resolution of the data. The difference operators above calculate the staggered (forward and backward) difference in the direction applied in order to avoid boundary assumptions. As shown in Figure 2.5 below, if the image has size  $m$  in direction 1, after applying  $\partial^+$  the difference obtained in that direction will be of size  $m - 1$ .



**Figure 2.5:** Original grid of image (gray) with location of calculated forward difference values (red triangles) in direction 1.

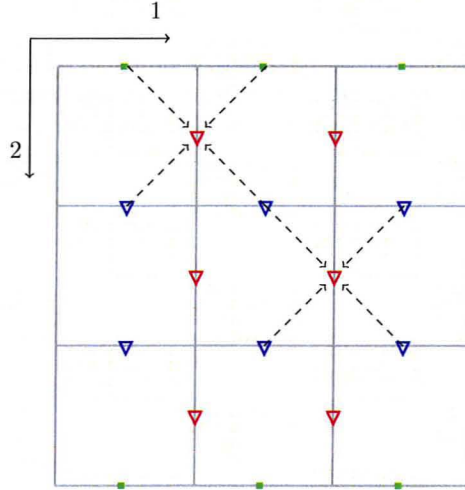
Additionally, there are two averaging operators  $\bar{A}$  and  $\hat{A}$  that for the most part

$\mathring{\mathcal{A}}$  is similar to  $\bar{\mathcal{A}}^T$ , but differ in two elements as shown below

$$\mathring{\mathcal{A}} = \frac{1}{2} \begin{pmatrix} 2 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 1 & 1 \\ 0 & \dots & 0 & 2 \end{pmatrix} \in \mathbb{R}^{m,m-1} \quad (2.35)$$

$$\bar{\mathcal{A}} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 1 \end{pmatrix} \in \mathbb{R}^{m-1,m} \quad (2.36)$$

These two operators are used in combination with the difference operators.  $\mathring{\mathcal{A}}$ , when applied after  $\partial^+$ , assumes that the slope extends over the boundary of  $\Omega$  linearly and computes the slope by averaging at the cell centers.  $\bar{\mathcal{A}}$  brings the cell-centered difference to the location of the staggered difference of the other direction. The combination of these two operators facilitates the computation of the magnitude of the gradient as will be explained later. In order to have a better idea of the purpose of the averaging operators assume that the staggered difference of  $\phi$  was computed in direction 1 as shown in Figure 2.5. In order to calculate an approximated difference in direction 2 that fits direction 1, then the staggered difference  $\partial^+$  can be taken in the second direction and averaged to fit the same grid as that of  $\partial^+$  applied in direction 1. Figure 2.6 provides a visual aid to the above explanation.



**Figure 2.6:** Original grid of  $\phi$  (gray) with location of calculated forward difference values in direction 2 (blue triangles) with assumed difference on boundary (green) showing the path by which the combined averaging operators compute and relocate the staggered difference to direction 1 (red triangles).

Letting a vectorized version of the image  $g$  and the level-set function  $\phi$  be denoted by  $\vec{g}$  and  $\vec{\phi}$  respectively one may use the following difference operators on  $\vec{g}$  and  $\vec{\phi}$ .

$$\begin{aligned} \partial_1^+ &= \mathbf{I} \otimes \mathbf{I} \otimes \partial^+ \in \mathbb{R}^{(knl, kn(l-1))} \\ \partial_2^+ &= \mathbf{I} \otimes \partial^+ \otimes \mathbf{I} \in \mathbb{R}^{(knl, k(n-1)l)} \\ \partial_3^+ &= \partial^+ \otimes \mathbf{I} \otimes \mathbf{I} \in \mathbb{R}^{(knl, (k-1)nl)} \end{aligned} \quad (2.37)$$

where  $\otimes$  is the Kronecker product,  $\mathbf{I}$  is the identity matrix and  $k$ ,  $n$  and  $l$  are the sizes of the data in the directions 3, 2 and 1 respectively. The operator  $\partial_d^+$  computes the

staggered difference in the direction  $d$ . To fit to  $\vec{g}$  and  $\vec{\phi}$  the averaging operators are extended as follows.

$$\begin{aligned}
 \mathcal{A}_{1,2} &= \mathbf{I} \otimes \bar{\mathcal{A}} \otimes \hat{\mathcal{A}} \in \mathbb{R}^{(k(n-1)l, kn(l-1))} \\
 \mathcal{A}_{1,3} &= \bar{\mathcal{A}} \otimes \mathbf{I} \otimes \hat{\mathcal{A}} \in \mathbb{R}^{((k-1)nl, kn(l-1))} \\
 \mathcal{A}_{2,1} &= \mathbf{I} \otimes \hat{\mathcal{A}} \otimes \bar{\mathcal{A}} \in \mathbb{R}^{(kn(l-1), k(n-1)l)} \\
 \mathcal{A}_{2,3} &= \bar{\mathcal{A}} \otimes \mathbf{I} \otimes \hat{\mathcal{A}} \in \mathbb{R}^{((k-1)ln, kl(n-1))} \\
 \mathcal{A}_{3,1} &= \hat{\mathcal{A}} \otimes \mathbf{I} \otimes \bar{\mathcal{A}} \in \mathbb{R}^{(kn(l-1), (k-1)nl)} \\
 \mathcal{A}_{3,2} &= \hat{\mathcal{A}} \otimes \hat{\mathcal{A}} \otimes \mathbf{I} \in \mathbb{R}^{(k(n-1)l, (k-1)nl)}
 \end{aligned} \tag{2.38}$$

The purpose of the operator  $\mathcal{A}_{d_1, d_2}$  is to “relocate” the calculated difference in direction  $d_1$  to the position of the difference in direction  $d_2$ . For instance, the location of the values computed by  $\mathcal{A}_{2,1} \partial_1^+ \phi$  is the same as those computed by  $\partial_2^+ \phi$ . The purpose of the averaging operators becomes clearer when looking at the following difference operators combined in the calculation of the magnitude of the gradient as follows:

$$\begin{aligned}
 \partial_{1,2} &= \mathcal{A}_{1,2} \partial_2^+ \\
 \partial_{1,3} &= \mathcal{A}_{1,3} \partial_3^+ \\
 \partial_{2,1} &= \mathcal{A}_{2,1} \partial_1^+ \\
 \partial_{2,3} &= \mathcal{A}_{2,3} \partial_3^+ \\
 \partial_{3,1} &= \mathcal{A}_{3,1} \partial_1^+ \\
 \partial_{3,2} &= \mathcal{A}_{3,2} \partial_2^+
 \end{aligned} \tag{2.39}$$

With these averaging operators the magnitude of the gradient of  $\phi$  can be calculated as

$$\begin{aligned}
 G_1(\phi) &= \sqrt{(\partial_1^+ \phi)^2 + (\partial_{1,2} \phi)^2 + (\partial_{1,3} \phi)^2} \\
 G_2(\phi) &= \sqrt{(\partial_{2,1} \phi)^2 + (\partial_2^+ \phi)^2 + (\partial_{2,3} \phi)^2} \\
 G_3(\phi) &= \sqrt{(\partial_{3,1} \phi)^2 + (\partial_{3,2} \phi)^2 + (\partial_3^+ \phi)^2}.
 \end{aligned} \tag{2.40}$$

Each one of the above is an approximation of  $|\nabla \phi|$ .  $G_d(\phi)$  calculates the difference in dimension  $d$  exactly, but in the other dimensions the difference is approximated due to assumptions over the boundary of the domain  $\Omega$ .

Finally, the divergence operator can be approximated by an operator  $A(\phi)$  of the following form.

$$A(\phi) = \sum_{i=1}^3 \partial_i^- \frac{1}{\hat{G}_i(\phi)} \partial_i^+ = - \sum_{i=1}^3 (\partial_i^+)^T \frac{1}{\hat{G}_i(\phi)} \partial_i^+ \tag{2.41}$$

where  $1/\hat{G}_i(\phi)$  is a square matrix in which the values in the main diagonal are those of  $G_i(\phi)$  inverted. This discretization of the divergence operator was suggested in [46, 50].

At this point equation (2.32) can be written as

$$\frac{\phi^{n+1} - \phi^n}{\tau} = H'(\phi^n) [\mu A(\phi^n) \phi^{n+1} - \nu - \lambda_1 |g - c_1^n|^2 + \lambda_2 |g - c_2^n|^2] \tag{2.42}$$

Using some mathematical manipulations the following form is obtained.

$$(\mathbf{I} - \tau \mu H'(\phi^n) A(\phi^n)) \phi^{n+1} = \phi^n + \tau H'(\phi^n) [\nu - \lambda_1 |g - c_1^n|^2 + \lambda_2 |g - c_2^n|^2] \tag{2.43}$$

Equation (2.43) can be solved iteratively using the steps described below.

Letting the iteration step be denoted by  $n$ , then the principal steps of the algorithm are the following.

1. Initialize the level-set function  $\phi^0 = \phi_0$  at  $n = 0$ .
2. Compute  $c_1(\phi^n)$  and  $c_2(\phi^n)$  by (2.27) and (2.28).
3. Solve (2.43) to obtain  $\phi^{n+1}$ .
4. **Optional Step:** Re-initialize  $\phi$  on a small neighborhood of  $\Gamma$  to the signed distance function as shown in (2.23).
5. Check if solution is stationary by using termination criteria as described in [34] on  $(\phi, c_1, c_2)$ ,  $E$  and  $\nabla E$ . If solution is not stationary,  $n = n + 1$  and repeat from step 2.

This last method proposed by Chan and Vese in [50] deals with the partitioning of an image into two possible regions with distinct constant values. Assumptions need only to be made about the continuity of  $\phi$  on the boundary  $\partial\Omega$ .

Since this problem is also a special case of the second Mumford-Shah functional in (2.7) for  $i \in \{1, 2\}$  then it is well-posed. This was proven initially by Mumford and Shah in [10] for smooth images and later in [20] for more general data.

To summarize, in this section a brief overview on some of the most influential methods for segmentation was described. The Chan-Vese methods proposed in [49, 50] were described in more detail in terms of the variational approach to solving the minimization problem and discretization.

## 2.2 Registration Methods

In this section a quick overview of the registration method used for this project will be discussed. Since this is not the subject of focus and also due to the fact the registration software in the FAIR framework [22] was used “out of the box” the thorough details of the representation of the images, the interpolations and the transformations will not be discussed. The actual objective functions used for the registration of the ellipsoids and of the template and reference images are discussed later in section 3.2.

### 2.2.1 The Basic Concept of Registration

In general, the problem in image registration is the following: Given an image  $\mathcal{T}$  called a template and an image  $\mathcal{R}$  called a reference where  $\mathcal{T}, \mathcal{R} : \Omega \in \mathbb{R}^d \rightarrow \mathbb{R}^d$ , find a suitable transformation  $y : \Omega \rightarrow \mathbb{R}^d$  such that the transformed  $\mathcal{T}$  is similar to the image  $\mathcal{R}$ . This can be written as an optimization problem where a joint functional such as the following must be minimized.

$$\mathcal{J}[y] = \mathcal{D}[\mathcal{T}[y], \mathcal{R}] + \alpha \mathcal{S}[y - y^{\text{ref}}] \quad (2.44)$$

Here,  $\mathcal{T}[y] = \mathcal{T}(y(x))$  is the transformed template image,  $\mathcal{D}$  is the image similarity measure and  $\mathcal{S}$  measures the reasonability of the transform.

One way to measure the similarity of images is to measure the sum of squared differences or the energy of the difference image. In a continuous setting

$$\mathcal{D}[\mathcal{T}[y], \mathcal{R}] = \frac{1}{2} \int_{\Omega} (\mathcal{T}(y(x)) - \mathcal{R}(x))^2 dx \quad (2.45)$$

Since non-rigid registration is an ill-posed problem [21, 30], regularization is necessary. The transformation reasonability term in (2.44) regularizes the problem. There is a number of choices for measuring the reasonability of the transformation  $y$  [21, 22], but these are not discussed here due to the fact that, for this project,  $y$  is restricted to rigid transformations, therefore the joint functional (2.44) simplifies to

$$\mathcal{J}[y] = \mathcal{D}[\mathcal{T}[y], \mathcal{R}]. \quad (2.46)$$

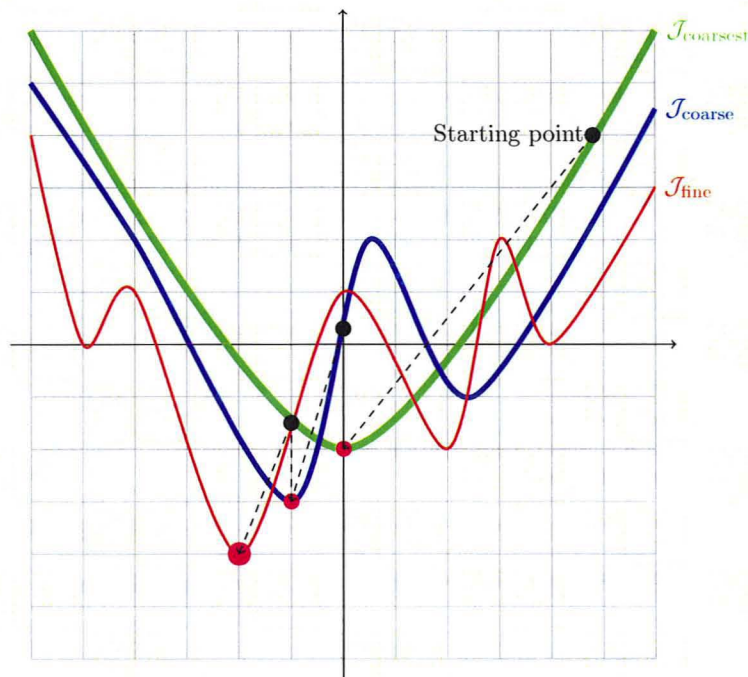
The transformation  $y$  is parameterized by a set of parameters  $\omega$  as  $y(\omega, x)$  that restrict the transformation to rotation and translation and therefore the registration becomes parameterized. As will be seen in Section 3.2, the distance measure used in this project is the sum of squared differences as proposed in (2.45).

In FAIR, image registration is facilitated by the built-in Gauss-Newton method with Armijo line-search (GNA). The GNA method need only be provided with the objective function, the gradient and the Hessian. The termination criteria are handled internally using the default parameters which can be overridden if necessary.

### 2.2.2 Multi-Level Registration

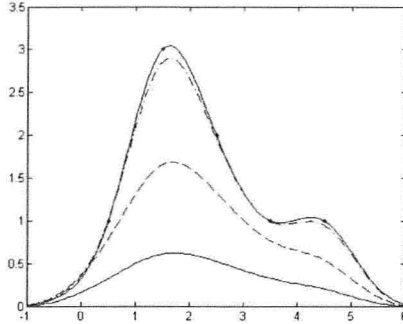
The FAIR framework allows for different of types of registration including multi-level, multi-scale or fixed grid. In the case of fixed grid, the registration is done at a single resolution without any modifications, however, this is not a recommended strategy [21, 22] due to the fact that the minimization algorithm may find a local minimum. In order to get an idea of why a fixed grid is not recommended, one can think of the images of two hands at a fine scale rotating on top of each other. When two fingers are on top of each other, a local minimum is found. However, the global minimum is found only when all five fingers are on top of each other.

Multi-scale and multi-level strategies are more effective in avoiding local minima due to the fact that registration starts by using very coarse details, such as position, orientation and size, and continues using incrementally finer details. Figure 2.9 shows the differences between multi-scale and multi-level strategies in the same image with different levels of coarseness. Figure 2.7 shows three typical joint functionals at different coarseness levels.



**Figure 2.7:** Functionals at different levels of coarseness. Using the starting point shown, the minimizer of  $J_{\text{coarsest}}$  is a good starting guess for  $J_{\text{coarse}}$  and in turn the minimizer of the last one is a good starting guess for  $J_{\text{fine}}$ . Using the multi-level or multi-scale strategy local minima in  $J_{\text{fine}}$  can be avoided.

In multi-scale registration an image is registered in multiple scales therefore a



**Figure 2.8:** Spline approximation with varying (smoothness) parameter  $\theta$ ,  $\theta = 0$  (solid line),  $\theta = 1$  (dashed),  $\theta = 10$  (dashed dotted),  $\theta = 100$  (dotted). The parameter  $\theta$  is a weighting factor controlling the smoothness of interpolating function. Example produced using function `E3MSsplineInterpolation1D` from the FAIR Framework [22].

scale-space representation of the image is needed. One method to obtain the scale-space representation of an image is to use a parameter which controls the smoothness of an interpolating function. In this thesis, this parameter will be referred to as the scale-space parameter. Figure 2.8 shows a 1D example of a set of points interpolated using a varying scale-space parameter  $\theta$ . Multi-scale registration is typically performed starting with a coarse scale or a large scale-space parameter and then gradually decreasing the scale-space parameter in order to register the image at finer scales.

In multi-level registration an image is registered in multiple resolutions. Registration is performed starting from the lowest desired resolution then increasing the resolution in order to register the image at finer levels. Figure 2.9 shows the differences between multi-scale and multi-level strategies in the same image at different levels of coarseness.

While both multi-scale and multi-level registration are effective in evading local minima, the multi-level strategy was chosen for the project-specific application. In order to perform registration using a multi-level approach as shown in Figure 2.7 the algorithm shown in 2.1 is used.

