# SEGMENTATION OF 3D BRAIN MRI IMAGES USING ACTIVE CONTOURS AND SHAPE PRIORS

by

Radina Droumeva

B.Sc., Simon Fraser University, 2008

THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN THE DEPARTMENT

OF

MATHEMATICS

© Radina Droumeva 2010

SIMON FRASER UNIVERSITY

Summer 2010

All rights reserved. However, in accordance with the Copyright Act of Canada, this work may be reproduced, without authorization, under the conditions for Fair Dealing. Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review, and news reporting is likely to be in accordance with the law, particularly if cited appropriately.

# APPROVAL

| | |
|---|---|
| **Name:** | Radina Droumeva |
| **Degree:** | Master of Science |
| **Title of Thesis:** | Segmentation of 3D Brain MRI Images Using Active Contours and Shape Priors |
| **Examining Committee:** | Dr. Ralf Wittenberg (Chair) |

Dr. JF Williams
Senior Supervisor

Dr. Steve Ruuth
Supervisor

Dr. Faisal Beg
Supervisor

Dr. Nilima Nigam
Examiner

**Date Approved:** Aug. 10, 2010

SFU SIMON FRASER UNIVERSITY
LIBRARY

# Declaration of
# Partial Copyright Licence

The author, whose copyright is declared on the title page of this work, has granted to Simon Fraser University the right to lend this thesis, project or extended essay to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users.

The author has further granted permission to Simon Fraser University to keep or make a digital copy for use in its circulating collection (currently available to the public at the "Institutional Repository" link of the SFU Library website <www.lib.sfu.ca> at: <http://ir.lib.sfu.ca/handle/1892/112>) and, without changing the content, to translate the thesis/project or extended essays, if technically possible, to any medium or format for the purpose of preservation of the digital work.

The author has further agreed that permission for multiple copying of this work for scholarly purposes may be granted by either the author or the Dean of Graduate Studies.

It is understood that copying or publication of this work for financial gain shall not be allowed without the author's written permission.

Permission for public performance, or limited permission for private scholarly use, of any multimedia materials forming part of this work, may have been granted by the author. This information may be found on the separately catalogued multimedia material and in the signed Partial Copyright Licence.

While licensing SFU to permit the above uses, the author retains copyright in the thesis, project or extended essays, including the right to change the work for subsequent purposes, including editing and publishing the work in whole or in part, and licensing other parties, as the author may desire.

The original Partial Copyright Licence attesting to these terms, and signed by this author, may be found in the original bound copy of this work, retained in the Simon Fraser University Archive.

Simon Fraser University Library
Burnaby, BC, Canada

# Abstract

Image processing is an important tool in research and treatment design in the medical industry. One area of image processing is segmentation: the process of finding objects in an image. This thesis develops a segmentation algorithm for finding the eyeballs in 3D MRI human brain images. An integral part of this is the prior knowledge of the general position, shape, and size of the eyeballs. The segmentation is done in two stages: first, the most likely position of the eyeballs is found using a shape prior (a representation of two spherical objects with specific size and distance apart), followed by a stage of more precise active contour segmentation in the relevant area. The design process of this algorithm can be used to develop a similar segmentation procedure to look for different objects in the human brain, or in other types of images.

**Keywords**: image processing; medical imaging; segmentation; active contours; shape priors;

# Contents

# List of Figures

# Chapter 1

# Introduction

Image processing is a well developed field which has existed for centuries. The original setting for this was to create and restore art, historical documents, etc. – these are physical images. However, in the last few decades, with the emergence of computers and their powerful influence on technological advancement, image processing has expanded into far more. Application areas are being opened up whenever image data can be gathered.

## 1.1 The Segmentation Problem

In the context of computing science, (digital) image processing encompasses image enhancement, restoration, and the extraction of some useful attribute/information. An example of the latter is image segmentation. Segmentation is a process applied to an image, which aims to produce a separation of the image into regions, called objects. A simple example of this is illustrated in Figure 1.1: we have an image containing a person and wish to separate the person from the background. The figure shows a few steps of a segmentation algorithm, with the blue curve indicating the separation of two regions: one is the person, and the other is the background. The background itself is also considered an object. The first snapshot of the algorithm shows an initial curve placed overtop of the image, to indicate our "guess" for what the segmentation is. The following two instances demonstrate how the curve evolves, using constraints induced by the particular segmentation algorithm and the image itself. The final shows the result: the desired separation of the man from the

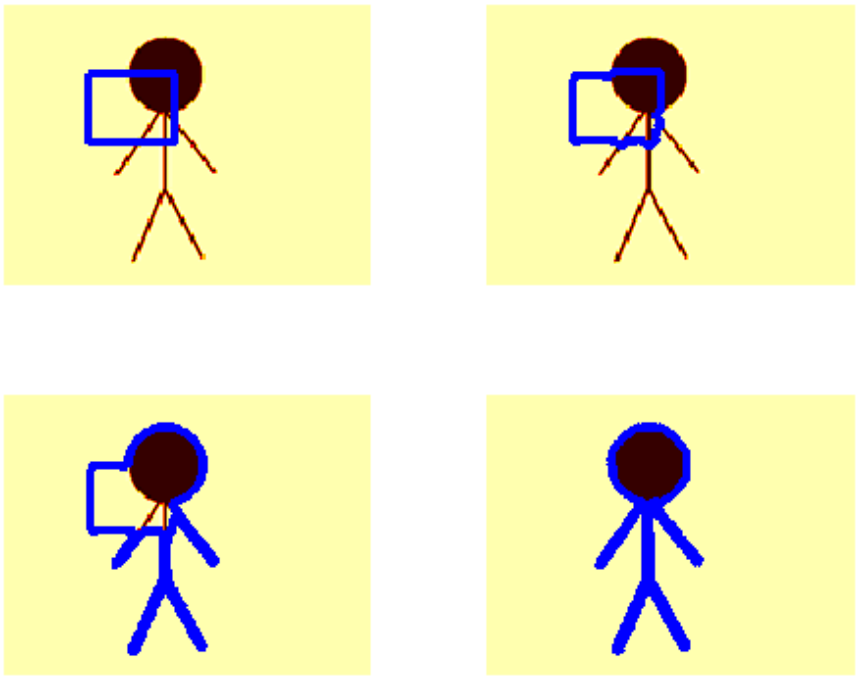

Figure 1.1: Segmentation: separating objects in an image

background has been achieved.

Image segmentation is a tool used in various applications – image editing software such as Photoshop; facial recognition software; video tracking; airport security and others. Medicine is another major field which has greatly benefited from computerized image processing. Segmentation in particular is used as an aid in pathology, computer assisted surgery, tumour detection, diagnosis and more. The difficulty of finding objects in an image varies greatly with the specific application. In medical imaging the main difficulties come from various sources: poor image quality due to noise or acquisition limitations, anatomical structures that naturally overlap, and great variability of shapes. The quality of a segmentation algorithm will be partially determined by how it handles these problems.

## 1.2 Existing Approaches for Segmentation

Many different approaches exist for the segmentation problem. One of the simplest and most intuitive is intensity based segmentation [11]. Here image pixels are grouped together (into objects) directly based on their intensity values. A more sophisticated approach, also based on pixel-by-pixel operations, involves morphological operations [11]. This approach relies heavily on treating pixels using elementary set operations and encompasses a number of tools, such as erosion and dilation, which are applied to neighbourhoods of pixels to group like-pixels together. This is followed by a labelling sweep of the transformed image, to assign like-pixels membership to the objects that result.

Other approaches have to do with direct transformations of the image data – eg. spherical wavelet transformations for 3D segmentation [18]. Statistical tools and theory can also contribute to developing segmentation algorithms. Principal component analysis (using eigenvalue analysis) is a statistical tool which aims to represent an image by few but integral pieces of information, which in turn provide information about what the major objects in the image are. A popular approach which shares this idea is k-means segmentation [10]. This is a combination of a statistical approach and intensity based segmentation. Another way statistics can be used is by modelling a posteriori position and shape of objects in algorithms which simultaneously use shape information and local segmentation techniques [13].

Although the above approaches describe powerful segmentation methods, this thesis explores a different category of procedures. Image gradient based edge detector algorithms, in particular, motivate the ultimate segmentation method used here – Chan and Vese's active contours without edges [5]. These are discussed in more detailed in Chapter 2.

## 1.3 Motivation for Deformable Models

To motivate the choice of Chan and Vese's algorithm, we first consider some major draw backs of the previously mentioned approaches for our chosen application. The nature of medical images makes it difficult to employ straight forward pixel-by-pixel based techniques. These images tend to be noisy and involve many overlapping objects. Many of the

above methods are sensitive to noise. Classical edge-detection approaches (eg. gradient based edge detection) can fail when the objects do not have clearly defined sharp boundaries, or have discontinuous boundaries. For these reasons, deformable models offer a more robust approach.

Deformable models use curves (surfaces when working with 3D images) to represent the desired segmentation. These curves are evolved iteratively based on some internal and external forces. The internal forces are dependent on the state of the evolving curve and are specified by the particular segmentation method selected – eg. conditions can be imposed to retain a level of smoothness for the curve at every iteration. This is done, for example, to keep the curve from becoming too jagged, even if the object we are trying to find has an artificially jagged edge (due to low image quality resulting in pixelated boundaries). The external forces define the portion of the segmentation method that depends on the particular image we are segmenting. These also vary from method to method, but must utilize information from the original image to guide the curve towards the object. By having both internal and external forces, we maintain a smooth curve being evolved and guided towards the areas deemed by the procedure to be objects. In this way, sensitivity to noise is minimized and problems due to object boundary discontinuity are avoided.

## 1.4 Notation and Main Approach

We now introduce some notation to make the above more explicit. Consider a 2D grayscale image, $I_0$. The quantity $I_0(x,y)$ represents the intensity value (a number between 0 and 255) at pixel location $(x,y)$: the $x^{th}$ row, $y^{th}$ column of the image. One way to represent the desired segmentation is using level sets. We define a function $\phi$ within the domain of the image, $\Omega$, whose zero level set will represent the object boundaries. We choose it to be positive "inside" an object, and negative "outside". This is illustrated in Figure 1.2: the image contains one object, in yellow, and the segmentation $\phi$ is defined on the whole domain. Here $C$ is the curve we are really interested in, represented by the zero level set of $\phi$. This function is what we evolve according to some internal and external forces, yet to be specified. Since we are evolving $\phi$, we introduce an artificial time, $t$, into the procedure.

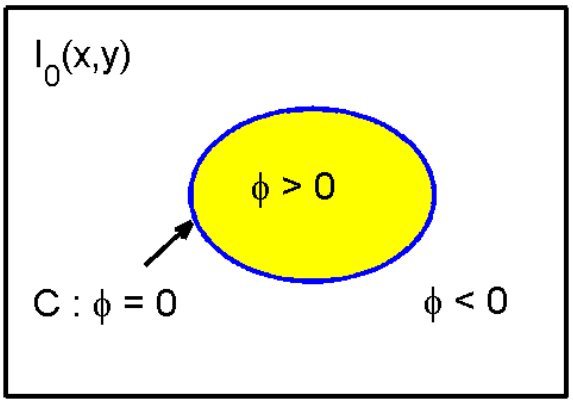


Figure 1.2: Curve $C$ representing the boundaries of our object, implicitly defined as the zero level set of a function $\phi$

In addition to the segmentation $\phi$, this method also uses shape information particular to our application. We will therefore need to use a representation of a target shape that contains information about what we are looking for. This can be naturally defined in a similar fashion as $\phi$, where the zero level set will represent the boundaries of the shape object we want to find. We call this shape representation $\psi$.

The particular application considered here is looking for eyeballs in human brain images. The images used are assumed to have been centered, cropped, and oriented the same way. A 2D slice of a full brain MRI image is shown in Figure 1.3. Assuming this is the case for all of our images, we can make an educated guess as to the definition of the shape object, $\psi$. Based on the relative size of the image, we create a level set representation of $\psi$, and position it somewhere on the right hand side of the image. We now perform a few iterations of an energy minimization evolution, to attract the shape prior to a more suitable position close to the actual eyeballs. We will allow translation of $\psi$, without any scaling or rotation. This is sufficient based on the assumptions that the images have been properly pre-processed, and the variation in eyes in humans is not great enough to affect this step [7].

Once we have found the general area in which the eyeballs are located, we aim to find a more precise boundary definition. Now, we constrict our domain to be only within a region encompassing the eyes, hence ignoring all the other objects in the brain. At this

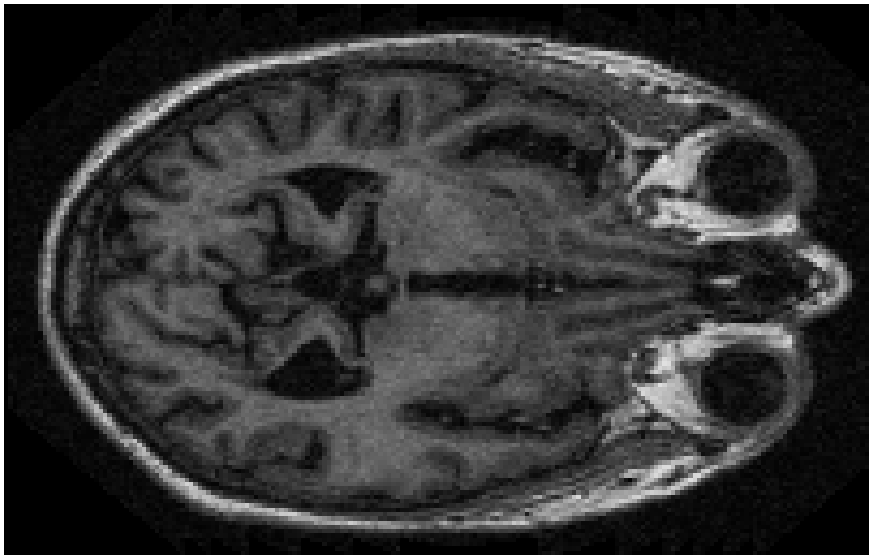

Figure 1.3: Pre-processed brain image: centered, cropped, oriented with eyes on the right

point we evolve $\phi$ for a few iterations, using the current definition of our shape prior $\psi$ as an initialization for $\phi$. This is indeed our best guess for where the object we aim to find is. The algorithm used here is the Chan and Vese active contours without edges segmentation method [5]. This involves a definition of the previously mentioned internal and external forces – specifying smoothness for the segmentation $\phi$, and using appropriate information of the active image domain, to accurately outline the eyes in the brain image.

The purpose for getting an accurate segmentation of the eyes is to allow medical researchers to explore variability in human eyes. By segmenting a large number of images, we are, for example, able to get a distribution of the average size, shape, and position of the eyeballs. This could be useful in medical research to try and correlate these results with certain disorders, or to diagnose abnormalities, etc. Moreover, we are interested in generating robust algorithms which can be applied to identify a generic anatomical feature in a large set of data. To be practical, our approach needs to be able to handle noisy data and provide an easy way to switch the target shape from eyeballs to something else, defined by the particular anatomical feature. Another important attribute is computational efficiency. This thesis aims to segment images in 3D, and this increases the amount of work our method will do drastically: a simple 100 by 100 pixel image in 2D has a total of $10^4$

pixels; however, in 3D we may be dealing with 100 by 100 by 100 pixel data, translating to a total of $10^6$. In fact, our images are 160 by 256 by 256 pixels (for a total of approximately $10^7$ pixels) for the 3D experiments.

The organization of this thesis is as follows. Chapter 2 explores the details of the Chan and Vese segmentation algorithm in more depth and presents some examples of the segmentation process. Chapter 3 discusses the use of shape information, and different approaches used in practice. A final model resulting in the combining of the Chan and Vese segmentation with the shape information incorporated is presented for 2D images. Chapter 4 generalizes this model to 3D and shows some results. Chapter 5 concludes the thesis and includes a discussion about suggested future research, which can improve the overall quality of the method.

## 1.5 A Note on Computations

All numerical computations presented in this thesis were performed on a laptop using Matlab. The focus of this thesis is to present a proof of concept algorithm rather than develop an optimal one. All timings are preliminary and presented merely as a way to compare approaches and not to make definitive claims about the speed of this approach.

# Chapter 2

# Chan and Vese's Active Contours Without Edges

This segmentation method finds all objects in an image, where boundaries of objects are not necessarily defined by gradient (a sharp jump in intensity values). The optimal segmentation arises from minimizing an energy, and is arrived at by evolving a level set curve. To facilitate smoothness in the curve evolution, a motion by mean curvature is employed. A Partial Differential Equation (PDE) formulation of the problem is presented, followed by an iterative method for solving the PDE numerically using finite differences. The benefits of this algorithm over others of its class include the flexibility of having objects with or without sharp edges, placing the initial curve anywhere, automatically finding interior contours, and low sensitivity to noise.