---

**Algorithm 2.1** Pseudocode used for multi-level registration, see [22].

---

```

 $\mathcal{T}, \mathcal{R}$  at  $l_{\max}$ 
for  $l = l_{\min}$  to  $l_{\max}$  do
    transfer images to level  $l$ 
    register  $\mathcal{T}_l$  to  $\mathcal{R}_l$ : find transformation  $y_l$ 
    use  $y_l$  in the next level
end for
    
```

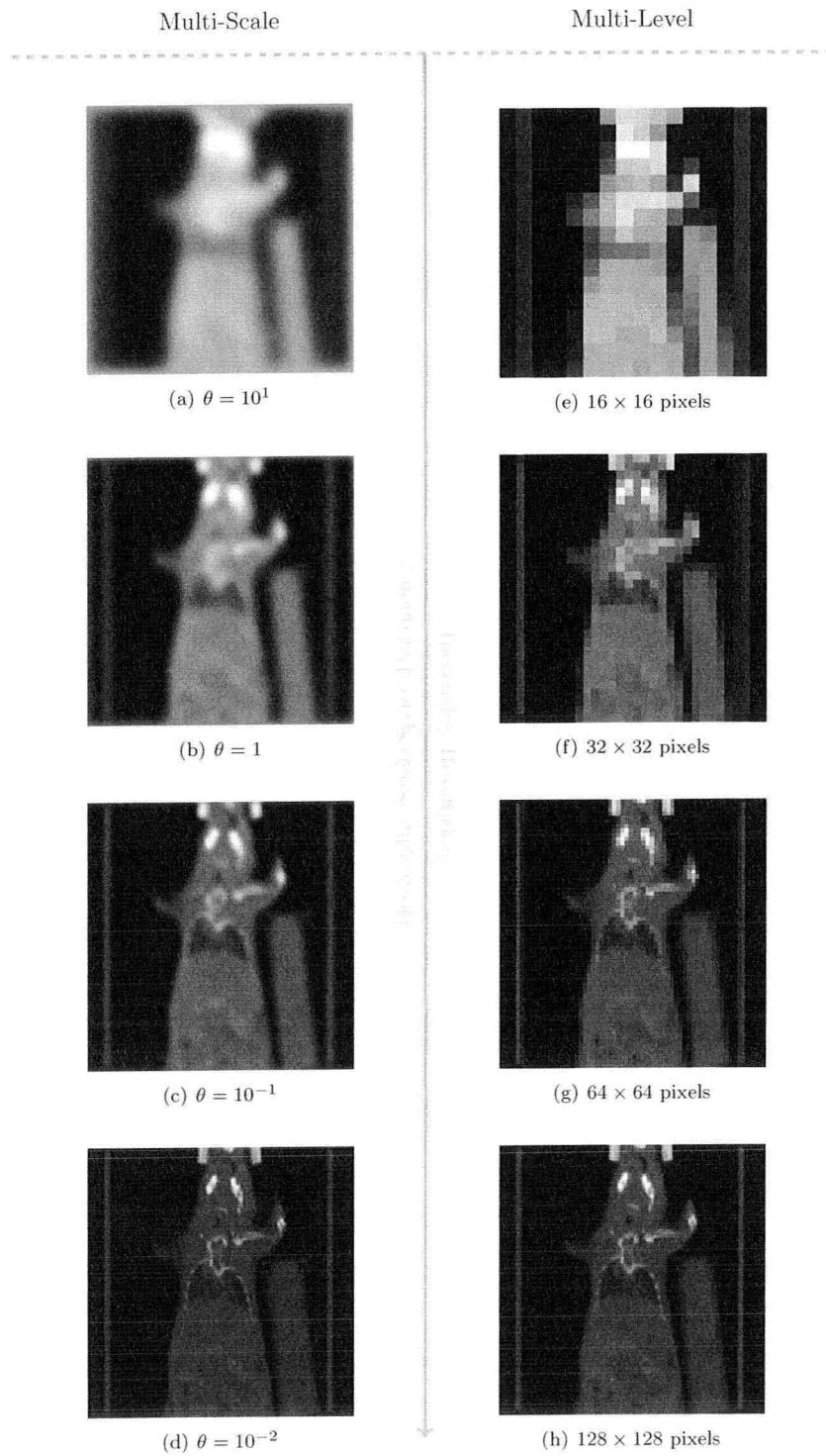
---

To summarize, the multi-level registration approach is chosen for this project because

- it focuses on essential minima,
- the starting value in consecutive levels is an educated guess towards the minimizer
- and reduces computation time.

To conclude, this chapter has provided an overview of the base theory behind the segmentation used in this project as well as an overview of the multi-level registration approach supported by the FAIR framework. More details about registration and

segmentation using the level-set approach can be found in [6, 10, 21, 22, 24, 47–50]. In the next chapter the details pertaining to the registration and segmentation used in this project will be presented along with a quick overview of the termination criteria used for the segmentation method.



**Figure 2.9:** Coronal slice of a mouse shown in different scales (Figure 2.9(a)–2.9(d)), using a varying scale-space parameter  $\theta$ , and different levels (Figure 2.9(e)–2.9(h)), using multiple resolutions.



## Chapter 3

# Project-Specific Methods for Segmentation and Registration

This chapter will focus on the project-specific objective functions and functionals used in registration and segmentation respectively. First, the focus will be on the modification of the Chan-Vese objective functional for segmentation [50] to meet the needs of the project-specific application. Following this, the objective functions used in the registration of the ellipsoids with the  $P_{SP}$  data and the segmented data along with the objective functions used for the ROI-based data registration will be discussed. Finally, an overview of the termination criteria for the segmentation and registration will be presented.

### 3.1 The Modification of the Chan-Vese Functional for Segmentation

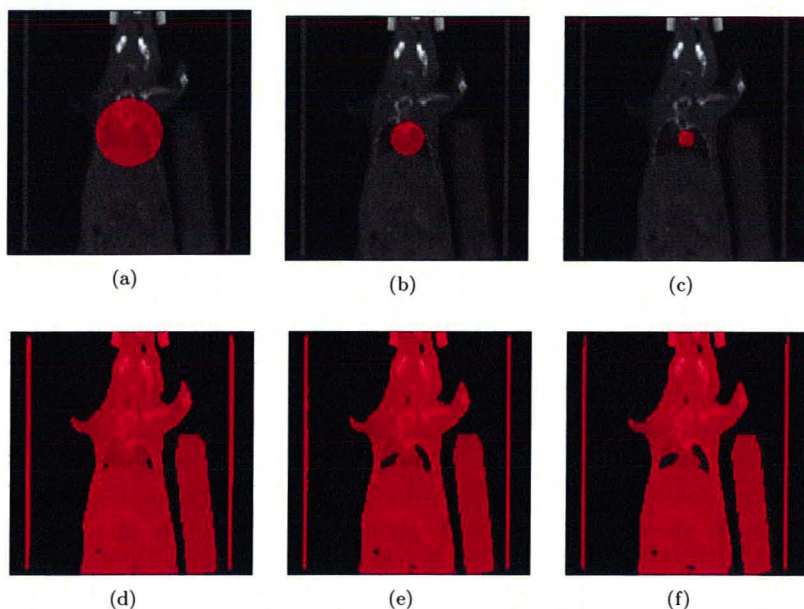
In this section, the project specific segmentation method will be discussed in detail starting with the ideas behind the construction of the objective functional. The discretization scheme, the algorithm used to satisfy the necessary conditions and the termination criteria used to check for stationarity on the segmentation algorithm will follow. The section concludes with a description of a number of alternative of objective functionals developed in the course of the thesis project

#### 3.1.1 Problem Statement and Model

When the Chan-Vese functional was tried on the project data it did not provide the expected result “out of the box” even though the initial guess is proximal to the expected result as shown in Figures 3.1(a)–3.1(c). Even with some fine-tuning of the parameters  $\nu$ ,  $\mu$ ,  $\lambda_1$  and  $\lambda_2$  the results were not satisfactory. For instance the parameters can be tuned for the initial guess in Figure 3.1(c), however this same set of parameters would not work for the initial guesses in Figures 3.1(a) and 3.1(b). As can be seen in Figure 3.1 the Chan-Vese segmentation method segments the entire image, but does not focus on a specific region as is required in the application of this thesis.

The first condition for the objective functional in this project is that, using the same set of parameters, example cases such as those shown in Figures 3.1(a) to 3.1(c) should segment the lungs. In other words, if the guess is larger, approximate or smaller to the volume of the lungs the resulting segmentation should show a “good” approximation of the lungs.

In order to achieve a good segmentation on the lungs some information about the underlying data can be used.



**Figure 3.1:** Typical initial guesses 3.1(a)–3.1(c) and results 3.1(d)–3.1(f) of the Chan-Vese segmentation method with segmented area shown in red. The parameters were set as suggested in [50]:  $\mu = 0.1 \cdot 255^2$ ,  $\nu = 0$ ,  $\lambda_1 = 1$  and  $\lambda_2 = 1$ .

1. One can use the fact that the underlying data is CT and therefore provides information about tissue density in Hounsfield units. By observing the data samples provided by Dr. Farncombe and the McMaster Centre for Preclinical Imaging, the lungs of the mouse range between  $-500HU$  and  $-100HU$ , so a rough approximation of the average value in the lungs in a CT scan would be  $-300HU$ . In humans the typical range of intensity values in the lungs is larger (see Table 2.2 in [9]).
2. In real-life experiments the subject exhibits high Hounsfield values around the lungs, typically greater than  $200HU$ , due to the hard tissue of the ribs [9]. This leads to the conclusion that in most subjects there is a high gradient between the inner part of the chest and the ribs.
3. The tidal volume of the lungs of the subject can be estimated by observation or by using statistical data [42].

From these three pieces of information the first and third were crucial in providing the expected result.

The most successful objective functional so far has been the following.

$$E(\phi) = \mu \int_{\Omega} H'(\phi) |\nabla \phi| + \nu \left| \mathbf{V}_0 - \int_{\Omega} H(\phi) \right| + \lambda \int_{\Omega} |g - c_0|^2 H(\phi) \quad (3.1)$$

where  $\mathbf{V}_0$  is a constant representing the approximate estimate of the volume of the lungs and  $c_0$  is a constant representing the average Hounsfield Unit value in the lungs. The parameters  $\mu$ ,  $\nu$  and  $\lambda$  are positive scalars which are chosen empirically. The first term in equation (3.1) asks that the curve  $\Gamma = \{\phi = 0\}$  be as short as possible; the second term asks that the volume inside  $\{\phi = 0\}$  be as close as possible to the volume  $\mathbf{V}_0$  defined by the user; the third term asks that the region of  $g$  inside the curve  $\Gamma$  be as close as possible to the user estimated average Hounsfield unit value  $c_0$  in the lungs.

An outline of the other objective functionals constructed will be presented in the next section in order to give an idea of the reasoning behind them. The main difference between the functional (3.1) and the other functionals is that the first makes use of two pieces of information based on the properties of CT data and the properties of the subject itself while the others are an attempt at generalizing the segmentation by using only the properties of the CT data.

The synthetic and practical experiments as well as the fine-tuning of the parameters carried out to test functional (3.1) will be discussed in Chapter 4 in more detail. In this chapter the focus remains on the functional (3.1).

In this thesis the problem of finding a minimizer for (3.1) is assumed to be well-posed since a rigorous proof is beyond the scope of this thesis. It is also assumed that  $g$  is measurable and square integrable. Furthermore, the problem is not convex since the solution depends on the starting guess  $\phi_0$ . However, in the project of this thesis effort was put to make the initial guess  $\phi_0$  vary as little as possible.

In order to find the evolution of  $\phi$  the same variational approach employed by Mumford-Shah and Chan-Vese is used along with the same boundary assumptions. The necessary condition for  $\phi$  is

$$\frac{\partial \phi}{\partial t} = H'(\phi) \left[ \mu \nabla \cdot \left( \frac{\nabla \phi}{|\nabla \phi|} \right) + \nu \frac{\mathbf{V}_0 - \int_{\Omega} H(\phi)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi)|} - \lambda |g - c_0|^2 \right] \quad (3.2)$$

Note that if the minimizer  $\phi^*$  satisfies  $\mathbf{V}_0 = \int_{\Omega} H(\phi^*)$  then the fraction  $\frac{\mathbf{V}_0 - \int_{\Omega} H(\phi^*)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi^*)|}$  is undefined. For this reason a small constant  $\beta \geq 0$  is introduced and (3.2) is modified to

$$\frac{\partial \phi}{\partial t} = H'(\phi) \left[ \mu \nabla \cdot \left( \frac{\nabla \phi}{|\nabla \phi|} \right) + \nu \frac{\mathbf{V}_0 - \int_{\Omega} H(\phi)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi)| + \beta} - \lambda |g - c_0|^2 \right] \quad (3.3)$$

One may suggest that using a slightly modified functional from that proposed in (3.1) such as

$$E(\phi) = \mu \int_{\Omega} H'(\phi) |\nabla \phi| + \nu \left| \mathbf{V}_0 - \int_{\Omega} H(\phi) \right|^2 + \lambda \int_{\Omega} |g - c_0|^2 H(\phi) \quad (3.4)$$

similar results may be obtained without the need for a regularizer  $\beta$ . In fact, the objective functional (3.4) does obtain similar results with some fine-tuning on the parameters, however, not as much time was spent in fine-tuning the parameters of (3.4). A comparison of the results obtained with (3.1) and (3.4) is outside the scope of this thesis due to time limitations, but will be considered for future research.

### 3.1.2 Discretization, Implementation and Termination Criteria for Segmentation

#### Discretization

The same discretization as in the Chan-Vese approach was used for (3.3). Letting the divergence operator in (3.3) be denoted by  $A(\phi)$  using the definition in (2.41) the following is obtained.

$$\frac{\phi^{n+1} - \phi^n}{\tau} = H'(\phi^n) \left[ \mu A(\phi^n) \phi^{n+1} + \nu \frac{\mathbf{V}_0 - \int_{\Omega} H(\phi^n)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi^n)| + \beta} - \lambda |g - c_0|^2 \right] \quad (3.5)$$

where  $\phi^n = \phi(n\tau, \mathbf{x})$ . Using some mathematical manipulations this last equation can be written as

$$[\mathbf{I} - \mu \tau H'(\phi^n) A(\phi^n)] \phi^{n+1} = \phi^n + \tau H'(\phi^n) \left[ \nu \frac{\mathbf{V}_0 - \int_{\Omega} H(\phi^n)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi^n)| + \beta} - \lambda |g - c_0|^2 \right]. \quad (3.6)$$

### Segmentation Algorithm

To solve equation (3.6) the following algorithm is used.

1. Set  $\phi^0 = \phi_0$ .
2. Solve equation (3.6).
3. Check if solution is stationary using criteria described in the next section. If not  $n = n + 1$  and repeat the last step.

### Termination Criteria for Segmentation

This section will provide an overview of the termination conditions used in this project for the segmentation step.

Generally, given a function  $F$ , the problem is to find  $x^k$  that is a good approximation of the minimizer  $x^*$  [34]. In order to find a good approximation of  $x^*$ , one may want a desired accuracy in  $x^k$  and in  $F^k = F(x^k)$ . If a problem is well-behaved there is an established relationship between desired accuracy in  $x^k$  and the desired accuracy in  $F^k$ , as shown in Section 8.2.2.1 of [34], however if the problem is ill-conditioned it might happen that while  $F^k$  is a good approximation to  $F(x^*)$ ,  $x^k$  is not a good approximation of  $x^*$ . For this reason a user-specified parameter  $\xi$  can be used to enforce accuracy on both  $x^k$  and  $F^k$ .

Assuming that the desired accuracy of the solution  $F(x^*)$  to  $F^k$  has to be up to  $p$  digits then by setting  $\xi = 10^{-p}$  a measure of absolute accuracy can be defined as

$$\theta^k = \xi(1 + |F^k|) \quad (3.7)$$

A sensible set of termination criteria are the following:

- C1.**  $|F^{k-1} - F^k| < \theta^k$ .
- C2.**  $\|x^{k-1} - x^k\|_2 < \sqrt{\xi}(1 + \|x^k\|_2)$ .
- C3.**  $\|\gamma^k\|_\infty \leq \sqrt[3]{\xi}(1 + |F^k|)$ , where  $\gamma^k$  is the gradient of the function  $F$  at point  $x^k$ .

Conditions **C1** and **C2** are designed to test whether the sequence  $\{x^k\}$  is converging, while condition **C3** tests the for necessary optimality condition  $\|\gamma(x^*)\| = 0$ . For well-scaled problems, the satisfaction of condition **C1** automatically implies the satisfaction of condition **C2**, however, for ill-conditioned problems condition **C2** forces the algorithm to find a better solution.

As stated in [34], the choice for the usage of  $\sqrt{\xi}$  in condition **C2** is due to the fact that using  $\xi$  might be an overly stringent condition. Furthermore, in condition **C3** the  $L_2$  norm can be used, but when it comes to a very large number of variables it is better to use the infinity norm due to the fact that it is a less stringent condition provided that condition **C1** and **C2** should be satisfied at the same time.

Conditions **C1** to **C3** are unsatisfactory if the initial point  $x^0$  is in such a close neighbourhood of the solution and that no further progress can be made, or if an iterate  $x^k$  lands by chance very close to the solution. In this case, an additional condition on the gradient must be satisfied.

- C4.**  $\|\gamma^k\|_\infty < \epsilon_A$

where  $\epsilon_A$  is a very small number greater than the machine epsilon  $\epsilon_M$ .

Finally, a condition that guarantees termination in a finite number of iterations must be enforced.