## 2.1 Motivation for the Algorithm

In the classical segmentation approaches using curve evolution, we begin with a curve $C$, defined over the domain of our image, $I_0$. We then evolve the curve according to some constraints imposed by the particular image, to obtain a final state of the curve. This final curve will define the boundaries of the objects found by the procedure. One of the original algorithms in this category is the classical snakes model [12]. In this model the parametrized curve, $C(s) : [0,1] \rightarrow \mathbf{R}^2$, is found to minimize the energy $E_1(C)$ :

$$E_1(C) = \alpha \int_0^1 |C'(s)|^2 ds + \beta \int_0^1 |C''(s)|^2 ds - \lambda \int_0^1 |\nabla I_0(C(s))|^2 ds \tag{2.1}$$

Here $\alpha$, $\beta$, and $\lambda$ are constant (positive) parameters. Since we are trying to minimize $E_1(C)$, we want to make the first two terms as small as possible, and the third as large as possible. The first two define the previously mentioned internal forces, imposing smoothness on our curve. The last term is the only one which depends on the particular image we are working on. This is the external force guiding the segmentation process. In the case of the classical snakes model, this is directly related to the gradient of the image. The reason for this is that we expect to see a sharp edge along the boundary separating two objects, and hence resulting in a large gradient in that area. By "edge" here we refer to a curve segment defining the separation of one object from another. So, we expect this energy to be minimized along such a boundary, while maintaining a smooth curve resting along it. The constant parameters can be chosen in a way defining the trade-off between how much we care about smoothness versus precision of the resulting curve that defines our segmentation. Looking at the energy terms above, we see that the larger we choose the parameter $\lambda$ to be, the more sensitive the procedure will be to image gradient without much regard for maintaining a smooth segmentation curve. On the other hand, by choosing $\alpha$ and $\beta$ to be large, we enforce smoothness of the curve. Some smoothness is necessary to keep the segmentation procedure from degrading and failing – this depends on the particular method used as well as on the nature of the objects in the given image.

The basic idea behind snakes, whose name comes from the similarity between a moving snake and the evolving curve over time, can be extended to more sophisticated methods in a variety of ways, resulting in a class of snake models, most of which are heavily reliant on gradients. More generally, an edge detector function can be defined to drive the speed of the smoothly evolving curve, such that it vanishes at edges (defining a divider between objects in the original image). Gradient based segmentation involves defining such a function, $g(|\nabla I_0|)$, with the property that it is positive in regions with low gradient, and vanishes in regions with high gradient: $\lim_{x\to\infty} g(x) = 0$. With this property we expect the curve to stop exactly at places with high gradients – this is where we expect the boundary of an object to be. A simple example of a class of functions with this property is:

$$g(|\nabla I_0|) = \frac{1}{1+|\nabla I_0|^p} \quad , \qquad \mathrm{p} \geq 1 \tag{2.2}$$

To consider ways which drive the motion of the curve $C$ (eg. in the normal direction), we first consider how to define it. Various methods define the curve we are interested in as the zero level set of a function, $\phi(x,y,t)$, as mentioned in Chapter 1: $C(t) = \{(x,y)|\phi(x,y,t) = 0\}$, where $t$ is the artificial time introduced for the evolution procedure. With this definition, we can evolve $C$ in its normal direction with speed $F$ by solving the following PDE:

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= |\nabla \phi| F \\ \phi(x,y,0) &= \phi_0(x,y) \end{aligned} \tag{2.3}$$

Here $\phi_0(x,y)$ is an initial "guess" for the segmentation. A common choice for $F$ is motion by mean curvature, in which the normal velocity of the curve equals its mean curvature [23]. This amounts to setting $F = \mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)$. One straightforward way to combine motion by mean curvature and gradient based edge detectors is to multiply the right hand side of equation (2.3) by a function $g(|\nabla I_0|)$, so that the velocity of the curve becomes 0 where $g$ is 0 – along a boundary. This results in the equations:

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= g(|\nabla I_0|)|\nabla \phi| \left(\mathrm{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)\right) \\ \phi(x,y,0) &= \phi_0(x,y) \end{aligned} \tag{2.4}$$

One issue that comes up from using an edge detector $g$ is that it depends on a gradient that is evaluated numerically on a discrete grid. This means that $g$ will never be exactly zero, so the curve's speed may never be exactly zero. This allows for the curve to sometimes accidentally pass through the boundary of an object and entirely miss that object – this is particularly problematic with explicit time stepping methods. Another issue with these edge detectors is that if the object has a smooth or blurry boundary, although the curve may slow down as it enters the boundary region, it is difficult to define a stopping

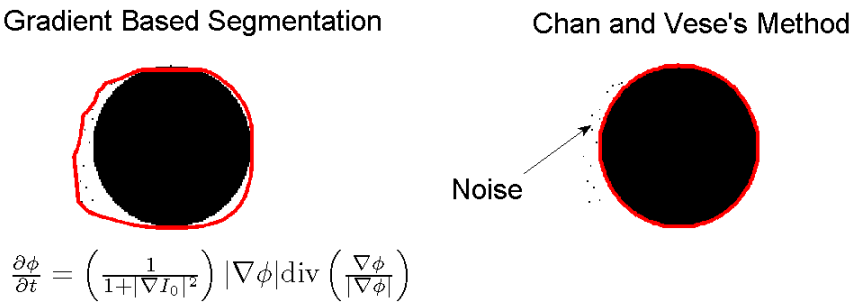


Figure 2.1: Noise sensitivity of a simple gradient based segmentation method, contrasted with Chan and Vese's method

criterion that will be consistent for other objects in the same image. Yet another issue is noise: the gradient around noisy pixels is misleadingly high, causing these methods to classify noise clusters as objects, or incorrectly stopping the curve evolution prematurely. One solution that is employed in these methods is smoothing the image $I_0$ before applying the segmentation procedure. For example, a convolution with a Gaussian is commonly used in practice. This can get rid of noise, but will also smooth the edges that are so essential to gradient based methods. An example of a poorly smoothed image is shown in the third picture of Figure 2.2.

An example of the problem with noise sensitivity for gradient based methods is shown in Figure 2.1. In the first picture we evolve a curve according to equation (2.4), with the function $g$ as in (2.2) with $p = 2$. We can see that the small amount of noise close to the left side of the boundary of the object causes the evolving curve to stop prematurely. This evolution took approximately 3,000 iterations (10 minutes). In contrast, the Chan and Vese method is shown in the second picture. We can see that the object is accurately found in spite of the noise present. The parameter values for this computation, to be described in more detail in the following section, are: $\varepsilon = 1, \mu = 1000, \lambda = 0.1, \Delta t = 0.001$, 15 iterations (3 seconds).

Using the main ideas of the snakes method, Chan and Vese's segmentation model is

Figure 2.2: Piecewise constant approximation to a brain image compared to a smoothed version of the same image

based on the Mumford-Shah energy functional [17], which uses piecewise smooth (constant in our case) approximations to regions of the image $I_0$ to determine the proper segmentation of objects. This is essentially intensity based segmentation, which can capture objects with or without sharp boundaries. It is not particularly sensitive to noise, as the piecewise constant approximation will not be affected by a few noisy pixels. The Mumford-Shah functional is:

$$E_{MS}(I_0, C) = \mu \,\text{Length}(C) + \lambda \int_{\Omega} |I_0(x,y) - I_{MS}(x,y)|^2 \, dxdy + \int_{\Omega \, C} |\nabla I_{MS}|^2 \, dxdy \tag{2.5}$$

The aim is to minimize this with respect to $I_{MS}$, which is the desired piecewise constant approximation of the original image, $I_0$. An illustration of this idea is shown in Figure 2.2, where the first picture is the original image, the second is the piece-wise constant approximation to it using only two separate regions (brain and background) obtained by Chan and Vese's method, and the last is a heavily smoothed version of the original image using a Gaussian filter. The last image is included to illustrate the dangers of smoothing an image prior to segmenting it – some gradient based methods do this to decrease noise sensitivity, but this may also result in loss of edge information.

Chan and Vese's method also employs an implicit level set definition for the evolving curve $C$. This has two major benefits: the initial curve imposed by choosing $\phi_0(x,y)$ can be anywhere in the image domain. The level set formulation also allows for automatic

detection of interior contours: eg. if the object we are looking for is an annulus, we can have a zero level contour along both circles defining the boundaries of the annulus, without special alterations to the code. Some simple edge detector based algorithms might start with the curve $C$ encompassing the whole annulus, and iteratively shrink until it reaches the outer boundary of the annulus, and terminating as the gradient would be high along this outer edge. These types of methods are called balloons or bubbles [21], and can in certain cases be sufficient depending on the application. It should be noted that the gradient-based segmentation algorithms have a nice intuitive interpretation as well, and so does Chan and Vese's method, as will be described in the next section. We aim to use the idea of representing an image by regions of constant intensity, as in the middle picture of Figure 2.2, by formulating it in mathematical terms.

## 2.2 The Model

### 2.2.1 Simple Case: Binary Image

Mathematically, Chan and Vese's method arises from an energy minimization problem. To help better explain the intuition behind it, let us first look at a very simple version of the segmentation problem. Consider an image $I_0(x,y)$, with the image domain denoted by $\Omega$. Suppose $I_0$ is composed of a constant intensity background and one object, of a different constant intensity. Denote these by $I_0^o$ and $I_0^i$, with superscripts denoting outside and inside the object, respectively. Let $C_0$ be the curve describing the desired segmentation: the boundary of the object. Now, to consider the basic idea of the model, we look at an appropriate energy we wish to minimize. For some curve $C$ in the image domain $\Omega$, we define the energy associated with this curve as:

$$\begin{aligned} E(C) &= E_{\text{inside}(C)} + E_{\text{outside}(C)} \\ &= \int_{\text{inside}(C)} |I_0(x,y) - c_1|^2 dxdy \\ &+ \int_{\text{outside}(C)} |I_0(x,y) - c_2|^2 dxdy \end{aligned} \tag{2.6}$$

where

$$c_1 = \text{average intensity of } I_0 \text{ inside C},$$
$$c_2 = \text{average intensity of } I_0 \text{ outside C}.$$

Notice that $E(C) \geq 0$ for any curve C in $\Omega$, and in particular, $E(C_0) = 0$, indicating that $C_0$ indeed achieves the optimal solution in the minimization problem. To better illustrate the nature of this energy, we consider the four possible unique placements of a curve $C$ in the image, and examine the values of the terms $E_{\text{inside}(C)}$ and $E_{\text{outside}(C)}$:

- Case 1: The curve $C$ is outside the object, encompassing the entire object and some background area. In this case, $E_{\text{outside}(C)} = 0$, as the average intensity outside the curve is precisely $I_0^o$. However, $E_{\text{inside}(C)} > 0$, since the intensity inside the curve will be a weighted average of $I_0^o$ and $I_0^i$, with weights determined by the respective areas of background and object.

- Case 2: The curve is strictly inside the object. This is essentially Case 1 in reverse. Here we have $E_{\text{inside}(C)} = 0$, but $E_{\text{outside}(C)} > 0$.

- Case 3: The curve partially encompasses the object and some of the background. In this case both $E_{\text{inside}(C)} > 0$ and $E_{\text{outside}(C)} > 0$.

- Case 4: The final case is when the curve lies on the boundary of the object, ie., $C = C_0$. In this case we have that both components of the energy are 0.

This is illustrated in Figure 2.3 and provides a natural interpretation of the basic idea behind Chan and Vese's method. Notice that this does not involve the gradient of the original image, but simply the distribution of intensity values within the image domain. Another point that comes out of this formulation is that this method is inherently insensitive to noise: one can imagine adding a few yellow pixels to the background in the image in Figure 2.3, and still Case 4 is optimal out of the ones presented. Theoretically, the absolute minimum would require the curve to break up into a few pieces, one of which is as in Case 4 and the others encompassing each (or connected groups) of the noise pixels. This can be easily addressed by adding some further conditions on the curve $C$.

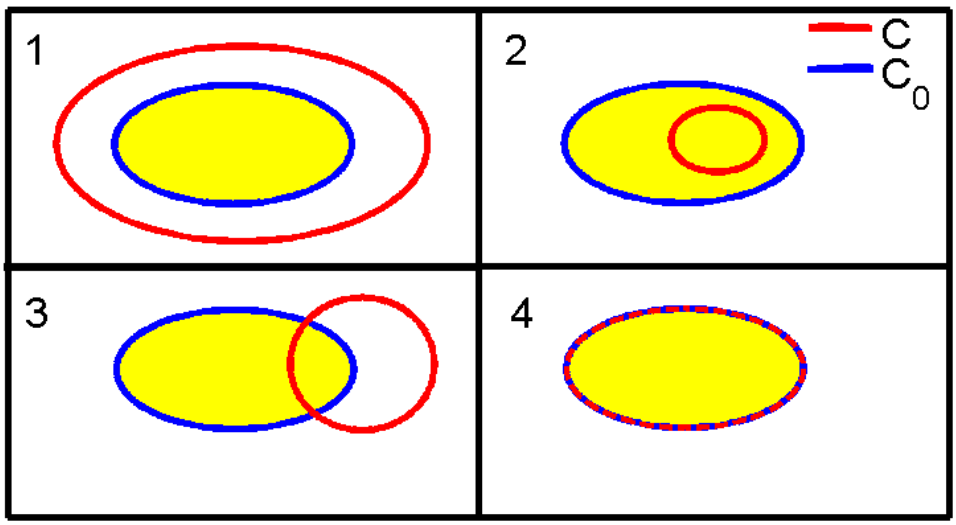


Figure 2.3: Illustrating the basic idea behind Chan and Vese's method: four scenarios showing how the energy is minimized

### 2.2.2 Adding Regularization

Using the idea illustrated here as a starting point, we can now begin to add some regularization to the segmentation procedure to arrive at the complete model. To this effect, Chan and Vese introduce two additional terms into the energy. The first is meant to minimize the length of $C$. This is to ensure that as the curve evolves, it does not get too wiggly – this could happen due to low image quality compromising the integrity of the boundary of an object, or if significant noise is present. The other term is to minimize the area inside $C$ – this can force the larger "object" to be deemed background. Also, if we are looking for more than one object, which requires a multiphase approach using this algorithm [6], this can reduce the clustering of nearby objects as one. The multiphase approach aims to look for more than a single object in an image, for example for two objects, by evolving two separate segmentation curves at the same time – this can be easily achieved by evolving two separate functions $\phi_1$ and $\phi_2$, whose respective zero level sets represent each of the two curves.

Including these two terms, the full energy becomes:

$$E_{CV}(c_1, c_2, C) = \mu \cdot \text{Length}(C) + \nu \cdot \text{Area}(\text{inside}(C)) + \lambda_1 \int_{\text{inside}(C)} |I_0(x,y) - c_1|^2 dxdy + \lambda_2 \int_{\text{outside}(C)} |I_0(x,y) - c_2|^2 dxdy \tag{2.7}$$

We have parameters $\mu \geq 0$, $\nu \geq 0$, $\lambda_1$ , $\lambda_2 > 0$. In this thesis the term corresponding to the area inside $C$ will be ignored: $\nu = 0$. This is because the length term gives more information about the regularity of the curve $C$ – consider a region encompassed by a smooth curve vs. jagged curve, with the area remaining unchanged. Also, there is an inequality bounding the area from above by the length squared [5]. Consider a curve of length $L = 2\pi\, r$. A circle maximizes the area of an object with boundary of a fixed length, so we have that $A \leq \pi\, r^2 = \dfrac{L^2}{4\pi}$. Therefore, minimizing the length of the curve also provides some regularization for the area of the object bounded by it. Furthermore, there is no intuitive reason in our application to want to minimize the area of our object, but there is to minimize the length – indeed the eyeballs are circular (spherical in 3D, where the length of our segmentation becomes surface area) and with smooth boundary.

The choice for the rest of the parameters describes the balance of importance of each of the respective terms in the energy. In particular, for our application we set $\lambda_1 = \lambda_2$ for now, to indicate that being inaccurate (w.r.t. the piecewise constant approximation to $I_0$) inside and outside the object should be penalized equally. Looking at the length term, we can see that making $\mu$ large will force the evolving curve to stay as short as possible. An example of this is mentioned above: adding a few noisy pixels may seem as though we will incorrectly classify them as objects if we only kept the two terms $E_{\text{inside}(C)}$ and $E_{\text{outside}(C)}$, but with this regularization, the curve will be discouraged from splitting up in such a way. On the other hand, if we expect our image to have many small objects, we want to avoid grouping clusters of nearby objects together, so we will have to choose a small value for $\mu$.

To build the Chan and Vese energy $E_{CV}$, we start with the simplest example where we have an image $I_0$ with two distinct intensity values, $I_0^o$ and $I_0^i$. This, of course, does not happen naturally, so we now consider how the basic idea generalizes to more typical grayscale

images. This borrows the ideas from Mumford and Shah's functional for segmentation [17]. The main goal here is to find the best piece-wise constant approximation to an image $I_0$, such that each region of constant intensity represents one object (or background). Let this approximation be $I_C$, where the subscript $C$ corresponds to the curve defining the boundary of separation between these regions. We can define this more precisely for any curve $C$, and we aim to find the best curve (as $C_0$ above) to make our approximation $I_C$ match $I_0$ best. Let $c_1$ be the average intensity of the pixels in the image $I_0$ located inside the region defined by $C$, and $c_2$ be the average intensity of $I_0$ outside the curve $C$. Then the definition of the piecewise constant approximation of $I_0$ defined by $C$ becomes:

$$I_C = \begin{cases} c_1, \text{ for pixels inside C,} \\ c_2, \text{ outside C.} \end{cases} \tag{2.8}$$

### 2.2.3 Full Energy

With this description of the general grayscale image problem, the generalization of the simple case is natural. So, our energy $E_{CV}$ remains unchanged, and we have yet to explicitly state the two terms $\mu \cdot \text{Length}(C)$ and $\nu \cdot \text{Area(inside}(C))$. In order to do that, two support functions are introduced. First, recall how our curve $C$ is defined:

$$\begin{aligned} \text{inside}(C) &= \{(x,y) \in \Omega \mid \phi(x,y) > 0\} \\ \text{outside}(C) &= \{(x,y) \in \Omega \mid \phi(x,y) < 0\} \\ C &= \{(x,y) \in \Omega \mid \phi(x,y) = 0\} \end{aligned}$$