- C5.**  $k < K$ , where  $k$  is the current iteration number and  $K$  is the maximum number of iterations allowed.

Although this condition may be satisfied before the algorithm has found a reasonable solution, it is important to have it, since for all practical purposes the algorithm should terminate in a finite amount of time.

In order to terminate the algorithm the following logical condition must be satisfied.

$$(\mathbf{C1} \text{ and } \mathbf{C2} \text{ and } \mathbf{C3}) \text{ or } \mathbf{C4} \text{ or } \mathbf{C5} \quad (3.8)$$

For segmentation step in the project of this thesis the aforementioned parameters are set as  $\xi = 10^{-7}$ ,  $\epsilon_A = 10^{-8}$  and  $K = 50$ . The functional (3.1) is discretized as follows

$$\begin{aligned} E(\phi^k) &= \mu \sum_{\Omega^h} h_1 \cdot h_2 \cdot h_3 \cdot H'(\phi^k) \sqrt{\mathcal{A}_1(\partial_1^+ \phi^k)^2 + \mathcal{A}_2(\partial_2^+ \phi^k)^2 + \mathcal{A}_3(\partial_3^+ \phi^k)^2} \\ &+ \nu |\mathbf{V}_0 - \sum_{\Omega^h} h_1 \cdot h_2 \cdot h_3 \cdot H(\phi^k)| \\ &+ \lambda_1 \sum_{\Omega^h} h_1 \cdot h_2 \cdot h_3 \cdot \|g - c_0\| H(\phi^k) \end{aligned} \quad (3.9)$$

where  $H$  is the Heaviside function,  $\Omega^h$  is the discretized domain  $\Omega$  and  $h_i$  is the discretization which in this case is 1.84mm/voxel-edge in each direction considering the resolution  $32 \times 32 \times 32$  voxels and metric space  $(58.88 \times 58.88 \times 58.88)\text{mm}^3$ . In (3.9),  $\mathcal{A}_i$  is an averaging operator acting along direction  $i$  that brings the staggered difference in direction  $i$  back to the centre of the cell defined as

$$\begin{aligned} \mathcal{A}_1 &= \mathbf{I} \otimes \mathbf{I} \otimes \bar{\mathcal{A}} \in \mathbb{R}^{kn(l+1), knl} \\ \mathcal{A}_2 &= \mathbf{I} \otimes \bar{\mathcal{A}} \otimes \mathbf{I} \in \mathbb{R}^{k(n+1)l, knl} \\ \mathcal{A}_3 &= \bar{\mathcal{A}} \otimes \mathbf{I} \otimes \mathbf{I} \in \mathbb{R}^{(k+1)nl, knl} \end{aligned} \quad (3.10)$$

where  $k$ ,  $n$  and  $l$  represent the size in cells of the data in each direction. Here,  $E(\phi^k)$  substitutes  $F^k$  in conditions **C1** and **C3**.

Condition **C2** can be checked by substituting the discretized  $\phi^k$  for  $x^k$  as

$$\|\phi^{k-1} - \phi^k\|_2 < \sqrt{\xi}(1 + \|\phi^k\|_2)$$

Condition **C3** and **C4** are both based on the gradient  $\gamma^k$ . These conditions can be checked using the right hand side of equation (3.5) therefore,

$$\gamma^k = H'(\phi^k) \left[ \mu A(\phi^k) \phi^k + \nu \frac{\mathbf{V}_0 - \int_{\Omega} H(\phi^k)}{|\mathbf{V}_0 - \int_{\Omega} H(\phi^k)| + \beta} - \lambda |g - c_0|^2 \right]. \quad (3.11)$$

In summary, using

$$\theta^k = \xi(1 + |E(\phi^k)|) \quad (3.12)$$

the termination conditions for segmentation can be checked by

$$\mathbf{C1.} \quad |E(\phi^{k-1}) - E(\phi^k)| < \theta^k.$$

$$\mathbf{C2.} \quad \|\phi^{k-1} - \phi^k\|_2 < \sqrt{\xi}(1 + \|\phi^k\|_2).$$

$$\mathbf{C3.} \quad \|\gamma^k\|_{\infty} \leq \sqrt[3]{\xi}(1 + |E(\phi^k)|)$$

$$\mathbf{C4.} \quad \|\gamma^k\|_{\infty} < \epsilon_A$$

with  $\gamma^k$  as defined in (3.11).

### 3.1.3 Alternative Objective Functional Trials for Segmentation

The objective of this section is to provide an idea of the reasoning used to construct of the objective functionals for lung segmentation.

As mentioned before, the functional (3.1) makes use of information from a property of the lungs exhibited in the CT data as well as a property of the subject itself. However, attempts were made to make use of only the average intensity value of the lungs in the CT data.

As mentioned in Section 3.1.1, there is a large change in values, from an average of  $-300HU$  in the inner part of the lung to  $\geq 200HU$  on the ribs. This leads to the conclusion that there is a high gradient on the border of the lungs and ribs. This means that one may make use of  $\nabla g$  in order to attract the contour, therefore

$$E(\phi) = \mu \int_{\Omega} H'(\phi)|\nabla\phi| + \nu \int_{\Omega} H(\phi) - \lambda_1 \int_{\Omega} |\nabla g|^2 H'(\phi) + \lambda_2 \int_{\Omega} |g - c_0|^2 H(\phi)$$

The first two terms in this functional try to minimize area of and volume inside  $\{\phi = 0\}$  respectively. The third term aims to attract the contour towards high gradients in the image. The last term aims to minimize the difference between the image  $g$  and  $c_0$  in  $\{\phi > 0\}$ . This is not possible to implement because the third term produces

$$\int_{\Omega} |\nabla g|^2 H''(\phi)\psi$$

when the first variation is taken. The reason this cannot be used is because, unlike the regularized  $H'$  which resembles the properties of  $H$  when integrated,  $H''$  does not produce an  $H'$  that resembles the properties of the Dirac delta function when integrated.

A modification of the above functional is the following

$$E(\phi) = \mu \int_{\Omega} H'(\phi)|\nabla\phi| + \nu \int_{\Omega} H(\phi) - \lambda_1 \int_{\Omega} |\nabla g|^2 H(\phi) + \lambda_2 \int_{\Omega} |g - c_0|^2 H(\phi)$$

While this is very similar to the last one, it tries to maximize the gradient in  $\{\phi > 0\}$  which means that the curve  $\{\phi = 0\}$  will try to include into the foreground all areas with high gradient. This was done with the intention of putting a high emphasis on the length penalty term, by choosing a large  $\mu$ . This method was deemed ineffective due to the fact that it does not work for all cases as shown in Figures 3.1(a)–3.1(c).

Another choice based on the last functional is the assumption that the curve resulting from the segmentation should be close to the initial curve. This condition is based on the assumption of proximity of the initial guess to the lungs.

$$\begin{aligned} E(\phi) &= \mu \int_{\Omega} H'(\phi)|\nabla\phi| + \nu \int_{\Omega} (H(\phi) - H(\phi_0))^2 \\ &\quad - \lambda_1 \int_{\Omega} |\nabla g|^2 H(\phi) + \lambda_2 \int_{\Omega} |g - c_0|^2 H(\phi) \end{aligned}$$

Here,  $\phi_0$  is the initial level-set function. This functional, however, has the same drawbacks as the one before and was not effective in achieving the desired result.

Another attempt was to extend the existing Chan-Vese approach using the fact that the initial guess is in close proximity to the expected result. This leads to the construction of the following functional.

$$\begin{aligned} E(\phi) &= \mu \int_{\Omega} H'(\phi)|\nabla\phi| + \nu \int_{\Omega} H(\phi) + \lambda_1 \int_{\Omega} |g - c_1(\phi)|^2 H(\phi) \\ &\quad + \lambda_2 \int_{\Omega} |g - c_2(\phi)|^2 (1 - H(\phi)) + \lambda_3 \int_{\Omega} |g - c_0|^2 H(\phi) \end{aligned}$$

This functional makes use of the constant  $c_0$  as a user input for the average value of the lungs, but also of  $c_1(\phi)$  and  $c_2(\phi)$  as defined in (2.27) and (2.28) respectively. The

reasoning is to use the user's knowledge about the lungs of the subject as well as to use the underlying data in order to refine the region of interest. This was not very successful due to the fact the result was not the one desired for cases such as that as shown in Figure 3.1(a), where regions exhibiting values of  $-300HU$  outside the lungs were also segmented.

A slightly modified version of the last functional which accounts for the fact that intensity values in the lungs may vary between  $-500HU$  and  $-100HU$  was also tried, however, this did not prove to be successful for all the three cases shown in Figures 3.1(a)–3.1(c).

$$\begin{aligned}
 E(\phi) &= \mu \int_{\Omega} H'(\phi) |\nabla \phi| + \nu \int_{\Omega} H(\phi) + \lambda_1 \int_{\Omega} |g - c_1(\phi)|^2 H(\phi) \\
 &+ \lambda_2 \int_{\Omega} |g - c_2(\phi)|^2 (1 - H(\phi)) + \lambda_3 \int_{\Omega} |g - c_a|^2 H(\phi) \\
 &+ \lambda_3 \int_{\Omega} |g - c_b|^2 H(\phi)
 \end{aligned}$$

where  $c_a = -500$  and  $c_b = -100$ . The first two terms minimize the area of and volume inside  $\{\phi = 0\}$ . Terms 3 and 4 make use of the values in the underlying data. The last two terms try to force the curve  $\{\phi = 0\}$  to include regions in the data with intensity values between the user-provided values  $c_a$  and  $c_b$ .

Alternatively, one may want to include values  $-300HU$  and exclude values such as  $-1000HU$  (the intensity of air).

$$\begin{aligned}
 E(\phi) &= \mu \int_{\Omega} H'(\phi) |\nabla \phi| + \nu \int_{\Omega} H(\phi) + \lambda_1 \int_{\Omega} |g - c_1(\phi)|^2 H(\phi) \\
 &+ \lambda_2 \int_{\Omega} |g - c_2(\phi)|^2 (1 - H(\phi)) + \lambda_3 \int_{\Omega} |g - c_a|^2 H(\phi) \\
 &+ \lambda_3 \int_{\Omega} |g - c_b|^2 (1 - H(\phi))
 \end{aligned}$$

This functional tries to include values close to  $c_a = -300HU$  in  $\{\phi > 0\}$  and values close to  $c_b = -1000HU$  in  $\{\phi < 0\}$ . Approximate results to using the Chan-Vese functional (2.27) are obtained.

Each of these functionals took an amount of time to be tested since the parameters need to be fine-tuned in order to ensure whether the model can produce the desired result.

## 3.2 Objective Functions for Ellipsoid Fitting and Registration

In this section the objective function used for ellipsoid fitting to find the initial and final guess of the region of interest (ROI) will be described in detail. Following, will be a description of how the ROI-weighted  $P_{CT}$  data is registered/aligned to the  $V_{CT}$  data.

### 3.2.1 Ellipsoid Fitting

As mentioned in the Introduction the first step to obtaining an initial guess for the ROI is to register an ellipse to a windowed version of the  $P_{SP}$  data. The reason for this is that it is assumed that the  $P_{SP}$  and  $P_{CT}$  data are pre-aligned and therefore the regions of the lungs appearing in the  $P_{SP}$  data are indicative of the location of the lungs in the  $P_{CT}$  data.

Before going into the details of the objective function, the function used to describe the ellipsoid will be discussed. The function

$$W(\omega, x) = \begin{cases} 1 & R < 0.5 \\ -\frac{\cos(2\pi R)}{2} + \frac{1}{2} & 0.5 \leq R \leq 1 \\ 0 & R > 1 \end{cases} \quad (3.13)$$

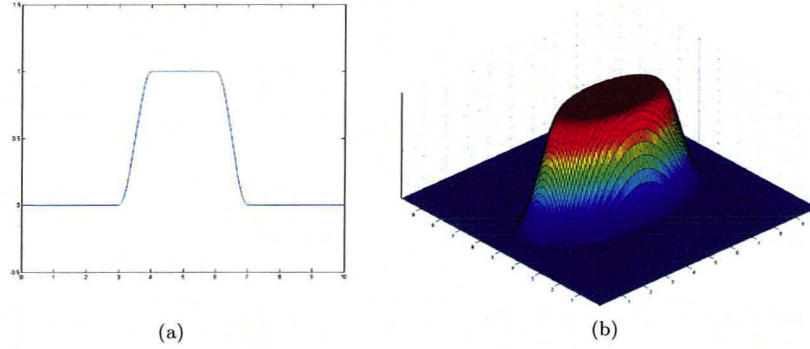
defines a smooth ellipsoid with  $0 \leq W(\omega, x) \leq 1$  where  $\omega = [\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6]^T$  with the first three parameters as the radii of the ellipsoid in direction  $i$  and the last three parameters as the spatial position of the ellipsoid along direction  $i$ . Furthermore,

$$R = \sqrt{\hat{r}_1^2 + \hat{r}_2^2 + \hat{r}_3^2} \quad (3.14)$$

with

$$\begin{aligned} \hat{r}_1 &= \frac{x_1 - \omega_4}{\omega_1} \\ \hat{r}_2 &= \frac{x_2 - \omega_5}{\omega_2} \\ \hat{r}_3 &= \frac{x_3 - \omega_6}{\omega_3} \end{aligned} \quad (3.15)$$

Figure 3.2 provides a visualization of the structure of the ellipsoid in one and two dimensions.



**Figure 3.2:** Visualization of the ellipsoid function  $W$  in (a) one and (b) two dimensions.

As mentioned in section 2.2, the FAIR Framework uses the Gauss-Newton method with Armijo line-search therefore it is necessary to compute the gradient of (3.13) with respect to  $\omega$ .

$$\partial_{\omega} W(\omega, x) = \begin{cases} \begin{bmatrix} -\frac{\pi \sin(2\pi R) \hat{r}_1^2}{\omega_1 R} \\ -\frac{\pi \sin(2\pi R) \hat{r}_2^2}{\omega_2 R} \\ -\frac{\pi \sin(2\pi R) \hat{r}_3^2}{\omega_3 R} \\ -\frac{\pi \sin(2\pi R) \hat{r}_1}{\omega_1 R} \\ -\frac{\pi \sin(2\pi R) \hat{r}_2}{\omega_2 R} \\ -\frac{\pi \sin(2\pi R) \hat{r}_3}{\omega_3 R} \end{bmatrix} & 0.5 \leq R \leq 1 \\ \vec{0} & \text{otherwise} \end{cases} \quad (3.16)$$

The objective function for the registration of the ellipse to some data  $u$  is defined as follows

$$F(\omega) = \|\omega_1^2 + \omega_2^2 + \omega_3^2\|_2 + \alpha \int_{\Omega} u(x)(1 - W(\omega, x)) dx \quad (3.17)$$



where  $\alpha > 0$ . The first term aims to minimize the three radii of the ellipsoid while the second term aims to exclude as much data as possible from the reverse ellipsoid therefore including as much data as possible inside the ellipsoid. The gradient of function (3.17) is

$$\nabla F(\omega) = \frac{2}{\|\omega_1^2 + \omega_2^2 + \omega_3^2\|_2} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ 0 \\ 0 \\ 0 \end{bmatrix} - \int_{\Omega} \alpha u(x) \nabla W(\omega, x) dx. \quad (3.18)$$

The Hessian matrix has been approximated by

$$\mathcal{H}F(\omega) = \frac{\pi}{\|\omega_1^2 + \omega_2^2 + \omega_3^2\|_2} \mathbf{I} \quad (3.19)$$

where  $\mathbf{I}$  is the identity matrix.

Equations (3.17), (3.18) and (3.19) are then used in Gauss-Newton method of the FAIR framework in order to find the optimal size and position of the ellipsoid that includes the data  $u$ .

In the project treated in this thesis the ellipsoid fitting is performed in two steps of the alignment process. First to find the initial guess  $\phi_0$  on a windowed version of the  $P_{SP}$  data, therefore  $u$  in (3.17) becomes

$$u = \text{windowed } P_{SP} = \begin{cases} 1 & \text{if } P_{SP} > 300 \\ 0 & \text{if } P_{SP} \leq 300 \end{cases}. \quad (3.20)$$

then to find the refined region of interest using the segmented  $P_{CT}$  data, therefore

$$u = \text{segmented } P_{CT} = \begin{cases} 1 & \text{in } \{\phi_{final} \geq 0\} \\ 0 & \text{in } \{\phi_{final} < 0\} \end{cases} \quad (3.21)$$

In this section the mathematical description of the smooth ellipsoid as well as the objective functions used to optimize its size and position were presented. Further details to the Gauss-Newton method with Armijo line-search used in the FAIR Framework can be found in [22]. In the next section the region-of-interest based registration is discussed.

### 3.2.2 Region-of-Interest Based Registration

In this section the objective function used to register the template data  $\mathcal{T}$ , or  $P_{CT}$ , and the reference data  $\mathcal{R}$ , or  $V_{CT}$ , will be discussed. At this stage only rigid transformations are considered to align the data. In order to register the data based on the region-of-interest, the final ellipsoid resulting from step 3 of the process shown in Figure 1.7 is used as a “weight” or spatial filter on the template data  $\mathcal{T}$ . The ROI-weighted template data is obtained as

$$\mathcal{T}^W(x) = \mathcal{T}(x)W(\omega_{opt}, x) \quad (3.22)$$

where  $\omega_{opt}$  is the set of parameters of the refined ROI.

The objective functional used in the following

$$\mathcal{J}[\nu] = \mathcal{D}[\mathcal{T}^W[y(\nu)], \mathcal{R}] = \frac{1}{2} \int_{\Omega} (\mathcal{T}^W(y(\nu, x)) - \mathcal{R}(x))^2 dx \quad (3.23)$$

where  $y$  is the transformation and  $\nu = [\nu_1, \nu_2, \nu_3, \nu_1, \nu_2, \nu_3]^T$  are the parameters of the rigid transformation, with the first three being the angle of rotation around axis  $i$  and the last three being the translation along axis  $i \in \{1, 2, 3\}$ .

By discretizing (3.23) the following objective function is obtained

$$J(\nu) = \frac{1}{2} \sum_{\Omega^h} h_1 \cdot h_2 \cdot h_3 \|\mathcal{T}_h^W(y(\nu)) - \mathcal{R}_h\|^2 \quad (3.24)$$

where  $T_h^W$  and  $\mathcal{R}_h$  represent the cell-centered discretization of  $T^W$  and  $\mathcal{R}$  respectively. The  $h_i$  is the metric size of the cell in direction  $i \in \{1, 2, 3\}$  and  $\Omega^h$  represents the cells of the discretized domain  $\Omega$ .

In order to use (3.24) in the FAIR framework the gradient and Hessian are computed as follows

$$\nabla J(y(\omega)) = h_1 h_2 h_3 \|T_h^W(y(\nu)) - \mathcal{R}_h\| dT^W(y) dy(\nu) \quad (3.25)$$

$$\mathcal{H}J(y(\nu)) = h_1 h_2 h_3 (dT^W(y) dy(\nu))^T (dT^W(y) dy(\nu)) \quad (3.26)$$

where  $dT^W(x) = dT(x) \hat{W}(\omega_{opt}, x)$  with  $\hat{W}(\omega_{opt}, x)$  being a diagonal matrix with values  $W(\omega_{opt}, x)$  in the main diagonal and

$$dT = \begin{pmatrix} \partial_1 T(y_1) & & \partial_2 T(y_1) & & \partial_3 T(y_1) & & \\ & \ddots & & \ddots & & \ddots & \\ & & \partial_1 T(y_n) & & \partial_2 T(y_n) & & \partial_3 T(y_n) \end{pmatrix}$$

where  $dT \in \mathbb{R}^{(n, dn)}$  with  $d$  being the number of dimensions,  $\partial_i$  the difference operator along dimension  $i$  and  $T(y_k)$  the value of  $T$  at cell  $y_k$  and finally

$$dy(\nu) = \begin{pmatrix} \partial_{\theta_1} y_1 & \partial_{\theta_2} y_1 & \partial_{\theta_3} y_1 & \partial_{d_1} y_1 & \partial_{d_2} y_1 & \partial_{d_3} y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \partial_{\theta_1} y_n & \partial_{\theta_2} y_n & \partial_{\theta_3} y_n & \partial_{d_1} y_n & \partial_{d_2} y_n & \partial_{d_3} y_n \end{pmatrix}$$

The equations in (3.24), (3.25) and (3.26) can now be used with the Gauss-Newton method of the FAIR Framework in order to find the parameters  $\nu$  of the rigid transformation  $y$ .

So far, the project-specific methods of segmentation and registration were described. In the next chapter, some experimental results including both synthetic and practical examples will be presented. The synthetic examples concern only the segmentation piece in order to provide some idea of the ‘‘preciseness’’ of the segmentation step while the real-life examples, provided by Dr. Troy Farncombe and the McMaster Centre for Preclinical Imaging, test the overall preciseness of the alignment process.

## Chapter 4

# Testing the Segmentation and Alignment Processes

In this chapter, the experiments carried out for the segmentation and alignment process will be discussed. Section 4.1 will show some typical experiments on synthetic data that was designed to test whether the model can extract the desired object from a  $2D$  image. In this same section, the tests carried out on  $2D$  slices of the data provided by the McMaster Centre for Preclinical Imaging for some typical initial guesses will also be shown. In section 4.2 the testing carried out on the whole alignment process will be shown along with the results of the alignment for the six weeks of data provided.

### 4.1 Segmentation Experiments

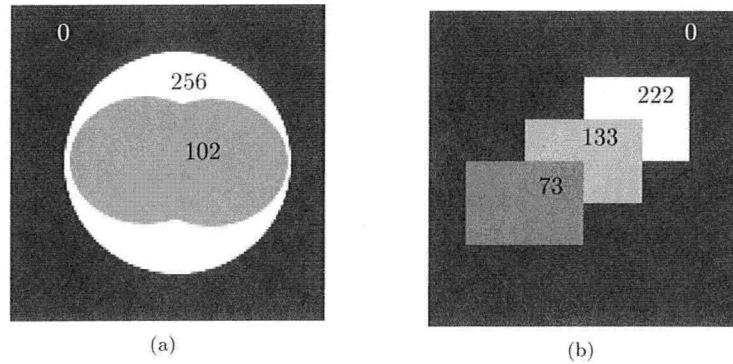
Segmentation experiments were carried out on 2D and 3D samples of synthetic and real-life data. However in this section, for ease of visualization, only experiments carried out on 2D data will be shown. The experiments in this section include the segmentation of

- synthetic data mimicking a cross-sectional slice of the chest of the mouse,
- synthetic data to observe the inclusion of regions with approximate value to the region of interest and
- a coronal slice of a mouse.

Two typical images constructed for the purpose of testing segmentation are the images shown in Figure 4.1. Figure 4.1(a) shows two circles in the middle surrounded by a circle with high intensity values which in turn is surrounded by low intensity values. This is done to simulate the lungs in the body with surrounding air in a cross-sectional slice. Figure 4.1(b) shows three squares with three different intensity values. This image was created to observe how the curve evolves according to (3.1) when there are objects in the vicinity of the object of interest that have similar intensity values. The resolution of the images is  $128 \times 128$  pixels and the metric space  $\Omega$  is  $(10 \times 10)mm^2$ .

For the image in 4.1(a), the objective is to extract the area in the centre of the image with value 102 therefore  $c_0 = 102$ . The area of this piece is about one half of the domain  $\Omega$  therefore  $V_0 = 50mm^2$ . Figures 4.2(a)–4.2(h) show some initial guesses along with some typical results. The parameters were found empirically and set to  $\mu = 0.5 \cdot 255^2$ ,  $\lambda = 10$ ,  $\nu = 10^5$  with time-step  $\tau = 0.78$  for all of the initial guesses shown.

Figures 4.3(a)–4.3(h) show the segmentation results on the image in 4.1(a) with some added noise. The noise is normally distributed with mean 0 and standard deviation 100. The parameter  $\mu = 255^2$  was the only one changed in this case.



**Figure 4.1:** Images with resolution  $128 \times 128$  and metric size  $(10 \times 10)mm^2$  used to test the segmentation method. The numbers on the images represent the intensity values of each distinct region.

By visual inspection it seems that the segmentation is working fairly well for the test image in 4.1(a) with and without noise. For larger amounts of noise, this segmentation method does not work as well. A workaround is to regularize the image using tools in the FAIR Framework [22], such as increasing the scale-space parameter, in order to minimize the noise after which segmentation can be performed.

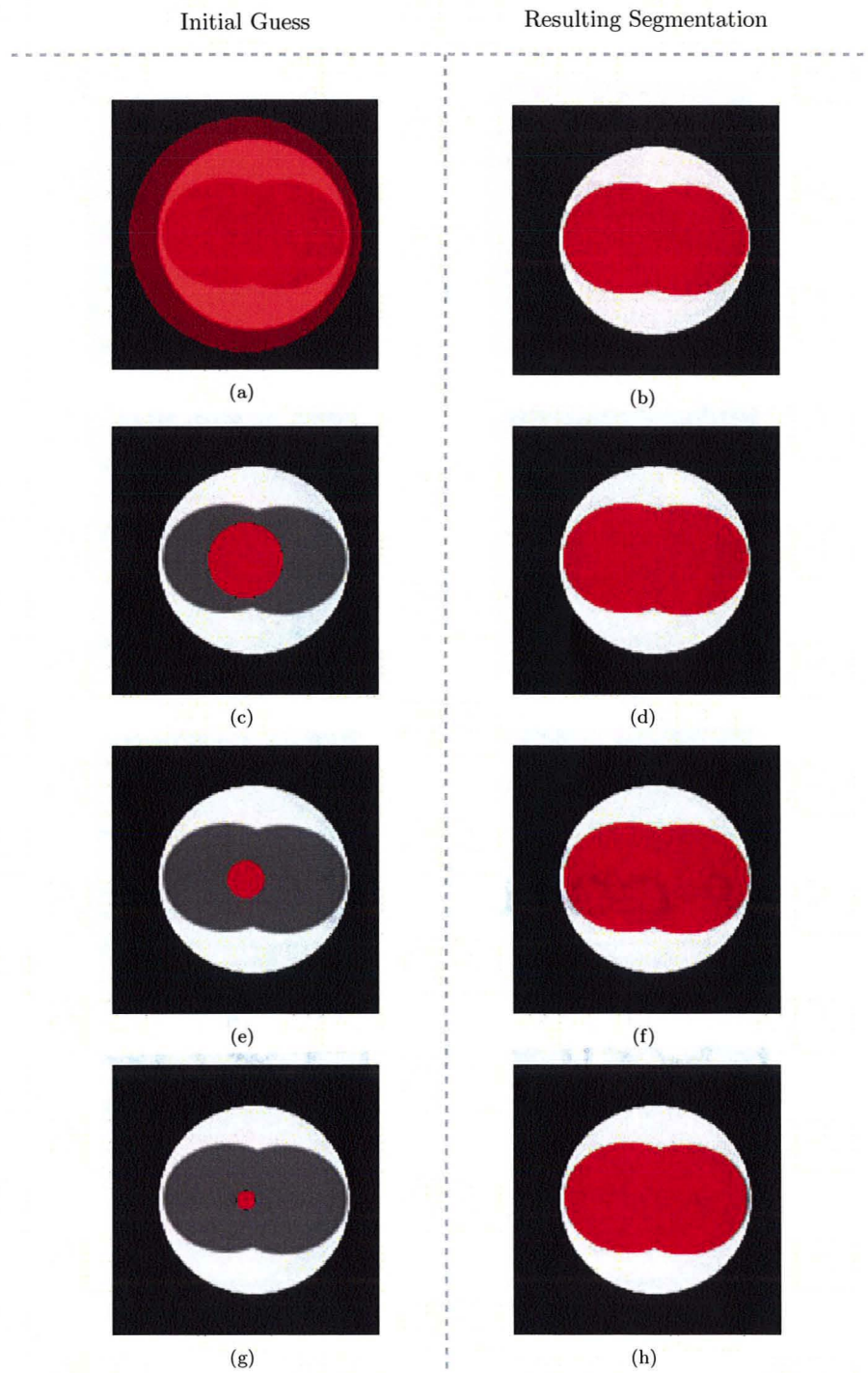
The next step is to test segmentation on Figure 4.1(b) in order to see what is segmented when regions close to the region of interest exhibit the same intensity values. Figures 4.4(a)–4.4(h) show some typical initial guesses and results. The area is set to  $V_0 = 31.25mm^2$  and  $c_0 = 133$ . The parameter  $\mu$  was switched back to  $0.5 \cdot 255^2$ .

As can be seen from Figures 4.4(a)–4.4(h) the object of interest is segmented, but due to the high emphasis on the penalty on the curvature a region not having the value  $c_0$  is included as well. This occurs in real-life data also, where the lungs are segmented along with most of the heart. For the real-life data used in the project this has not posed a problem due to the fact that the heart is expected to be included in the refined ellipsoidal ROI.

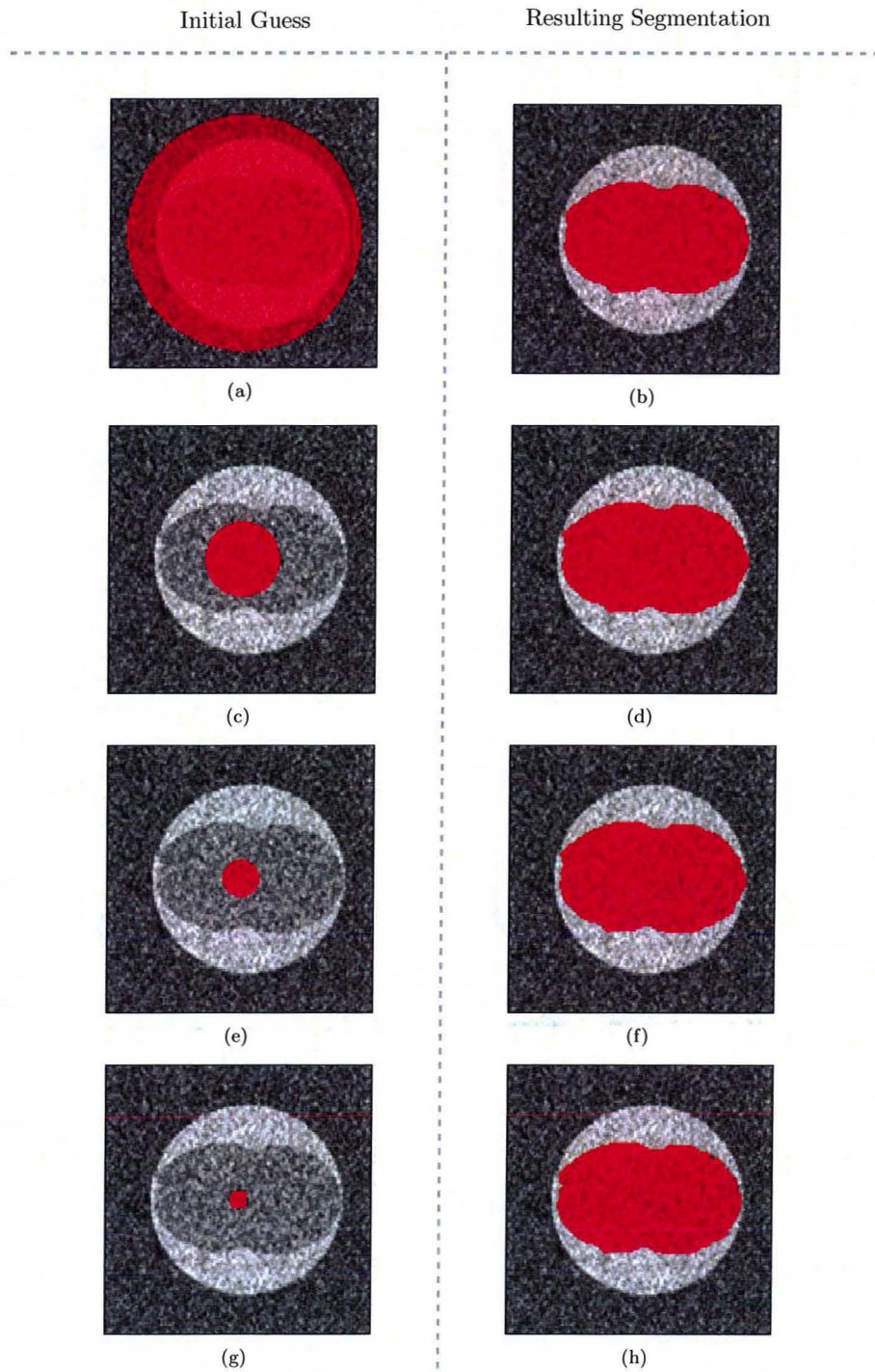
The synthetic experiments so far have shown promising results. The next step is to test the segmentation method in a 2D slice of the data provided by Dr. Troy Farncombe and the McMaster Centre for Preclinical Imaging. The data, that is initially  $256 \times 256 \times 256$  voxels, is first reduced to  $128 \times 128 \times 128$  voxels by averaging neighboring values in each direction, then it is further reduced to  $128 \times 128$  pixels by extracting the  $72^{nd}$  coronal slice. The reduction in resolution is done mainly to reduce computation time for experiments. The metric space of the image is  $(58.88 \times 58.88)mm^2$ . The experiments run on this particular slice are shown in Figures 4.5(a)–4.5(h). The parameters used for this experiment are  $\mu = 0.5 \cdot 255^2$ ,  $\nu = 10^5$ ,  $\lambda = 1$  with time-step  $\tau = 0.046$ . The average value of the lungs is set to  $c_0 = -300$  and the area of the lungs is set to  $V_0 = 71mm^2$ . The time-step was chosen small in order for the curve to expand more “conservatively” and not include other areas with average value around  $-300HU$  such as fat and excess skin in the arms of the mouse. If the time step is increased by a factor of 2 or more the segmented region flows outside the lungs, therefore to have the desired segmentation the time-step must be small.

The initial guesses shown in Figure 4.5 are chosen in such as way that they resemble the initial guesses produced by the first step of the alignment process. Typical cases include large, medium and small initial guesses and, as it seems from the results in 4.5, the refined region of interest includes the lungs satisfactorily.