In order to track the curve $C$ as it evolves, we really need to track $\phi(x,y)$ and refer to its zero level set to extract $C$. With this in mind, we wish to specify the length and area terms in the energy, but also translate the remaining two integrals into a convenient form. For this reason we introduce two support functions: the Heaviside function, $H(s)$, and the delta function, $\delta(s)$:

$$H(s) = \begin{cases} 1 \text{ if s} \geq 0, \\ 0 \text{ if s} < 0\ . \end{cases}$$

We define $\delta(s) = \frac{d}{ds}H(s)$ : the derivative of the Heaviside function, in the sense of distributions. This provides a simple way to represent the regions inside($C$) and outside($C$). From the definition of $C$ and from Figure 1.2, we can see that these regions are equivalent to $H(\phi(x,y))$ and $(1-H(\phi(x,y)))$, respectively. This helps us with the last two integrals in the energy $E_{CV}$: we can now integrate over the whole domain $\Omega$ by multiplying each of the integrands by the respective Heaviside function terms. For the first two terms, length and area, we now have explicit expressions:

$$\begin{aligned} \text{Length}(C) &= \int_\Omega |\nabla H(\phi(x,y))| dxdy \\ &= \int_\Omega \delta(\phi(x,y))|\nabla\phi(x,y)| dxdy \end{aligned} \tag{2.9}$$

and

$$\text{Area}(C) = \int_\Omega H(\phi(x,y)) dxdy \tag{2.10}$$

The length term arises from noticing the fact that $H(\phi(x,y))$ only changes (has nonzero gradient) across the boundary of the curve $C$ – when $\phi$ goes from negative to positive. The area term can be easily established from Figure 1.2: the area inside($C$) is precisely the area where $\phi > 0$. Putting everything together, we now have a complete expression for the energy:

$$\begin{aligned} E_{CV}(c_1,c_2,\phi) = \mu &\int_\Omega \delta(\phi(x,y))|\nabla\phi(x,y)| dxdy \\ &+ \nu \int_\Omega H(\phi(x,y)) dxdy \\ &+ \lambda_1 \int_\Omega |I_0(x,y) - c_1|^2 H(\phi(x,y)) dxdy \\ &+ \lambda_2 \int_\Omega |I_0(x,y) - c_2|^2 (1 - H(\phi(x,y))) dxdy \end{aligned} \tag{2.11}$$

Now that the full model is developed, it can be transformed into a numerical problem and solved using finite differences or any other robust discretization method. Before presenting the numerical scheme used here, we first discuss the existence of a minimum for the

energy (2.11). Intuitively, we can refer to the simplest version of the model: for a simple image of two distinct intensities $(I_0^o, I_0^i)$, with $\mu = \nu = 0$, we find that $\phi$ defined so that it is 0 along the boundary of the single object makes the energy 0, giving us the desired minimum. In the more general case where the image $I_0$ is not necessarily piecewise constant, and the length and area terms are included $(\mu, \nu \neq 0)$ there are rigorous proofs available to assert existence [16], [14]. As well as formal treatment, this includes an extension to approximated versions of these types of problems [14].

In [16], for example, a proof is provided for the existence of a minimum of a closely related functional:

$$E(u,B) = \|u-g\|_{L^2(R)} + l(B) \tag{2.12}$$

The set-up of the problem is as follows:

- $R$ is a rectangle – this is satisfied as $\Omega$, our image domain, is indeed a rectangle.
- $B$ is a finite set of curves in $C^1$, defined on $R$ – these are the possible curves defined by the zero level set of our segmentation $\phi$.
- $u$ is piecewise constant on $R - B$ – this is equivalent to the piecewise constant approximation of $I_0$ in the Mumford and Shah formulation of the segmentation problem. This has been referred to as $I_C$ in (2.8).
- $g$ is a bounded function on $R$ – our image $I_0$
- $l(B)$ here is equivalent to a regularization of the curve (the zero level set of $\phi$). In particular, the case when $l(B)$ is the length of $B$ is covered in [16].

As well as the above, Mumford and Shah also provide a proof of existence [17]. We can now proceed to look at a numerical version of the problem.

## 2.3 Euler-Lagrange formulation and Numerical Scheme

### 2.3.1 Heaviside function approximation

In order to minimize the energy (2.11), we first find the Euler-Lagrange equations associated with it and then discretize them accordingly. Before proceeding with this, a special treatment of the Heaviside function and the delta function is necessary – we need a regularized version of these in order to be able to set up a scheme. To maintain the regularized delta function in $C^1$ (with one continuous derivative), for example, we require that the Heaviside function regularization stays in $C^2$. One example of a regularized version of $H(s)$, with small parameter $\varepsilon$, is proposed in [24] as follows:

$$H_\varepsilon^1(s) = \begin{cases} 1 \text{ if } s > \varepsilon\,, \\ 0 \text{ if } s < -\varepsilon\,, \\ \frac{1}{2}\left[1 + \frac{s}{\varepsilon} + \frac{1}{\pi}\sin\left(\frac{\pi s}{\varepsilon}\right)\right] \quad \text{if } |s| \le \varepsilon. \end{cases} \tag{2.13}$$

Then the associated approximation to the delta function, $\delta_\varepsilon^1(s) = \dfrac{dH_\varepsilon^1(s)}{ds}$, becomes:

$$\delta_\varepsilon^1(s) = \begin{cases} 0 \text{ if } |s| > \varepsilon\,, \\ \frac{1}{2\varepsilon}\left[1 + \cos\left(\frac{\pi s}{\varepsilon}\right)\right] \quad \text{if } |s| \le \varepsilon. \end{cases} \tag{2.14}$$

Chan and Vese use a different approximation, which is a $C^\infty(\Omega)$ regularization:

$$\begin{aligned} H_\varepsilon^2(s) &= \frac{1}{2}\left[1 + \frac{2}{\pi}\arctan\left(\frac{s}{\varepsilon}\right)\right], \text{ and} \\ \delta_\varepsilon^2(s) &= \frac{1}{\pi\varepsilon}\,\frac{\varepsilon^2}{\varepsilon^2 + s^2}\,. \end{aligned} \tag{2.15}$$

Both of these approximations are valid, as they both have the property that as $\varepsilon \to 0$, $H_\varepsilon(s) \to H(s)$. These are illustrated in Figure 2.4. The parameter $\varepsilon$ describes the "amount" of regularization applied, and choosing it to match the spatial discretization cell size, $\Delta x$, means that the support of the regularized functions is on the same order as our discretization – ie. the effect of applying either one of these functions to $\phi$ is reflected on just a few pixels close to its 0 level set. Since our discretization is naturally provided by the pixelation

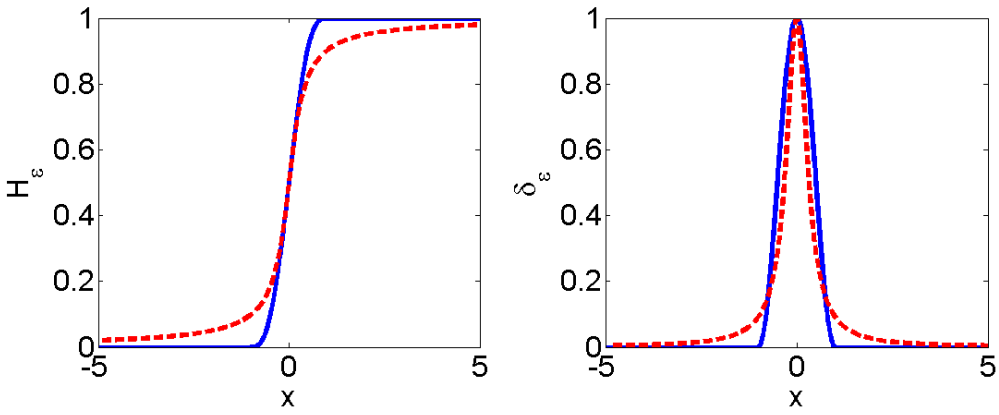


Figure 2.4: Comparing two regularizations of the Heaviside and respective delta functions

of the image, we take $\varepsilon = \Delta x = 1$. For the purposes of comparison, in the figure we have used $\varepsilon = 1$ for the first definition of the delta function, and $\varepsilon = \frac{1}{\pi}$ for the second – this is simply a scaling factor between the two versions.