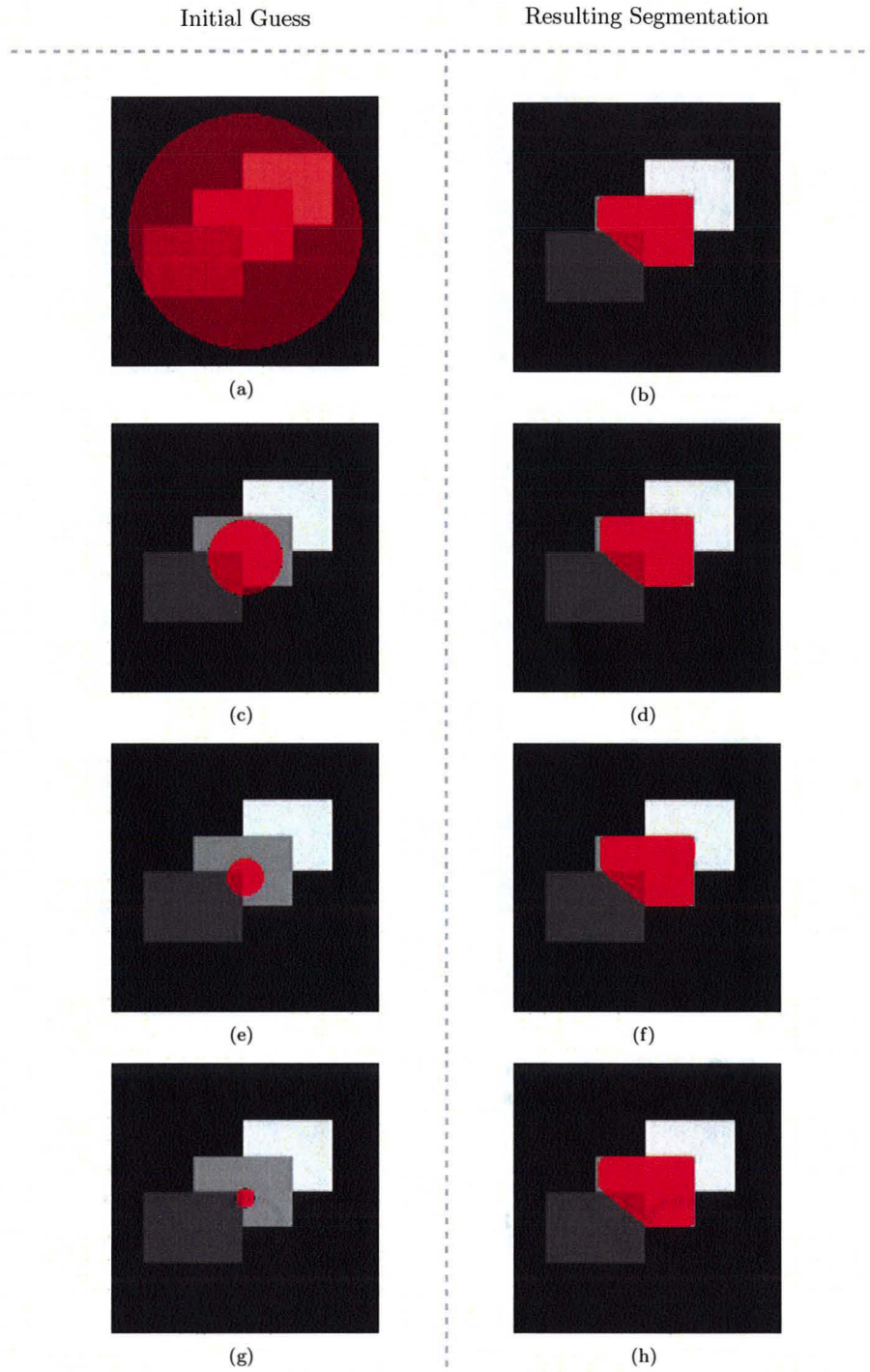
The important features of the parameters in the above experiments are the following.



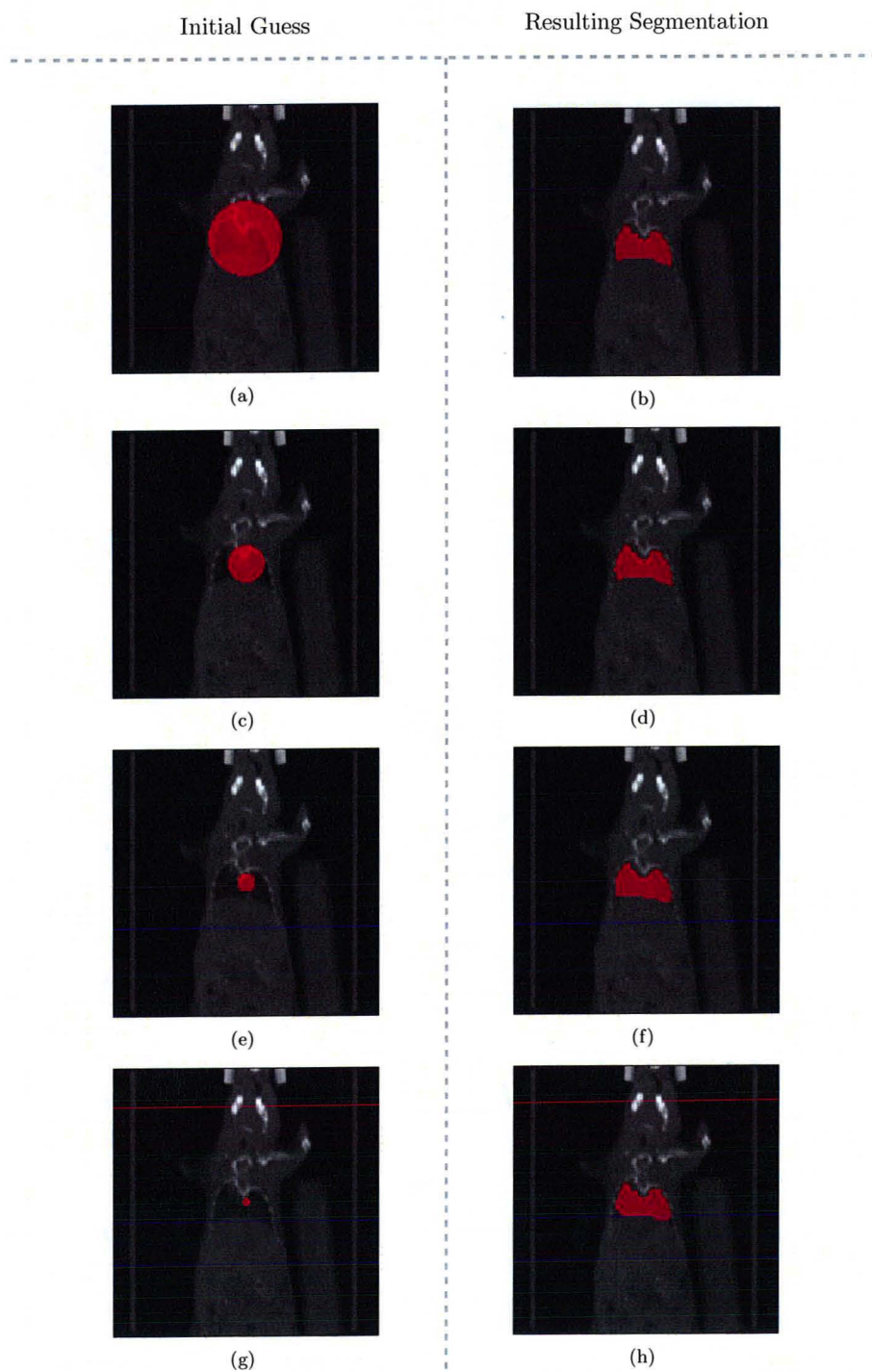
**Figure 4.2:** Typical initial guesses (left) and their respective results (right) for image 4.1(a). The area in red represents the segmented region  $\{\phi \geq 0\}$ . For all initial guesses on the left the segmented area approximates the region of interest.



**Figure 4.3:** Typical initial guesses (left) and their respective results (right) for image 4.1(a) with added normally distributed noise. The area in red represents the segmented region  $\{\phi \geq 0\}$ . Even with a certain amount of noise in the image, approximate results to those in Figure 4.2 are obtained.



**Figure 4.4:** Typical initial guesses (left) and their respective results (right) for image 4.1(b). The area in red represents the segmented region  $\{\phi \geq 0\}$ . For all initial guesses on the left the region of interest is included in the segmented area.



**Figure 4.5:** Typical initial guesses (left) and their respective results (right) for coronal slices of laboratory data. The area in red represents the segmented region  $\{\phi \geq 0\}$ . For all initial guesses on the left the segmented region approximates the lungs.



- The parameters  $\mu$  and  $\nu$  were not modified for any of the experiments therefore when it comes to segmenting data over a number of weeks since the user need not correct the parameters for each week. In fact, as will be seen in section 4.2, the parameters are kept the same for all the six weeks of data provided by the McMaster Centre for Preclinical Imaging.
- The time-step has to be small when it comes to real data due to the fact that many small regions in the body of the subject might exhibit the same average intensity of values. Some of these regions might be away from the expected ROI and may therefore be obstructive to the registration step. It might not be always possible to avoid including these regions, but a small time-step tends to keep the segmented region local to the initial guess. In addition, a large  $\mu$  also ensures exclusion of regions far from the expected ROI since it penalizes the length of the curve  $\Gamma = \{\phi = 0\}$ .

The quality of the segmentation experiments at this stage in the project has been based on visual inspection. Defining what is a good segmentation result has shown to be quite challenging and is part of ongoing and future research.

While the segmentation step alone is hard to benchmark, the overall performance and precision of the alignment process can be tested numerically as will be seen in the following section.

## 4.2 Testing the Alignment Process

In this section the results of the overall alignment process will be described. The experiments discussed here were carried out on 3D data. The alignment process was tested using a number of simulated rigid transformations.

In order to get an idea of how much rotation and displacement there is in real-life between ventilation and perfusion data the alignment process was run on all six weeks of data provided. This was done to get an idea of what to consider a “valid” transformation so that the alignment process would not be discarded as ineffective due to some unrealistic displacement or rotation.

The alignment for the six weeks of data was done using the following parameters.

- The  $P_{SP}$  data is windowed at 300.
- Ellipsoid fitting to windowed  $P_{SP}$  data (to obtain the initial guess) parameter  $\alpha$  in (3.13) set to  $\alpha = 1$ .
- Segmentation parameters in (3.6) set to  $\mu = 0.5 \cdot 255^2$ ,  $\nu = 10^5$ ,  $\lambda = 1$  and  $\tau = 0.046$ .
- Ellipsoid registration to ROI of  $P_{CT}$  data parameter set to  $\alpha = 1$  in (3.13).
- The objective function (3.24) used for the registration of the ROI of the  $P_{CT}$  data to the  $V_{CT}$  data is not parameterized.
- The resolution at which multi-level registration is performed is  $2^i \times 2^i \times 2^i$  voxels, for  $i = \{3, 4, 5, 6\}$ .
- The resolution at which the ellipsoid is registered is  $32 \times 32 \times 32$  voxels.
- The resolution at which segmentation is performed is  $16 \times 16 \times 16$  voxels.
- The metric space is unchanged at  $(58.88 \times 58.88 \times 58.88)mm^3$  for all resolutions.

The resulting transformations are shown in Table 4.1. These transformations were tested visually for correctness by applying the resulting transformation to the

Week	Transformation Parameters		Completion Time (s)
	$\theta$ (rad)	$\mathbf{d}$ (mm)	
1	-0.0114	0.4806	$1.2107 \cdot 10^3$
	0.0005	-0.1256	
	-0.0041	-0.2083	
2	-0.0049	0.3866	$1.0754 \cdot 10^3$
	-0.0008	-0.0651	
	-0.0016	-0.1206	
3	-0.0095	0.4089	$1.4651 \cdot 10^3$
	0.0008	0.0317	
	-0.0067	-0.1200	
4	-0.0102	0.2921	$1.1299 \cdot 10^3$
	0.0004	-0.2967	
	-0.0008	-0.3034	
5	0.0002	-0.0552	$1.0866 \cdot 10^3$
	-0.0016	-0.1344	
	0.0015	-0.0150	
6	0.0046	0.2886	$1.3306 \cdot 10^3$
	-0.0022	0.0800	
	0.0023	-0.0237	

**Table 4.1:** Rigid transformation parameters found by applying the alignment process to the six weeks of data provided by the McMaster Centre for Preclinical Imaging;  $\theta$  represents the angle of rotation and  $\mathbf{d}$  the displacement.

template data  $\mathcal{T}$  then overlapping it with the reference data  $\mathcal{R}$  to check for differences. Table 4.1 provides a list of the transformations found.

The simulated rigid transformations applied on the six weeks of  $P_{CT}$  data were chosen about ten times larger than those shown in Table 4.1. The alignment process uses the same set of parameters mentioned above. The experiment setting is the following:

- 10 rigid transformations were applied to six weeks of perfusion data of a mouse to simulate the misaligned ventilation data
- The transformations were chosen randomly using a normal distribution
  - with mean 0 for rotation and translation
  - and standard deviations  $\sigma_{rotation} = 0.3\text{rad}$  and  $\sigma_{translation} = 7\text{mm}$ ;
  - the transformations range approximately between  $\theta_i \in [-0.3, 0.3]\text{rad}$  and  $d_i \in [-16, 16]\text{mm}$ .
- The magnitude of the absolute error was analyzed
  - based on a fixed ROI and ten transformations
  - and on a fixed transformation over six different ROI.

Note that this type of test does not test the accuracy of the alignment process over perfusion and ventilation data as in reality, however, the results obtained using the warped samples provide an idea of the accuracy of the alignment process developed in this thesis.

The magnitude of the absolute error in rotation and translation of the resulting transformations is plotted in Figure 4.6 and 4.7. The magnitude of the average and maximum absolute errors by fixed ROI over 10 transformations were

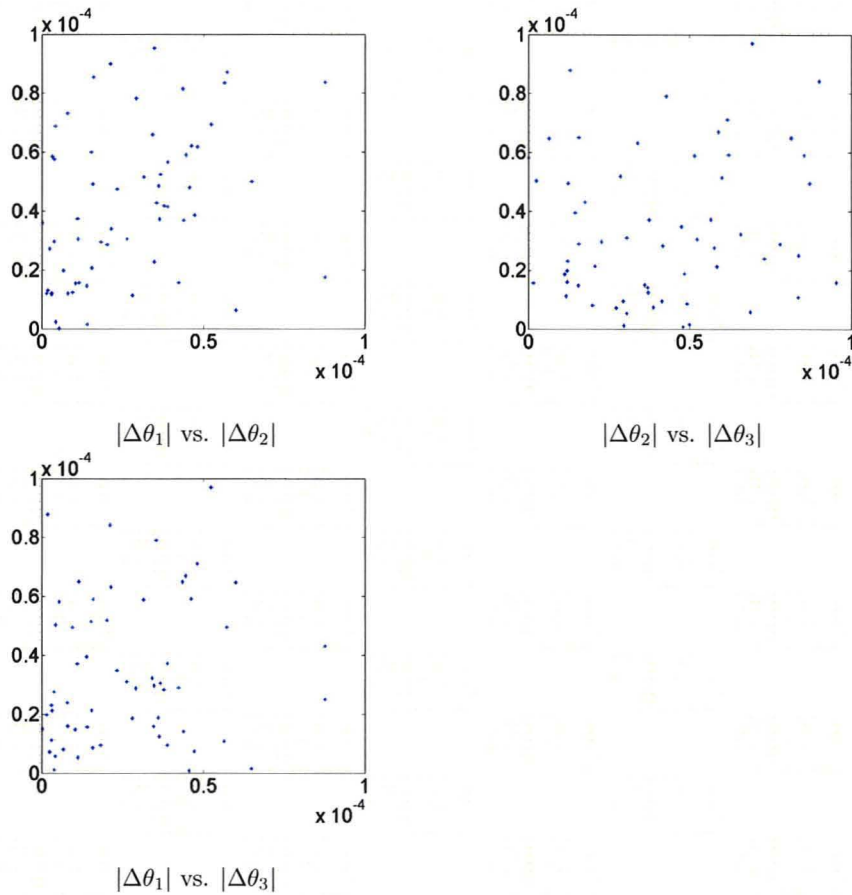
$$\begin{cases} E_{rotation} = [3.9724, 5.8880, 9.0892] \times 10^{-5}\text{rad} \\ E_{translation} = [1.1853, 1.9847, 3.3910] \times 10^{-3}\text{mm} \end{cases}$$

$$\begin{cases} E_{\text{rotation}} = [0.5349, 0.8427, 1.0261] \times 10^{-4} \text{rad} \\ E_{\text{translation}} = [1.5536, 2.4020, 3.1456] \times 10^{-3} \text{mm} \end{cases}$$

The magnitude of the average and maximum absolute errors by fixed transformation over 6 ROI were

$$\begin{cases} E_{\text{rotation}} = [5.6360, 8.0771, 7.9695] \times 10^{-5} \text{rad} \\ E_{\text{translation}} = [2.6128, 2.1828, 3.8097] \times 10^{-3} \text{mm} \end{cases}$$

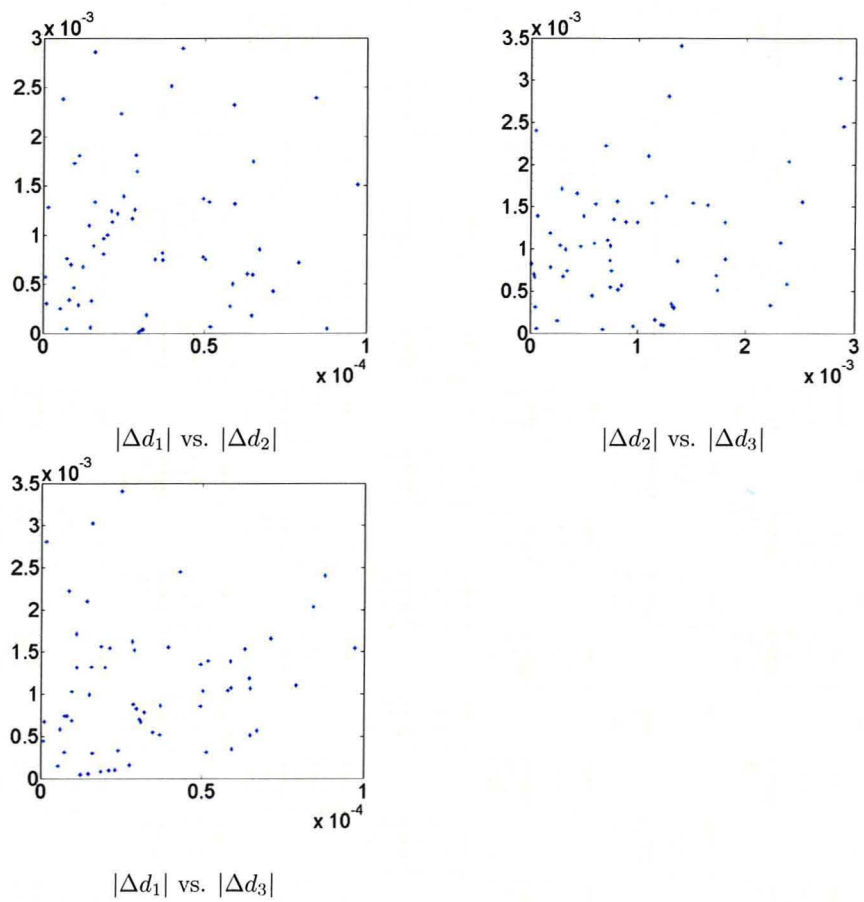
$$\begin{cases} E_{\text{rotation}} = [1.2403, 1.3783, 1.2517] \times 10^{-4} \text{rad} \\ E_{\text{translation}} = [4.5065, 3.6320, 5.4822] \times 10^{-3} \text{mm} \end{cases}$$



**Figure 4.6:** Magnitude of the absolute error of rotation (radians) for the transformations applied.

For all simulated rigid transformations the rotation parameters are off by one hundredth of a degree and the translation parameters are off by at most a hundredth of a millimeter. This result is promising when considering that registration is performed at most at  $64 \times 64 \times 64$  voxels whereas the initial resolution is  $256 \times 256 \times 256$  voxels. At higher resolutions better accuracy may be achieved.