To see why the second choice of regularization is best for us, we consider the differences between the two, with the nature of our equations in mind. As will be shown below, the Euler-Lagrange equations have a $\delta_\varepsilon(\phi)$ dependence, which causes our update equations to only apply in regions where our regularization has support. We can see that the first option for regularizing the Heaviside function results in the delta function being nonzero only on the interval $[-\varepsilon, \varepsilon]$ (2.14). This means that only the level set curves in a neighbourhood around the zero level set of $\phi$ will be evolved. Although this may sound computationally desirable, it has one major drawback, forcing us to choose (2.15): evolving only a few level sets can lead to a local minimum of our energy. The $C^\infty$ regularization allows us to evolve our equation on all level set curves simultaneously, as $\delta_\varepsilon^2$ is never 0, and this makes the method more flexible and more likely to find a global minimum. Therefore, this has the benefit of making our method insensitive to initial conditions and capable of automatically detecting interior contours. These are desirable features for any segmentation algorithm.

### 2.3.2 Euler-Lagrange Equations

With this choice of regularization, we will now drop the subscript and use $H_\varepsilon, \delta_\varepsilon$ as defined in equations (2.15). Now the Euler-Lagrange equations can be derived using these functions. Recall the full energy as before:

$$\begin{aligned} E_{CV}(c_1,c_2,\phi) = \mu &\int_\Omega \delta(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ &+ \nu \int_\Omega H(\phi(x,y))dxdy \\ &+ \lambda_1 \int_\Omega |I_0(x,y)-c_1|^2 H(\phi(x,y))dxdy \\ &+ \lambda_2 \int_\Omega |I_0(x,y)-c_2|^2 (1-H(\phi(x,y)))dxdy \end{aligned} \tag{2.16}$$

To obtain an appropriate update equation for $c_1$ and $c_2$ we hold $\phi$ fixed, we minimize $E_{CV}$ with respect to each of $c_1$ and $c_2$ by differentiating it appropriately and setting the derivative to 0. We show the details of this computation for $c_1$ and simply state the resulting equation for $c_2$ and the Euler-Lagrange equation for $\phi(x,y,t)$ (recall $t$ is an artificial time for the purposes of evolving the segmentation curve, and we may drop the absolute value as the terms are squared):

$$\begin{aligned} \frac{\partial E_{CV}}{\partial c_1} &= -\lambda_1 \int_\Omega 2\,(I_0 - c_1)\,H(\phi)\,dxdy = 0 \\ \int_\Omega I_0\,H(\phi)\,dxdy - c_1 &\int_\Omega H(\phi)\,dxdy = 0 \text{ , by absorbing all constants into the RHS} \\ \int_\Omega I_0\,H(\phi)\,dxdy &= c_1 \int_\Omega H(\phi)\,dxdy \\ c_1 &= \frac{\int_\Omega I_0\,H(\phi)\,dxdy}{\int_\Omega H(\phi)\,dxdy} \end{aligned} \tag{2.17}$$

This gives a nice expression for $c_1$ in terms of $\phi$. In a similar fashion, we obtain expressions for $c_2$ and $\phi$, with natural boundary conditions arising from the energy minimization, and a predefined initial condition:

$$c_2 = \frac{\int_\Omega I_0\,(1-H(\phi))\,dxdy}{\int_\Omega (1-H(\phi))\,dxdy} \tag{2.18}$$

$$\frac{\partial \phi}{\partial t} = \delta_\varepsilon(\phi)\ \left[\mu\ \mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \nu - \lambda_1\ (I_0 - c_1)^2 + \lambda_2\ (I_0 - c_2)^2\right]\ , t > 0$$

$$\phi(x,y,0) = \phi_0(x,y)\ , \text{in } \Omega$$

$$\frac{\delta_\varepsilon(\phi)}{|\nabla\phi|}\ \frac{\partial\phi}{\partial\vec{\eta}} = 0\ \ , \text{on the boundary of } \Omega \qquad (2.19)$$

Here $\vec{\eta}$ denotes the exterior normal to the boundary of our domain, so that the term $\frac{\partial\phi}{\partial\vec{\eta}}$ is the normal derivative of $\phi$ at the boundary. This is the PDE we will solve numerically over the image domain to obtain the segmentation defined by the stationary state of $\phi$.

### 2.3.3 Numerical Scheme

We now consider how to discretize the above equations. It is important to keep in mind that although this discussion is in 2D, we wish to generalize to 3D and we must consider speed of computations as a significant factor in designing a numerical scheme. First, we denote by $\Delta x$ the spatial discretization step, both in the x and y directions. We take this to be naturally dictated by the image pixelation as 1 – the "length" of a single pixel. Similarly, denote by $\Delta t$ the artificial time step. Let the size of the image be $m$ pixels by $n$ pixels, $i$ be the index along the x direction and $j$ along the y, so that $I_{0,i,j}$ denotes the intensity of the image $I_0$ at position $x = i, y = j$. Similarly, we index time by superscript $k$.

For the time derivative we use a simple forward Euler discretization. Although this is not the best choice for 2D in terms of accuracy and stability, the purpose of this thesis is to develop an approach for segmentation which works in 3D. For this reason, we focus on the simplest approach to perform our experiments, and any improvements made to the time and space derivative discretization in the future are encouraged. Also, this is in fictitious time and we are looking for steady states, so the temporal accuracy is not important. Thus, we approximate the LHS of the equation (2.19) as:

$$\frac{\partial \phi(x,y,t)}{\partial t} \approx \frac{\phi_{i,j}^{k+1} - \phi_{i,j}^{k}}{\Delta t} \tag{2.20}$$

Next, we consider how to discretize the divergence term by first writing it as:

$$\text{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) = \frac{\partial}{\partial x}\left(\frac{\phi_x}{(\phi_x^2+\phi_y^2)^{\frac{1}{2}}}\right) + \frac{\partial}{\partial y}\left(\frac{\phi_y}{(\phi_x^2+\phi_y^2)^{\frac{1}{2}}}\right) \tag{2.21}$$

We will discretize the above piece-by-piece, by first introducing the following notation:

- $\Delta_-^x \phi_{i,j} = \phi_{i,j} - \phi_{i-1,j}$, a backward difference
- $\Delta_+^x \phi_{i,j} = \phi_{i+1,j} - \phi_{i,j}$, a forward difference
- $\Delta_-^y \phi_{i,j} = \phi_{i,j} - \phi_{i,j-1}$
- $\Delta_+^y \phi_{i,j} = \phi_{i,j+1} - \phi_{i,j}$

Notice that each of these are first derivative approximations, as our $\Delta x = 1$, and that $\Delta_-^x(\Delta_+^x \phi_{i,j})$ results in a centered difference estimating the second x derivative of $\phi$ at the point $x = i, y = j$. Examining the divergence equation, if we only had $\text{div}(\nabla\phi)$, we could replace the term $\frac{\partial}{\partial x}\phi_x$ with $\Delta_-^x(\Delta_+^x \phi_{i,j})$, to obtain a centered difference. As the problem that we are solving is parabolic, for the divergence term it is most suitable to stick to centered differences. Due to the symmetry of equation (2.21) of interchanging x to y, let us only consider the first half of it. Since we now have a term in the denominator, it is most straightforward to replace all x derivatives inside the brackets by $\Delta_+^x$, so that combined with the $\Delta_-^x$ in front of the bracket we will get centered differences. However, to deal with the $\phi_y$ term, we must immediately use a centered difference, since the partial outside the bracket is with respect to x, not y, so that $\phi_y(i,j) \approx \frac{1}{2}(\phi_{i,j+1} - \phi_{i,j-1})$ in the first half of the divergence term. Combining these arguments, we get the following approximation to the first portion of the divergence term:

$$\frac{\partial}{\partial x}\left(\frac{\phi_x}{(\phi_x^2+\phi_y^2)^{\frac{1}{2}}}\right) \approx \Delta_-^x\left(\frac{\Delta_+^x \phi_{i,j}}{\left((\Delta_+^x \phi_{i,j})^2 + \frac{1}{4}(\phi_{i,j+1} - \phi_{i,j-1})^2\right)^{\frac{1}{2}}}\right) \tag{2.22}$$

By the symmetry of the divergence term we have, we can do the same for the y-portion: replace the outer partial derivative with a backward difference, all the inside y-derivatives with forward differences, and use a centered difference for the x derivative in the denominator. All the rest of the terms in the full equation for $\phi$ (2.19) follow naturally, to give the full discrete version of the equation as:

$$\begin{aligned}\frac{\phi_{i,j}^{k+1}-\phi_{i,j}^{k}}{\Delta t} = \delta_\varepsilon(\phi_{i,j}^k)\cdot\Bigg[&\mu\,\Delta_-^x\cdot\left(\frac{\Delta_+^x\phi_{i,j}^k}{\left((\Delta_+^x\phi_{i,j}^k)^2+\frac{1}{4}(\phi_{i,j+1}^k-\phi_{i,j-1}^k)^2\right)^{\frac{1}{2}}}\right)\\&+\mu\,\Delta_-^y\cdot\left(\frac{\Delta_+^y\phi_{i,j}^k}{\left(\frac{1}{4}(\phi_{i+1,j}^k-\phi_{i-1,j}^k)^2+(\Delta_+^y\phi_{i,j}^k)^2\right)^{\frac{1}{2}}}\right)\\&-\nu-\lambda_1\left(I_{0,i,j}-c_1(\phi^k)\right)^2+\lambda_2\left(I_{0,i,j}-c_2(\phi^k)\right)^2\Bigg]\end{aligned} \tag{2.23}$$

This discretization of the divergence term is proposed in [19] with some minor differences. However, Chan and Vese use an implicit scheme: they replace the terms in the numerator of the fractions with their respectives at time step k+1. We cannot afford to use an implicit scheme, as pointed out in the start of this subsection, as we will be generalizing this scheme for 3D. From the experiments that follow, the fully explicit version of this scheme used here performs well and in reasonable time (seconds in 2D, minutes in 3D). There are some stability issues that arise, and care needs to be taken with parameter choice. Notice that any restrictions imposed by the forward Euler time discretization are further affected by our choice for the parameter $\mu$. Indeed, our choice for $\mu$ varies from 1 to 1000, and if the image at hand requires a large $\mu$ we are forced to lower our $\Delta t$ to retain stability. This is mostly done by experimenting with various parameter values and by visual inspection of the algorithm's step-by-step progress. A more detailed discussion of parameter choice is presented in the following section along with some segmentation examples.

The natural boundary condition in (2.19) is implemented at each iteration by forcing the values of the outer-most pixels to be the same as their immediate neighbours: the left-

most column of pixels take on the same values as the column immediately next to it, and the top row copies the values of the second row. Similarly for the right and bottom pieces of the boundary of the image domain. This is done immediately following the update for $\phi$, as in Step 2 below. We now proceed to formulate the iterative method for evolving $\phi$ numerically, as used by Chan and Vese:

- Initialization: Define an initial guess for the segmentation, $\phi^0$, defined over the whole image domain.
- Step 1: Compute $c_1(\phi^k)$ and $c_2(\phi^k)$ as defined in equations (2.17) and (2.18). We replace integration with summation over all the pixels in the domain.
- Step 2: Compute $\phi^{k+1}$ for the whole domain using equation (2.23) and apply boundary conditions. Note that this equation can be easily rearranged to isolate $\phi^{k+1}$ on the LHS and the remaining terms are already computed for the current time level. Also note that computing $\delta_\varepsilon(\phi^k_{i,j})$ is simply done by plugging the value of $\phi$ at the respective pixel at coordinates (i,j) into our approximation to the delta function; similarly for the Heaviside function when computing $c_1$ and $c_2$.
- Repeat Step 1 and 2 for next time level until the desired segmentation is achieved.

Since we are aiming to arrive at a stationary point of the above procedure, a possible stopping condition is to look at some norm of $\phi_t$ and compare it to some small threshold value close to 0. This, however, can be unreliable depending on the particular image we have and the result we are looking for. This stopping criterion would be appropriate in the continuum but does not necessarily apply for our application – our problem is inherently discrete and it is difficult to estimate a reliable threshold which would hold for all images. Since we are dealing with pixels, the "final" position of $\phi$'s 0 level set may oscillate some distance of order 1 – our discretization size. For this thesis, the above iterative procedure is terminated after a preset number of iterations, determined by visual inspection and experiments. By running the procedure on a training set of images, we are able to estimate the total number of iterations necessary to reach our goal segmentation. At this stage of the method developed in this thesis, computations in 2D tend to take seconds, so this is feasible. However, a better stopping criterion will be greatly beneficial in the future to facilitate

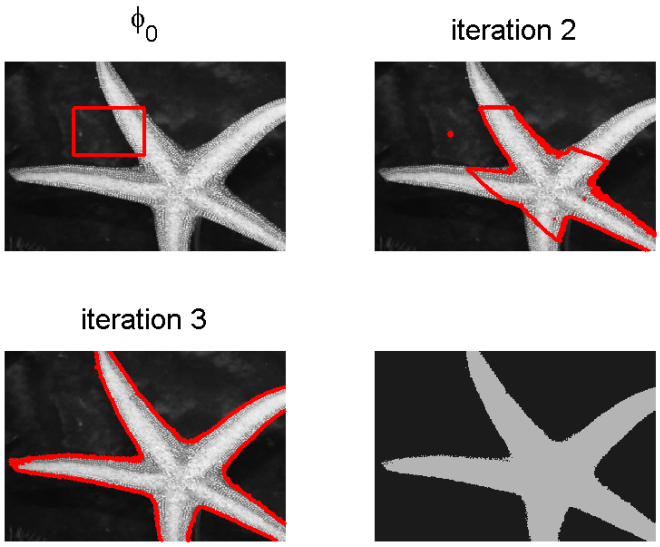


Figure 2.5: An intermediate step in Chan and Vese's segmentation method

the automated processing of large data sets.

Another point made by Chan and Vese is the possible need to reinitialize $\phi$ every few iterations. This arises from the use of the delta function in the update equation for $\phi$ – the nature of the delta function can cause $\phi$ to become flattened, and this can translate to losing important information about the current state of the segmentation procedure. One option is to reset $\phi$ to a signed distance function from its zero level set. This may not be necessary depending on the application and number of iterations necessary to arrive at a satisfactory solution, and reinitializing too often can prevent interior contours from developing (ie. a brand new zero level set from arising, as opposed to the "old" one simply moving along). For now, we will not consider reinitializing. We now proceed to some examples.

## 2.4 Segmentation Examples

To demonstrate how the curve is guided by the above procedure, we consider a simple example in Figure 2.5 (photograph provided by photographer and digital artist Krystal Shea). In the first picture, we superimpose the zero level set of the initialization, $\phi_0$, followed by the second and third iterations of the method. The final picture shows the piecewise

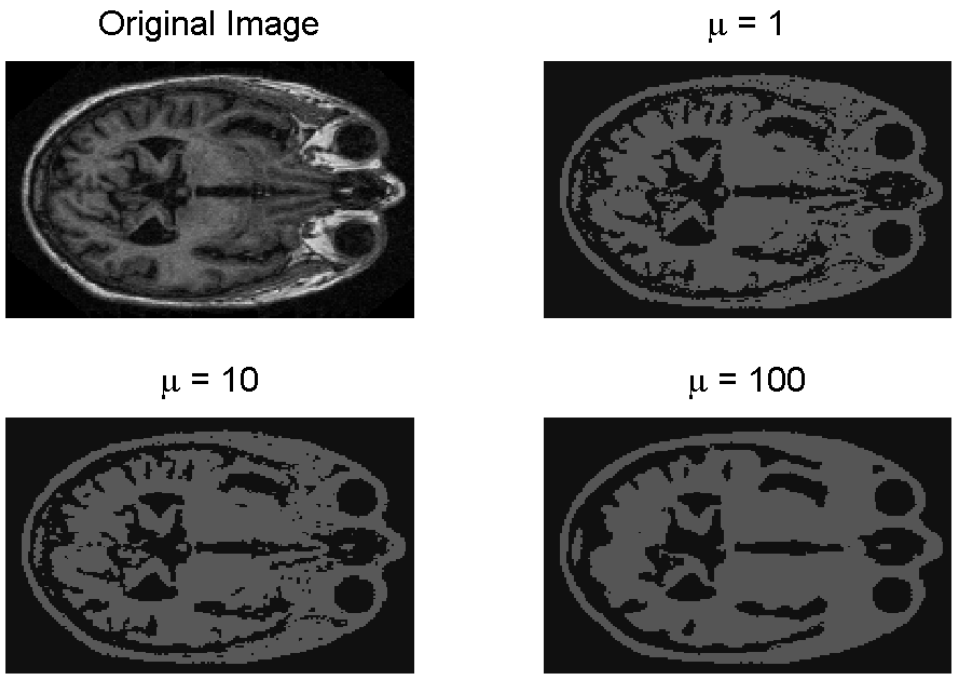


Figure 2.6: Exploring the parameter $\mu$ in a 2D brain image

constant approximation to the original image resulting from our segmentation. For this experiment the image size is 395 by 535 pixels, we take $\varepsilon = 1$ (equal to the spatial discretization step size), $\mu = 100$, $\lambda = 1$ and $\Delta t = 0.01$. For this simple image only 3 iterations are sufficient to arrive at the final segmentation, and the computation time is approximately 1 second.

The choice of parameters was determined through experiments, but the intuition behind these values is also important: the larger $\mu$ is, the more our energy is penalized for a large length of the curve described by the zero level set of $\phi$ – this means that the method favours larger objects and hence bypasses small features that may be present in the starfish, or noise in the background. The value of $\lambda$ being 1 results from the fact that we essentially only require two parameters to scale the problem (ignoring $\varepsilon$): $\mu$ and $\Delta t$. The time step value could have been chosen to be larger, but that would cause the method to find the correct segmentation in one step; so, for illustration purposes we have set it to 0.01.