The average and maximum errors by ROI and by transformation are similar even though the ROI vary considerably. Table 4.2 shows the refined ROI values for each week. The reason why the refined ROI is so small in week 6 is due to the fact that



**Figure 4.7:** Magnitude of the error in displacement (millimeters) for the transformations applied.

Week	Ellipse Paramaters (mm)					
	$r_1$	$r_2$	$r_3$	$c_1$	$c_2$	$c_3$
1	6.4212	7.1103	5.8149	31.7222	28.7090	31.0768
2	4.8325	5.5471	3.5433	32.1205	29.8128	29.3310
3	4.1047	6.1601	3.4566	31.6564	27.1298	33.2126
4	4.2579	4.4618	3.0731	31.9380	32.1899	32.7206
5	2.4686	3.5114	2.1633	33.0492	27.5942	30.6513
6	2.0770	2.0770	0.7248	29.4400	29.4400	27.7073

**Table 4.2:** Refined ROI parameters for each week.:  $r_i$  represent the radii of the ellipsoid and  $c_i$  the centre of the ellipsoid, with  $i \in \{1, 2, 3\}$  as the direction.

the initial guess has radius zero since the intensity values of the  $P_{SP}$  data in this week are all below 300 and therefore the windowed  $P_{SP}$  contains no data. The segmentation method terminates from reaching the maximum number of allowed iterations before it can find a better guess.

In overall, the results obtained by these experiments are positive and indicate that the current strategy used to align  $P_{SP}$  data to the  $V_{SP}$  data as described in the process shown in Figure 1.7 is effective. For the simulated transformations, using the aforementioned parameters for the alignment, the process takes in average 3 hours and 15 minutes to complete with 4 minutes and 15 seconds for each week. The time for a single week is distributed between each step as follows.

- Ellipsoid fitting takes 0.5 seconds.
- Segmentation takes 20 seconds.
- Registration takes 3 minutes and 20 seconds.
- File I/O operations 25 seconds.

The machine used is a standard PC using 2 GB of RAM with an AMD Turion X2 processor at 2 GHz and the code is running in Matlab. If higher resolutions were used both in terms of segmentation and registration the results may be more accurate but the computing time and memory requirements would be much larger.

In this chapter some main experimental results were presented in order to give an idea of the effectiveness of the alignment process. In the next chapter, the main features of the applications will be discussed along with suggestions for the improvement of computation time and areas of future research.

## Chapter 5

# Discussion on Current and Future Research

This chapter will discuss some issues of the alignment strategy, specifically with the segmentation and registration methods. A large part of the focus will be given to the segmentation piece. Future research concerning segmentation and alternatives to some parts of the alignment process will also be discussed.

### 5.1 Issues in Segmentation and Registration

The work completed in this thesis has shown some promising results, however, there are a number of issues that still need to be addressed in order to improve the work done. These issues span across the entire alignment process including the registration of the ellipsoids, the segmentation step and the ROI-based registration of the data.

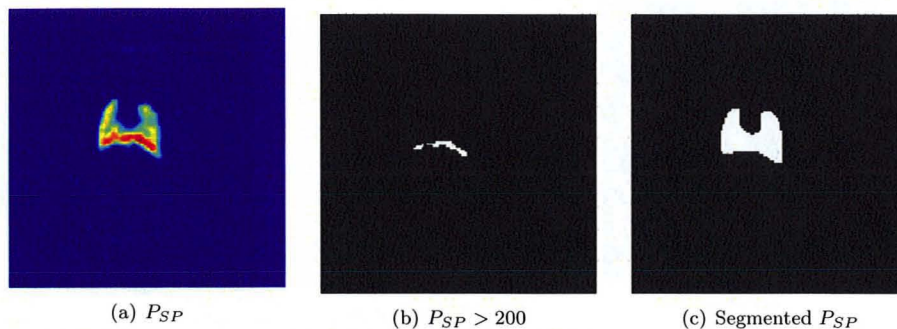
#### Ellipsoid Fitting

As mentioned before, the first step to making an initial educated guess for the segmentation process is to register an ellipsoid to the windowed  $P_{SP}$  data. Given that  $P_{SP}$  data may vary in each scan as the condition of the subject deteriorates or improves the windowing value may need to be different for each scan. While it is possible to set a different thresholding value for each week of data, it is not preferable.

As mentioned in Chapter 1, the  $P_{SP}$  data is windowed to remove regions outside the lungs with small amounts of radiation from the contrast agent. These regions look like a “cloud” around the high intensity values of the lungs, see Figure 5.1(a). The cloud is caused by minor traces of contrast agent spreading possibly through the blood in other areas. If the ellipse were to be registered directly to this data the cloud would also be included and this is not desired, that is why the lungs are windowed at above 300.

Even though windowing is a fast way to remove the cloud, it is not the most reliable as values in the  $P_{SP}$  data may widely vary in different scans. For instance, looking at Figure 5.1(b) a very small part of the lungs has intensity values above 200. If the windowing value was 300, no data would appear and in fact this happened during the experiments in the sixth week of data where the initial guess and refined ROI were very small.

This situation can be avoided, however, by using segmentation instead of windowing. The good news is that there is no need to design a new objective functional since the Chan-Vese approach will produce the desired result. Figure 5.1(c) shows the segmented  $P_{SP}$  data using the Chan-Vese approach without any modifications. While this is a much better way to filter out the surrounding cloud, it needs to be tested against



**Figure 5.1:** Coronal slice of (a)  $P_{SP}$  data from week six showing surrounding cloud, (b) windowed at above 200 shown in white and (c) segmented using the Chan-Vese method with segmented area shown in white.

instances where the Chan-Vese approach would include parts of the cloud. For the data of this project, however, this has shown nice results.

In a further outlook, the segmentation could replace the initial ellipsoid registration entirely by using the resulting  $\phi$  of the Chan-Vese segmentation on the  $P_{SP}$  data as the starting guess for the level-set function in (3.6) for the ROI-refinement step.

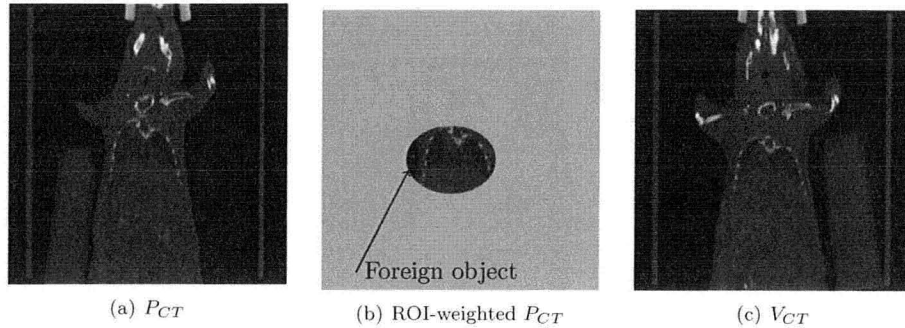
### Segmentation

Currently, the segmentation method makes use of a number of parameters ( $\mu$ ,  $\nu$  and  $\lambda$ ) and a time-step for the flow of  $\phi$  in order to fine-tune it to the desired behaviour. In this project, the time-step has been chosen empirically and is kept constant throughout the evolution of the contour. This is not a preferred strategy since it has a slow rate of convergence [34]. It might happen that the curve, after a certain number of iterations, may change very little and, assuming the termination criteria are not fulfilled, the time-step may need to be increased in order to make significant progress. Furthermore, the parameters of the objective functional (3.1) were also chosen empirically which means that these might have to be slightly changed depending on the type of data being segmented. The optimization of the parameters and the time-step is outside of the scope of this project.

Another issue with the segmentation is that the minimizer is found by using the steepest descent method which has slow convergence [34]. The choice of the steepest descent method in this project is based on the work of [50] and it is also a fast and simple way to test whether the objective function achieves the desired result. The next step is to implement segmentation using the conjugate gradient method in order to improve convergence and completion time.

### ROI-Based Data Registration

The drawback with the registration of the ROI-weighted  $P_{CT}$  data to the  $V_{CT}$  data is that it could be very sensitive to artifacts in the scan such as the inclusion of a foreign object in the ROI. Figure 5.2 shows an example where the  $P_{CT}$  and  $V_{CT}$  data have to be aligned but the ROI-weighted  $P_{CT}$  data includes a small part of the water cylinder. This object could have adverse effects on the registration since it might align the edges of the water cylinders in the two images instead. There may be alternatives to fitting an ellipsoid to the segmented  $P_{CT}$  data, but for now fitting an ellipsoid has shown to be the simplest and most logical way of including additional data from surrounding tissues of the lungs. There may also be other factors that may cause the same problem such as an implant that is later removed and therefore appears in one scan but not the next.



**Figure 5.2:** Given (a) the  $P_{CT}$  data, proper alignment of (b) the ROI-weighted  $P_{CT}$  data to the (c)  $V_{CT}$  data could be obstructed by the existence of the small water cylinder edge in the ROI-weighted  $P_{CT}$  data.

Another topic of discussion for the registration of the data is the restriction to rigid transformations. For the simulated transformations used to test the alignment process, rigid transformations suffice, however, in the future valid transformations may include affine and elastic models.

## 5.2 Conclusions

The final goal of the project is to automatically align ventilation and perfusion SPECT data in order to be able to study the effects of newly developed drugs for various lung diseases including fibrosis and cancer. Due to the fact that the SPECT data from different scans can have large structural differences, pre-aligned CT data can be utilized to correct the alignment problems in the SPECT data. The objective of this thesis was to develop a fully automated process by which different modalities of SPECT data can be aligned by using the respective CT data over a number of weeks. Here, a new objective functional for segmentation was proposed based on the work of [10, 49, 50] in order to find a region of interest in the CT data and utilize it in order to find a rigid transformation aligning the perfusion and ventilation CT data.

Some of the main highlights of this project are the usage of only two pieces of information, namely lung volume and average HU intensity value in the lungs, in order to segment the lungs of one subject over a number of weeks and the usage of multi-level registration in order to avoid local minima and decrease computation time.

The results of the synthetic and real-life experiments so far have been promising and indicate that with some further refinement of the alignment process better precision and lower computation time can be achieved.



## Bibliography

- [1] A. Blake and A. Zisserman. Localizing Discontinuities Using Weak Continuity Constraints. *PRL*, 6:51–59, 1987.
- [2] A. Yezzi Jr. and S. Kichenassamy and A. Kumar and P. Olver and A. Tannenbaum. A geometric snake model for segmentation of medical imagery. *IEEE Transactions of Medical Imaging*, 16(2):, 1997. .
- [3] B. Dacorogna. *Direct Methods in the Calculus of Variations*. Springer, New York, 1989.
- [4] B. Van Brunt. *Calculus of Variations*. Universitext. Springer-Verlag, , 2004.
- [5] C. Li and C. Xu and C. Gui and M. D. Fox. Level set evolution without re-initialization: a new variational formulation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 430–436, 2005.
- [6] C. R. Vogel. *Computational Methods for Inverse Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.
- [7] D. L. Pham and C. Xu and J. L. Prince. A Survey of Current Methods in Medical Image Segmentation. In *Annual Review of Biomedical Engineering*, volume 2, pages 315–338. 2000.
- [8] D. L. Pham and C. Xu and J. L. Prince. A Survey of Current Methods in Medical Image Segmentation. In *Annual Review of Biomedical Engineering*, volume 2, pages 315–338. 2000.
- [9] D. Mukherjee and S. Rajagopalan, editor. *CT and MR Angiography of the Peripheral Circulation: Practical Approach with Clinical Protocols*. Informa Healthcare, March 2007.
- [10] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.
- [11] E. Haber and J. Modersitzki. A Multilevel Method for Image Registration. *SIAM J. Sci. Comput.*, 27(5):1594–1607, 2006.
- [12] E. Rouy and A. Tourin. A Viscosity Solutions Approach to Shape-From-Shading. *SIAM Journal on Numerical Analysis*, 29(3):867–884, 1992.
- [13] E. Seeram. *Computed Tomography: Physical Principles, Clinical Application and Quality Control*. Saunders, Philadelphia, Ps, USA, 2 edition, 2001.
- [14] G. Aubert and L. A. Vese. A Variational Method in Image Recovery. *SIAM J. Numer. Anal.*, 34(5):1948–1979, 1997.

- [15] G. Dal Maso and J.-M. Morel and S. Solimini. A variational method in image segmentation: Existence and approximation results. *Acta Mathematica*, 168:89–151, 1992.
- [16] H. Zhao and T. F. Chan and B. Merriman and S. Osher. A variational level set approach to multiphase motion. *J. Comput. Phys.*, 127(1):179–195, 1996.
- [17] Howard A. Anton. *Calculus Early Transcendentals Single Variable*. John Wiley & Sons, 2006.
- [18] I. Dindoyal and T. Lambrou and J. Deng and A. Todd-Pokropek. Level Set Snake Algorithms on the Fetal Heart. In *ISBI*, pages 864–867. IEEE, 2007.
- [19] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, November 1986.
- [20] J.-M. Morel and S. Solimini. Segmentation of Images by Variational Methods: A Constructive Approach. *Revista Matematica Universidad Complutense de Madrid*, 1:169–182, 1988.
- [21] J. Modersitzki. *Numerical Methods for Image Registration*. Oxford University Press, 2004.
- [22] J. Modersitzki. *FAIR: Flexible Algorithms for Image Registration*. SIAM, 2009.
- [23] J. Palmer and U. Bitzén and B. Jonson and M. Bajc. Comprehensive Ventilation/Perfusion SPECT. *The Journal of Nuclear Medicine*, 42:1288–1294, 2001.
- [24] J. S. Suri and D. Wilson and S. Laxminarayan. *Handbook of Biomedical Image Analysis: Volume 1: Segmentation Models Part A (Topics in Biomedical Engineering International Book Series)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- [25] J.L. Marroquin. Surface Reconstruction Preserving Discontinuities. In *Massachusetts Institute of Technology Memo*, Cambridge, MA, USA, 1984.
- [26] K. Avinash and M. Slanley. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.
- [27] L. Ambrosio and V. Tortorelli. Approximation of Functionals Depending on Jumps by Elliptic Functionals via  $\Gamma$ -Convergence. *Comm. Pure Applied Math.*, 43:999–1036, 1990.
- [28] L. P. Lebedev and M. J. Cloud. *The calculus of variations and functional analysis : with optimal control and applications in mechanics*. Series on stability, vibration, and control of systems., Series A, v. 12 . World Scientific Pub Co Inc, , 2004.
- [29] M. Bajc and B. Jonson. Ventilation/perfusion SPECT an essential but underrated method for diagnosis of pulmonary embolism and other diseases. *European Journal of Nuclear Medicine and Molecular Imaging*, 2009.
- [30] M. Droske and M. Rumpf. A Variational Approach to Non-Rigid Morphological Image Registration. *SIAM Appl. Math.*, 64:668–687, 2003.
- [31] M. Kass and A. Witkin and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, January 1988.
- [32] M. Sussman and P. Smereka and S. Osher. A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow. *Journal of Computational Physics*, 114(1):146–159, September 1994.

- [33] O. Schillaci. Hybrid SPECT/CT: a new era for SPECT imaging? *European Journal of Nuclear Medical Molecular Imaging*, 32(5):521–524, 2005.
- [34] P. E. Gill and W. Murray and M. H. Wright. *Practical optimization*. Academic Press Inc., London, 1981.
- [35] R. A. Adams. *Sobolev Spaces*. Pure and Applied Mathematics; a series of monographs and textbooks. Academic Press, INC, 1975.
- [36] R. Jensen. Uniqueness of Lipschitz Extensions: Minimizing the Sup Norm of the Gradient. *Archive for Rational Mechanics and Analysis*, 123:51–74, September 1993.
- [37] R. Malladi and A. Sethian and B. C. Vemuri. A topology independent shape modeling scheme. In , editor, *Proc. SPIE Conf. Geometric Methods Computer Vision II*, volume 2031, pages 246–258. San Diego, CA, 1993.
- [38] R. Malladi AND J. A. Sethian. Image processing via level set curvature flow. In *Proc. Natl. Acad. Sci. USA Vol. 92, pp. 7046-7050, July 1995 Applied Mathematics*, page , , 1995. .
- [39] R. Malladi and J. A. Sethian and B. C. Vemuri. Shape Modeling with Front Propagation: A Level Set Approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(2):158–175, February 1995.
- [40] R. P. Fedkiw. The Ghost Fluid Method For Numerical Treatment Of Discontinuities And Interfaces, 2000.
- [41] R. P. Fedkiw and T. Aslam and B. Merriman and S. Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152(2):457–492, 1999.
- [42] R. R. Seeley and T. D. Stephens and P. Tate. *Essentials of Anatomy & Physiology*. McGraw-Hill, Fifth edition, 2005.
- [43] S. Chen and B. Merriman and S. Osher and P. Smereka. A simple level set method for solving Stefan problems. *J. Comput. Phys.*, 135(1):8–29, 1997.
- [44] S. Esedoglu and J. Shen. Digital inpainting Based On The Mumford-Shah-Euler Image Model. *European J. Appl. Math*, 13:353–370, 2002.
- [45] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. volume PAMI 6, pages 721–741, 1984.
- [46] S. Osher and E. Fatemi. Nonlinear Total Variation Based Noise Removal Algorithms. *Physica D*, 60:259–268, 1992.
- [47] S. Osher and J. A. Sethian. Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations. *Journal of Computational Physics*, 79:12–49, 1988.
- [48] T. F. Chan and J. Shen and L. A. Vese. Variational PDE models in image processing. *Notices Amer. Math. Soc.*, 50(1):14–26, 2003.
- [49] T. F. Chan and L. A. Vese. A level set algorithm for minimizing the Mumford-Shah functional in image processing. In *VLSM '01: Proceedings of the IEEE Workshop on Variational and Level Set Methods (VLSM'01)*, page 161, Washington, DC, USA, 2001. IEEE Computer Society.
- [50] T. F. Chan and L. A. Vese. Active Contours Without Edges. *IEEE Transactions on Image Processing*, 10(2):, 2001.

- [51] T. F. Chan and S. Esedoglu and K. Ni. Histogram Based Segmentation Using Wasserstein Distances. pages 697–708. 2008.
- [52] V. Caselles and J.-M. Morel and C. Sbert. An axiomatic approach to image interpolation. In *ICIP '97: Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3-Volume Set-Volume 3*, page 376, Washington, DC, USA, 1997. IEEE Computer Society.
- [53] V. Caselles and R. Kimmel and G. Sapiro. Geodesic Active Contours. *International Journal of Computer Vision*, 22:61–79, 1995.
- [54] V. Casselles and F. Catté and T. Coll and F. Dibos. A geometric model for active contours in image processing. *Numer. Math.*, 66():1–31, 1993. .
- [55] W. Kalender. *Computed Tomography: Fundamentals, System Technology, Image Quality, Applications*. Erlangen: Publics Corporate Pub., 2005.
- [56] Zhao, Hong-Kai and Chan, T. and Merriman, B. and Osher, S. A variational level set approach to multiphase motion. *J. Comput. Phys.*, 127(1):179–195, 1996.

## Appendix A

# Matlab Code Extracts

In this appendix the Matlab code extract for the most important parts of the thesis application will be shown. The layout of the code is the following.

- First, the code for the ellipsoid and the objective function used in the ellipsoid fitting steps is shown.
- Following, is the lung segmentation step code, which includes the calculation of the Heaviside function and the divergence operator.
- Next, the distance function and the objective function used in the registration step are shown.
- Finally, the main routine by which all of the above steps are integrated is shown.

### A.1 Matlab Code for Ellipsoid Fitting

Listing A.1: `ell3func.m`

```

1 function [c,dc,dxc] = ell3func(z,x)
2 % parameters:
3 % z = [r1,r2,r3,x0(1),x0(2),x0(3)]
4 %      -> the radii and center of the ellipsoid
5 % x - the points at which to calculate values
6 % output:
7 % c - evaluated ellipsoid
8 % dc - the gradient with respect to radius and center
9 % dxc - the gradient of the ellipsoid
10
11 dc = [];
12 dxc = [];
13 doDerivative = 0;
14 doXDerivative = 0;
15
16 %calculate gradient if necessary
17 if nargin >= 2
18     doDerivative = 1;
19     if nargin == 3
20         doXDerivative = 1;
21     end
22 end
23
24 x = reshape(x, [], 3);
25 x0 = [z(4), z(5), z(6)];
26 x(:,1) = (x(:,1)-x0(1))/z(1);
27 x(:,2) = (x(:,2)-x0(2))/z(2);
28 x(:,3) = (x(:,3)-x0(3))/z(3);

```

```

29
30 % the magnitude of the radii
31 radius = sqrt(x(:,1).^2 + x(:,2).^2 + x(:,3).^2);
32
33 c = zeros(size(radius));
34
35 if doXDerivative
36     dxc = zeros(size(x));
37 end
38
39 if doDerivative
40     drc = zeros(size(radius));
41     dx0 = zeros(size(x));
42 end
43
44 %find the region J where the ellipse is 'smooth'
45 J = find(radius<=1 & radius>=0.5);
46 %calculate the values at the smooth region
47 c(J) = -cos(2*pi*radius(J))/2+0.5;
48
49 %calculate the derivatives in the smooth region
50 if doXDerivative
51     dxc(J,1) = pi*sin(2*pi*radius(J))/z(1)./radius(J).*x(J,1);
52     dxc(J,2) = pi*sin(2*pi*radius(J))/z(2)./radius(J).*x(J,2);
53     dxc(J,3) = pi*sin(2*pi*radius(J))/z(3)./radius(J).*x(J,3);
54 end
55 %derivative with respect to radius and center
56 if doDerivative
57     drc(J,1) = -pi*sin(2*pi*radius(J))/z(1)./radius(J).*x(J,1).^2;
58     drc(J,2) = -pi*sin(2*pi*radius(J))/z(2)./radius(J).*x(J,2).^2;
59     drc(J,3) = -pi*sin(2*pi*radius(J))/z(3)./radius(J).*x(J,3).^2;
60
61     dx0(J,1) = -pi*sin(2*pi*radius(J))/z(1)./radius(J).*x(J,1);
62     dx0(J,2) = -pi*sin(2*pi*radius(J))/z(2)./radius(J).*x(J,2);
63     dx0(J,3) = -pi*sin(2*pi*radius(J))/z(3)./radius(J).*x(J,3);
64 end
65
66 %reuse J for points flat at value 1
67 J = find(radius<=0.5);
68 %set flat points to 1
69 c(J) = 1;
70 %set derivative in the flat regions to 0
71 if doXDerivative
72     dxc(J,1) = 0;
73     dxc(J,2) = 0;
74     dxc(J,3) = 0;
75 end
76
77 if doDerivative
78     drc(J,1) = 0;
79     drc(J,2) = 0;
80     drc(J,3) = 0;
81
82     dx0(J,1) = 0;
83     dx0(J,2) = 0;
84     dx0(J,3) = 0;
85     dc = [drc,dx0];
86 end

```

Listing A.2: objfctn3Dell.m

```

1 function [f,p,df,d2f] = objfctn3Dell(z,alpha,I,X)
2 % calculates the objective function used to minimize
3 % the ellipse fitting the data I
4 % parameters:
5 % I - data
6 % X - data points
7 % z - ellipse parameters

```

```

8 % alpha - positive scalar
9
10 %get ellipse
11 [c,dc] = ell3func(z,X);
12 p = 0; %used for the GNA function options
13
14 %calculate the objective function, gradient and Hessian
15 f = norm(z(1:3)) + alpha*I'*(1-c);
16 df = 1/norm(z(1:3))*[z(1),z(2),z(3),0,0,0] - alpha*I'*dc;
17 d2f = alpha*pi/norm(z(1:3))*eye(6);

```

## A.2 Matlab Code for Lung Segmentation

Listing A.3: Heaviside.m

```

1 function [H,dH] = Heaviside(z,epsilon)
2 % Calculates a smooth/regularized Heaviside function
3 % epsilon - determines the steepness of the slope
4 % z = [r1,r2,r3,c1,c2,c3] - are the parameters of the ellipse where
5 %   r(i) - is the radius in direction i
6 %   c(i) - is the position along direction i
7
8 % calculating the Heaviside function
9 H = 0.5*(1+(2/pi)*atan(z/epsilon));
10
11 if nargin < 2, return; end;
12 % if necessary calculate the first
13 % derivative of the Heaviside function
14 dH = (epsilon/pi)*spdiags(reshape(1./(epsilon^2+z.^2), [],1),0, ...
15     length(z),length(z));

```

Listing A.4: getGradientOperator.m

```

1 function [G1d,G1a,G2d,G2a,G3d,G3a] = getGradientOperator(Omega,m)
2 % calculates the gradient operator given the space Omega and the
3 % resolution m
4 %
5 % Output:
6 % G1d,G2d,G3d - the difference operators in direction 1,2,3
7 % G1a,G2a,G3a - averaging operators to be used on the difference
8 %               operators to bring the values back to the cell
9 %               centers
10 % outside these are used to calculate the magnitude of
11 % the gradient as follows:
12 % |Grad| = sqrt(G1a*(G1d*phi).^2 + G2a*(G2d*Phi).^2 + ...)
13
14 if length(Omega) == 2,
15     G3d = @(Phi) [];
16     G3a = @(Phi) [];
17
18 %difference averaging operators
19 %these are named as avd1_2 =>
20 %     averaging operator on dimension 1
21 %     corresponding to dimension 2
22 avd1 = kron(speye(m(2)),av_dash(Omega,m,1));
23 avd2 = kron(av_dash(Omega,m,2),speye(m(1)));
24
25 %difference operators
26 %forward difference
27 d1p = kron(speye(m(2)),dp(Omega,m,1));
28 d2p = kron(dp(Omega,m,2),speye(m(1)));
29
30 % this part is the "imprecise difference" for the direction

```

```

31  % that is not focal for the computation
32  G1d = d1p;
33  G2d = d2p;
34
35  G1a = avd1;
36  G2a = avd2;
37  else
38  %difference averaging operators
39  %these are named as avd1_2 => averaging operator on dimension 1
40  %                               corresponding to dimension 2
41  avd1 = kron(speye(m(3)),kron(speye(m(2)),av_dash(Ω,m,1)));
42  avd2 = kron(speye(m(3)),kron(av_dash(Ω,m,2),speye(m(1))));
43  avd3 = kron(av_dash(Ω,m,3),kron(speye(m(2)),speye(m(1))));
44
45  %difference operators
46  %forward difference
47  d1p = kron(speye(m(3)),kron(speye(m(2)),dp(Ω,m,1)));
48  d2p = kron(speye(m(3)),kron(dp(Ω,m,2),speye(m(1))));
49  d3p = kron(dp(Ω,m,3),kron(speye(m(1)),speye(m(2))));
50
51  % this part is the "imprecise difference" for the direction
52  % that is not focal for the computation
53  G1d = d1p;
54  G2d = d2p;
55  G3d = d3p;
56
57  G1a = avd1;
58  G2a = avd2;
59  G3a = avd3;
60  end
61
62  function w = dp(Ω,m,i)
63  % calculates the forward staggered difference
64  h = Ω./m;
65  w = spdiags(ones(m(i),1)*[-1,1],[0,1],m(i)-1,m(i))/h(i);
66
67  function M = av_dash(Ω,m,i)
68  % helper operator for the calculation of the
69  % averaging operators above
70  M = spdiags(ones(m(i),1)*[1,1],[-1,0],m(i),m(i)-1)/2;
71  M(1,1) = 1;
72  M(end,end) = 1;

```

Listing A.5: `getDivergenceOperator.m`

```

1  function A = getDivergenceOperator(Ω,m,β)
2  % calculates the discrete divergence operator
3  % \div(\grad\phi/|\grad\phi|)
4  %
5  % Ω - the spatial domain
6  % m - the resolution
7  % β - a small positive regularizing constant for the
8  %     inverse gradient magnitude
9  %output:
10 % A - the linearized divergence operator
11
12 sdiag = @(a) spdiags(reshape(a,[],1),0,length(a),length(a));
13
14 if length(Ω) == 2,
15     %difference averaging operators
16     %these are named as avd1_2 =>
17     %     averaging operator on dimension 1
18     %     corresponding to dimension 2
19     avd1_2 = kron(av_dash(Ω,m,2),av_dot(Ω,m,1));
20     avd2_1 = kron(av_dot(Ω,m,2),av_dash(Ω,m,1));
21
22     %difference operators
23     %forward difference

```



```

24     d1p = kron(speye(m(2)), dp(Ω, m, 1));
25     d2p = kron(dp(Ω, m, 2), speye(m(1)));
26
27
28     %this part is the "imprecise difference" for the direction
29     %that is not focal for the computation
30     d1a_2 = avd1_2*d2p;
31     d2a_1 = avd2_1*d1p;
32
33     G1 = @(Phi) sqrt((d1p*Phi).^2 + (d1a_2*Phi).^2 + β^2);
34     G2 = @(Phi) sqrt((d2a_1*Phi).^2 + (d2p*Phi).^2 + β^2);
35
36     A = @(Phi) -d1p'*sdiag(1./G1(Phi))*d1p ...
37           - d2p'*sdiag(1./G2(Phi))*d2p;
38 else
39     %difference averaging operators
40     %these are named as avd1_2 =>
41     %     averaging operator on dimension 1
42     %     corresponding to dimension 2
43     avd1_2 = kron(speye(m(3)), kron(av_dash(Ω, m, 2), av_dot(Ω, m, 1)));
44     avd1_3 = kron(av_dash(Ω, m, 3), kron(speye(m(2)), av_dot(Ω, m, 1)));
45
46     avd2_1 = kron(speye(m(3)), kron(av_dot(Ω, m, 2), av_dash(Ω, m, 1)));
47     avd2_3 = kron(av_dash(Ω, m, 3), kron(speye(m(1)), av_dot(Ω, m, 2)));
48
49     avd3_1 = kron(av_dot(Ω, m, 3), kron(speye(m(2)), av_dash(Ω, m, 1)));
50     avd3_2 = kron(av_dot(Ω, m, 3), kron(av_dash(Ω, m, 2), speye(m(1))));
51
52     % difference operators for the
53     % staggered forward difference
54     d1p = kron(speye(m(3)), kron(speye(m(2)), dp(Ω, m, 1)));
55     d2p = kron(speye(m(3)), kron(dp(Ω, m, 2), speye(m(1))));
56     d3p = kron(dp(Ω, m, 3), kron(speye(m(1)), speye(m(2))));
57
58     %this part is the "imprecise difference" for the
59     % direction that is notfocal for the computation
60     d1a_2 = avd1_2*d2p;
61     d1a_3 = avd1_3*d3p;
62
63     d2a_1 = avd2_1*d1p;
64     d2a_3 = avd2_3*d3p;
65
66     d3a_1 = avd3_1*d1p;
67     d3a_2 = avd3_2*d2p;
68
69     %each of the computed gradient magnitudes G(i) is exact
70     % in direction i while the other directions have
71     % approximation/assumptions at the boundary
72     G1 = @(Phi) sqrt((d1p*Phi).^2 + ...
73                   (d1a_2*Phi).^2 + ...
74                   (d1a_3*Phi).^2 + β^2);
75     G2 = @(Phi) sqrt((d2a_1*Phi).^2 + ...
76                   (d2p*Phi).^2 + ...
77                   (d2a_3*Phi).^2 + β^2);
78     G3 = @(Phi) sqrt((d3a_1*Phi).^2 + ...
79                   (d3a_2*Phi).^2 + ...
80                   (d3p*Phi).^2 + β^2);
81
82     A = @(Phi) - d1p'*sdiag(1./G1(Phi))*d1p ...
83           - d2p'*sdiag(1./G2(Phi))*d2p ...
84           - d3p'*sdiag(1./G3(Phi))*d3p;
85 end;
86
87
88
89 function w = dp(Ω, m, i)
90 %helper function that calculates forward differences
91 h = Ω./m;
92 w = spdiags(ones(m(i), 1)*[-1, 1], [ 0, 1], m(i)-1, m(i))/h(i);

```

```

93
94 function M = av_dash(Ω,m,i)
95 %this is a helper matrix for the av_dot matrix
96 %while the av_dot takes the average within a column, this matrix,
97 %in combination with the av_dot matrix, calculates the
98 %average between 4 adjacent cells with the assumption
99 %of a contiving slope on the boundary
100 % (always used together with av_dot:
101 % eg. if av_dot(m(1)) -> av_dash(m(2)) )
102 M = spdiags(ones(m(i),1)*[1,1],[-1,0],m(i),m(i)-1)/2;
103 M(1,1) = 1;
104 M(end,end) = 1;
105
106 function M = av_dot(Ω,m,i)
107 %takes the average within cells in one direction
108 %
109 %      | 1 1      |
110 % A 0.5 * |      1 1      |
111 %      |      1 1      |
112 M = spdiags(ones(m(i),1)*[1,1],[0,1],m(i)-1,m(i))/2;

```

Listing A.6: driverChanVese3D.m

```

1 % driver for ChanVese approach in 3D
2 % variables that are assumed to exist at this point
3 % in the alignment process:
4 % -> eOpt - minimal ellipse [parameters] including SP-P lungs
5 % -> Vol - approximate volume of lungs [user input]
6 % -> c0 - approximate average value of lungs [user input]
7 % -> X - the domain grid
8 close all
9
10 %cell size
11 h = Ω./m;
12
13 % this parameter regularizes the volume penalty term
14 β = 1e-5;
15 % parameter for the slope of the Heaviside function
16 ε = 1e-4*min(h);
17
18 % termination criteria
19 tol = 1e-7;
20 eptol = 1e-8;
21
22 %get the image at a desired resolution from CTML
23 % -> CTML contains the data in many resolutions
24 u = CTML{sr_level}.T(:);
25
26 % viewers
27 viewI = @(F,H) imgmontageoverlay(F,H,CTML{sr_level}.Ω, ...
28                               CTML{sr_level}.m,'direction','zxy');
29 viewS = @(Phi) imgmontage(Phi,CTML{sr_level}.Ω, ...
30                          CTML{sr_level}.m,'direction','zxy');
31
32 %objective function parameters
33 % μ - based on Chan-Vese experiments
34 % rest - found emmrically
35 μ = 0.5*255^2;
36 lambda = 1;
37 ν = 1e5;
38 τ = 1e-1*min(h);
39
40 % reshape the grid in order to construct \Phi
41 x = getCenteredGrid(CTML{sr_level}.Ω,CTML{sr_level}.m);
42 x = reshape(x,[,3]);
43
44 %construct initial \Phi based on eOpt resolved before
45 maxEOpt = max(eOpt(1:3));

```

```

46 Phi = sqrt(1/(eOpt(1)/maxEOpt)^2*(x(:,1)-eOpt(4)).^2 ...
47         + 1/(eOpt(2)/maxEOpt)^2*(x(:,2)-eOpt(5)).^2 ...
48         + 1/(eOpt(3)/maxEOpt)^2*(x(:,3)-eOpt(6)).^2);
49 % the reason for using maxEOpt is to keep the initial cone
50 % 'steep' so as to not be affected by the change in \tau
51 Phi = maxEOpt - Phi;
52
53 %calculate initial H
54 [H,dH] = Heaviside(Phi,epsilon);
55 %initialize Hc in order to compare for the stopping criteria
56 Hc = -H;
57
58 % figure(1); clf; viewS(Phi);
59 figure(2); clf; viewI(u,H)
60
61 % A(phi) = \div[ \nabla\phi/|\nabla\phi| ]
62 A = getDivergenceOperator(CTML{sr_level}.Omega,CTML{sr_level}.m,beta);
63 I = speye(length(u),length(u));
64
65 %calculate the gradient operator in order to
66 [G1d,G1a,G2d,G2a,G3d,G3a] = ...
67     getGradientOperator(CTML{sr_level}.Omega,CTML{sr_level}.m);
68
69 %show a 'table' of values
70 fprintf('iter| E | gradient | \Delta_E | \Delta_Phi \n');
71 fprintf('-----\n');
72
73 % follow the flow
74 % \phi' = \mu H' * A(\phi)
75 %         - \nu H' * (u-c_0)^2
76 %         + \lambda H' * (Vol - \int_{\Omega} H)
77 % -----
78 %         (|Vol - \int_{\Omega} H| + \beta)
79 %
80 % Note: \phi is cell-centered, grad/|grad| comes staggered
81 %
82 % \phi^{k+1} - \phi^k
83 % ----- = \mu * H' (\phi) * A(\phi^k) * \phi^{k+1}
84 %         \tau         + H' (\phi) * RHS
85 %
86 %
87 % RHS := - \nu H' * (u-c_0)^2
88 %         + \lambda H' * (Vol - \int_{\Omega} H)
89 % -----
90 %         (|Vol - \int_{\Omega} H| + \beta)
91 iter = 0;
92 Phic = Phi;
93 %calculate initial energy
94 Ec = \mu*sum(prod(h)*dH*sqrt(G1a*(G1d*Phi).^2 + ...
95         G2a*(G2d*Phi).^2 + ...
96         G3a*(G3d*Phi).^2)) ...
97     + \nu*abs(Vol - sum(prod(h)*H)) ...
98     + lambda*sum(prod(h)*(u-c0).^2.*H);
99 %iterate until convergence
100 while (norm(Hc - H) > tol)
101     rhs = dH*( - lambda1*(u-c1(H)).^2 + lambda2*(u-c2(H)).^2 ...
102             - lambda3*(u-c0).^2 ...
103             + lambda4*(Vol - sum(prod(h).*H))./ ...
104             (abs(Vol - sum(prod(h).*H)) + \beta));
105
106     AA = (I - \mu*\tau*dH*A(Phi));
107     Phi = AA \ (Phi + \tau*rhs);
108
109     iter = iter + 1;
110
111 %calculate new H
112 [H,dH] = Heaviside(Phi,epsilon);
113
114 % calculations of the gradient and function

```

```

115 % for stopping criteria
116 grad = dH*( - lambda1*(u-c1(H)).^2 + lambda2*(u-c2(H)).^2 ...
117 - lambda3*(u-c0).^2 ...
118 + lambda4*(Vol - sum(prod(h).*H))./ ...
119 (abs(Vol - sum(prod(h).*H)) + beta)) ...
120 + mu*A(Phi)*spdiags(dH);
121 E = mu*sum(prod(h)*dH*sqrt(G1a*(G1d*Phi).^2 ...
122 + G2a*(G2d*Phi).^2 ...
123 + G3a*(G3d*Phi).^2)) ...
124 + lambda3*abs(Vol - sum(prod(h).*H)) ...
125 + lambda4*sum(prod(h)*(u-c0).^2.*H);
126 fprintf(' %2d | %-12.4e | %-12.4e | %-12.4e | %-12.4e \n', ...
127 iter, E, norm(grad), E-Ec, norm(Phic - Phi));
128
129 %stopping criteria
130 STOP(1) = abs(Ec - E) < tol*(1 + abs(E));
131 STOP(2) = norm(Phic - Phi) < sqrt(tol)*(1+norm(Phi));
132 STOP(3) = norm(grad,'inf') <= tol^(1/3)*(1 + abs(E));
133 STOP(4) = norm(grad,'inf') < eptol;
134 STOP(5) = iter > maxSegIter;
135
136 if(all(STOP(1:3)) || any(STOP(4:5)))
137     if(all(STOP(1:3)))
138         fprintf('Terminated by 3 conditions.\n');
139     elseif(STOP(4))
140         fprintf('Found point close to gradient = 0.\n');
141     elseif(STOP(5))
142         fprintf('Maximum number of iterations reached.\n');
143     end
144     break;
145 end
146
147 %update energy and Phi
148 Ec = E;
149 Phic = Phi;
150
151 if(debug)
152     figure(2); viewI(u,H);
153     pause(0.1);
154 end
155 end

```

### A.3 Matlab Code for Image Registration

Listing A.7: *SSD.m*

```

1 function [J,dJ,H] = SSD(Tc,Rc,O,m)
2 % sum of squared differences function used for registration
3 % parameters:
4 % Tc - template image
5 % Rc - reference image
6 % O - spatial domain
7 % m - resolution
8
9 doH = 0;
10 doG = 0;
11 dJ = 0;
12 H = 0;
13 %find whether the gradient and hessian need to be calculated
14 if nargin == 2
15     doG = 1;
16 elseif nargin == 3
17     doG = 1;
18     doH = 1;
19 end

```

```

20 h = prod( $\Omega$ ./m);
21
22 %calculate the difference image
23 J = h/2*norm(Tc-Rc)^2;
24 if doG
25     dJ = h*(Tc-Rc)';
26 end
27 if doH
28     H = h*speye(length(Tc));
29 end

```

Listing A.8: rigidObjFctn.m

```

1 function [f,p,df,d2f] = rigidObjFctn(TD, Rc,  $\Omega$ , m, E, w)
2 % calculates the objective function for the image registration
3 % parameters:
4 % TD - template image coefficient (to be used for interpolation)
5 % Rc - reference image
6 %  $\Omega$  - the spatial domain
7 % m - the resolution
8 % E - the optimal ellipsoid found after the refinement step
9 % w = [ $\theta_1, \theta_2, \theta_3, d_1, d_2, d_3$ ] - the transformation
10 %   paramters:  $\theta_i$  - the rotation around direction i
11 %                $d_i$    - the translation along direction i
12 % output:
13 % f - the evaluated objective function
14 % df,H - the gradient and the Hessian
15
16 doG = 0;
17 doH = 0;
18
19 if nargin == 3
20     doG = 1;
21 elseif nargin == 4
22     doH = 1;
23 end
24 % E = ellipse on X
25 p = []; % parameters for plots in GNA
26 df = 0;
27 d2f = 0;
28
29 %calculate the template image and its gradient if necessary
30 if ~doG && ~doH
31     X = getCenteredGrid( $\Omega$ ,m);
32     Y = trafo(w,X);
33     Tc = inter(TD, $\Omega$ ,Y);
34 else
35     X = getCenteredGrid( $\Omega$ ,m);
36     % get the transformation
37     [Y dY] = trafo(w,X);
38     % and calculate the values of the template on
39     % the new transformed points Y
40     [Tc dT] = inter(TD, $\Omega$ ,Y);
41 end
42
43 % calculate the objective function and
44 % its gradient and Hessian if necessary
45 if ~doG && ~doH
46     f = SSD(Tc.*E,Rc, $\Omega$ ,m); % the difference image
47 elseif doG
48     [f df] = SSD(Tc.*E,Rc, $\Omega$ ,m); % the difference image
49     dT = spdiags(E,0,length(E),length(E))*dT;
50     df = df*dT*dY; % get gradient of objecctive function
51 else
52     [f df d2f] = SSD(Tc.*E,Rc, $\Omega$ ,m); % the difference image
53     dT = spdiags(E,0,length(E),length(E))*dT;
54     df = df*dT*dY; % get gradient and Hessian
55     d2f = (d2f*dT*dY)'*(dT*dY);

```

```

56 end
57
58 % show the data T(Y), R and the difference image in a plot
59 p = struct('Tc',Tc,'Rc',Rc,'Ω',Ω,'m',m,'Yc',Y,'Jc',f);

```

## A.4 Main Routine

Listing A.9: `regsetup.m`

```

1 % start main routine integrating
2 % ellipse fitting, segmentation and registration
3 clc; clear; close all;
4
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 %***** START OF DECLARATIONS %*****
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8
9 %file path constructor to retrieve data files
10 absPathToImg = 'C:\Users\Alvin\ThesisProject\FAIR-ext\';
11 folderName = @(w,s) fullfile(absPathToImg,'images', ...
12     'Test_Registration',['Week',int2str(w)],s);
13
14 % debug - parameter for segmentation only
15 debug = 1;
16
17 % level at which ellipse fitting
18 % and segmentation will be performed
19 sr_level = 5; % resolution (2^5)^3=32^3
20
21 % default maximum vumber of iterations for segmentation
22 maxSegIter = 50;
23
24 %size of spatial domain of data
25 Ω = 58.88*[1,1,1];
26
27 %note: here the maximum resolution will be 128^3
28 %     therefore if resolution is higher than this
29 %     it will be reduced to 128^3
30
31 %used in ellipse objective function
32 alpha = 1e0;
33 %ellipse initial guess
34 e0 = 29.44*[1,1,1,1,1,1]';
35
36 %segmentation data
37 % Vol - user input parameter representing
38 %     the volume of the lungs in mm^3
39 % c0 - user input parameter for the
40 %     average value of the lungs in HU
41 Vol = prod(Ω./7); %approximate volume of lungs for rat
42 c0 = -300; %approximate average value inside lungs for rat
43 viewImage('reset','viewImage','imgmontage','colormap','gray(256)');
44 st = cputime;
45 %data for all weeks
46 for week=1:6
47     m = 128*[1,1,1];
48     %variable calculating the time for the alignment process
49     startimeweek = cputime;
50
51     fprintf('Week: %d\n',week);
52
53     %'loadAllData' routine loads all the ventilation
54     % and perfusion data for one week in μltilevel mode as:
55     % CTML - Multilevel CT data
56     %     -> CTML{i}.T = CT-P

```

```

57      %           -> CTML{i}.R = CT-V
58      % SPML - Multilevel SPECT data
59      %           -> SPML{i}.T = SP-P
60      %           -> SPML{i}.R = SP-V
61      loadAllData
62
63      % viewer/debugger parameter
64      viewImage('disp')
65
66      %%
67      %% ELLIPSOID FITTING %%
68      %%
69
70      %window the lungs in the SP-P data at above 300
71      Tperf = (SPML{sr_level}.T > 300);
72      fprintf('Starting ellipse search with:\n e0 = %s', ...
73             mat2str(e0));
74      hfctn = @(ec) objfctn3Dell(ec,alpha,Tperf(:), ...
75                             getCenteredGrid(Ω, SPML{sr_level}.m));
76      % use Gauss-Newton-Armijo method
77      % to find the minimal ellipsoid
78      [eOpt eHist] = GNA(hfctn,e0);
79      clear Tperf %discard Tperf to save space
80
81      fprintf('Found optimal ellipse\n eOpt = %s\n', mat2str(eOpt));
82
83      %%
84      %% SEGMENTATION %%
85      %%
86
87      outfile = ['setupRegData_week',vm2str(week),'.mat'];
88      if ~exist(outfile,'file')
89          % start segmentation part
90          fprintf('Using optimal ellipse data to segment.\n');
91          % start segmentation by using eOpt
92          % as the initial guess
93          driverChanVese3D
94          fprintf('Done\n');
95      else
96          % if for this week of data the ellipse
97          % already exists then load the saved eOpt
98          load(outfile);
99      end
100
101      % now that the lungs are indicated by {\phi>0}
102      % start second ellipse registration part
103      fprintf('Fitting ellipse on segmented data.\n');
104      hfctn = @(ec) objfctn3Dell(ec,alpha,H > 0.5,x);
105      [eOpt eHist] = GNA(hfctn,e0);
106
107      % save the ellipse for future use on the same data
108      eOptList(:,week) = eOpt;
109      fprintf('Found final optimal ellipse\n eOpt = %s\n', ...
110             mat2str(eOpt));
111
112      % setup the ellipse E and heaviside H
113      % in multiple resolutions
114      setupRegData
115
116      % set the options for display and interpolation of the
117      % data during registration
118      viewImage('disp')
119      inter('reset','inter','splineInter3Dmex','regularizer', ...
120           'moments','θ',1e-2);
121      trafo('reset','trafo','rigid3D');
122
123      %%
124      %% MULTILEVEL REGISTRATION %%
125      %%

```

```

126
127     w0global = trafo('w0');
128     % start multi-level registration from (2^3)^3 = 8^3 voxels
129     % incrementatly up to (2^6)^3 = 32^3 voxels
130     fprintf('Starting multi-level registration\n');
131     for level=3:6
132         % at the lowest level start with
133         % the initial guess w0global
134         if level==3
135             w0 = w0global;
136         else
137             w0 = wOpt;
138         end
139         % use the minimzer of the last level as
140         % the starting point of the new level
141         fprintf('Starting at:\n w0=%s\n', mat2str(w0));
142         Ω = CTML{level}.Ω;
143         m = CTML{level}.m;
144         fprintf('Using:\n m=%s\n',mat2str(m));
145         E = EML{level}.T(:); %use ellipsoid with level 'level'
146
147         [TD, RD] = inter('coefficients',CTML{level}.T, ...
148             CTML{level}.R,Ω);
149         objFctn = @(w) rigidObjFctn(TD, inter(RD,Ω, ...
150             getCenteredGrid(Ω,m)), Ω, m, E, w);
151
152         % initialize plots to view registration
153         FAIRplots('set','mode','PIR-ML','fig',level,'plots',1, ...
154             'direction','zxy');
155         FAIRplots('init',struct('Tc',TD,'Rc',RD,'Ω',Ω,'m',m));
156
157         %use Gauss-Newton-Armijo method to find the minimizer
158         [wOpt wHist] = GaussNewtonArmijo(objFctn, w0, ...
159             'maxIter', 100,'LSMaxIter', 10, ...
160             'Ystop',w0global,'Plots',@FAIRplots);
161
162         fprintf('Optimal movement:\n wOpt=%s\n', mat2str(wOpt));
163         % keep a list of the optimal values
164         % and their history for analysis
165         wOptList(:,level-2,week) = wOpt;
166         wHistList{level-2,week} = wHist;
167     end
168     fprintf('Registration Complete\n');
169     totaltimeweek(week) = cputime - starttimeweek; %total time
170 end
171 fprintf('Total time: %f\n\n',cputime - st);
172 % save history, minimizers and totaltime
173 save('results.mat','wOptList','wHistList','eOptList', ...
174     'totaltimeweek');

```