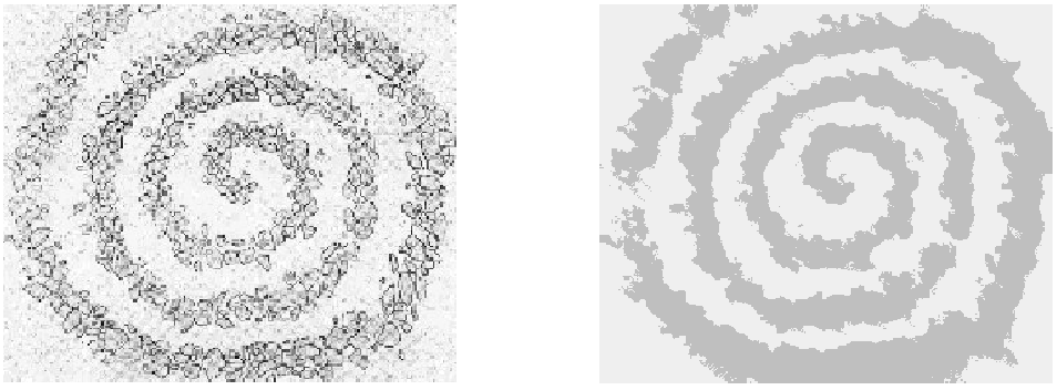

Figure 2.7: Segmenting a spiral with boundaries not defined by gradient, with $\mu = 1000$

To further explore the various options for selecting the parameter $\mu$, we show an example of the same image of a 2D slice of a brain MRI image segmented using three different values for $\mu$: 1,10, and 100, shown in Figure 2.6. The size of the image is 128 by 201 pixels and we use $\varepsilon = 1$ (from now on this will be the choice for $\varepsilon$, unless otherwise stated). In this case we set $\lambda$ to 0.1 in order to keep $\Delta t$ at 0.01 and maintain the desired numerical stability. We require 10 iterations to arrive at each segmentation, with approximately 20 seconds of computation time for each case. We can see that the case with $\mu = 1$ gives the most detail sensitive segmentation, while $\mu = 100$ gives a more "rough" approximation to what is the object and what is the background in the image. The middle case with $\mu = 10$ illustrates the continuum of segmentations available across all values of $\mu$ in between. This example illustrates that it is important for the segmentation procedure to have some previous knowledge about the specific application at hand: we need to combine experiments with different parameter values and knowledge of the nature of the object we are looking for.

Now we consider the flexibility of this algorithm to handle images with objects whose boundaries are not defined by gradient but by cognitive boundaries (ones humans can see, but computers have a hard time recognizing). The example is Figure 2.7: an image tested by Chan and Vese of a spiral constructed by chunks of darker intensities, on a background of lighter chunks. The domain size is 289 by 329 pixels with $\mu = 1000$ – a large value compared to experiments with other images – selected in hopes of discouraging our curve from breaking up to encompass each of the small chunks of dark intensity pixels, and encourage

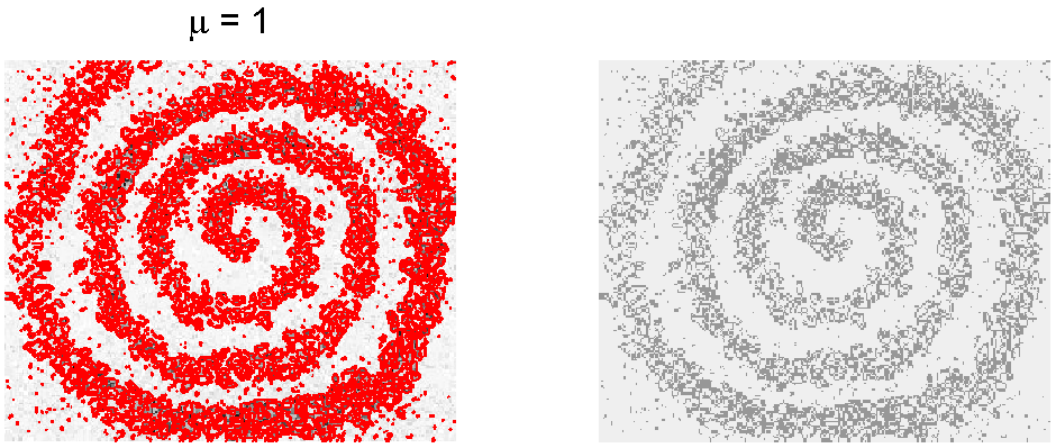


Figure 2.8: Demonstrating the importance of choosing an appropriate value for $\mu$: spiral image segmentation with $\mu = 1$

it to rather stay connected and further bunch all of these chunks together. The first window of the figure shows the original image. We have not superimposed the zero level set of $\phi$ for visual clarity, but the second window shows the piecewise constant approximation to the image defined by our $\phi$. In contrast, Figure 2.8 shows what happens if we instead set $\mu = 1$: now the energy prefers to optimize the piecewise constant approximation and allow the length of the curve to grow by allowing many splits. For both of these we have set $\lambda$ = 0.5 and $\Delta t$ to 0.001. For the first case we require 55 iterations (120 seconds), and we only run the second case for 15 iterations to demonstrate the difference between large and small $\mu$.

Now we consider a blurred version of the same image to ensure we obtain (almost) the same segmentation, and to provide another example of intermediate steps of the procedure. This image is the same size, and we use the same parameter values ($\mu = 1000$): see Figure 2.9.

There is one important issue with Chan and Vese's method. It is important to understand that the method is designed to separate the image domain into exactly two distinct objects, solely based on intensity. So, if we have an image with two entirely separate objects but which happen to have similar intensities, different enough from the background intensity,

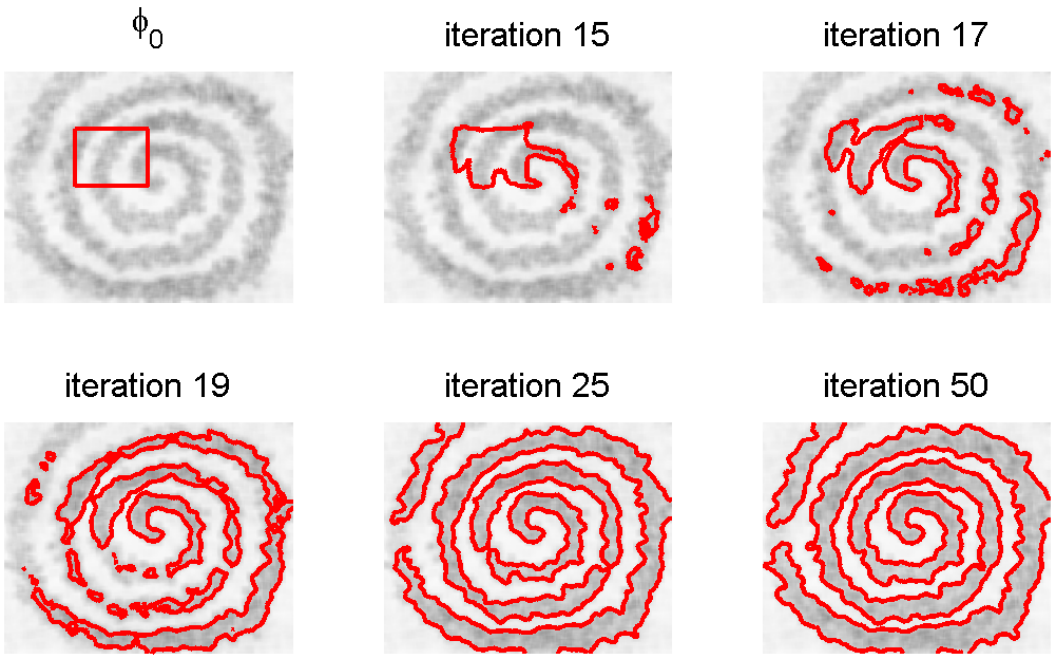


Figure 2.9: Intermediate steps shown for blurred spiral image

Chan and Vese's method will group those two as a single object. Two examples of this issue are shown below. In the first, we have three geometrical shapes and a background: shown in Figure 2.10. What we may first expect to see is all three shapes grouped together as one – this will be most people's intuition, to separate the objects from the background. However, the computer will see it a different way. What is going on here is that the intensity of the partial rectangle is much closer to that of the background than to the circle and triangle. This results in a greater benefit for the rectangle to be grouped together with the background. More precisely, the intensity of the background in the original image is 156, while the intensity of the background in the resulting piecewise constant approximation is 159 – they are nearly the same, indicating that our energy was not particularly affected by merging the white rectangle in with the background. This issue can be easily remedied by the multiphase approach of evolving more than one segmentation function $\phi$ [6], so that we have an opportunity to further minimize the energy. Another example of this issue is shown in Figure 2.11 (photograph by Krystal Shea): the silhouette of a woman in the mirror is grouped together with the frame of the mirror, for similar reasons – their intensities are very similar. It is important to understand the difference between what humans consider an

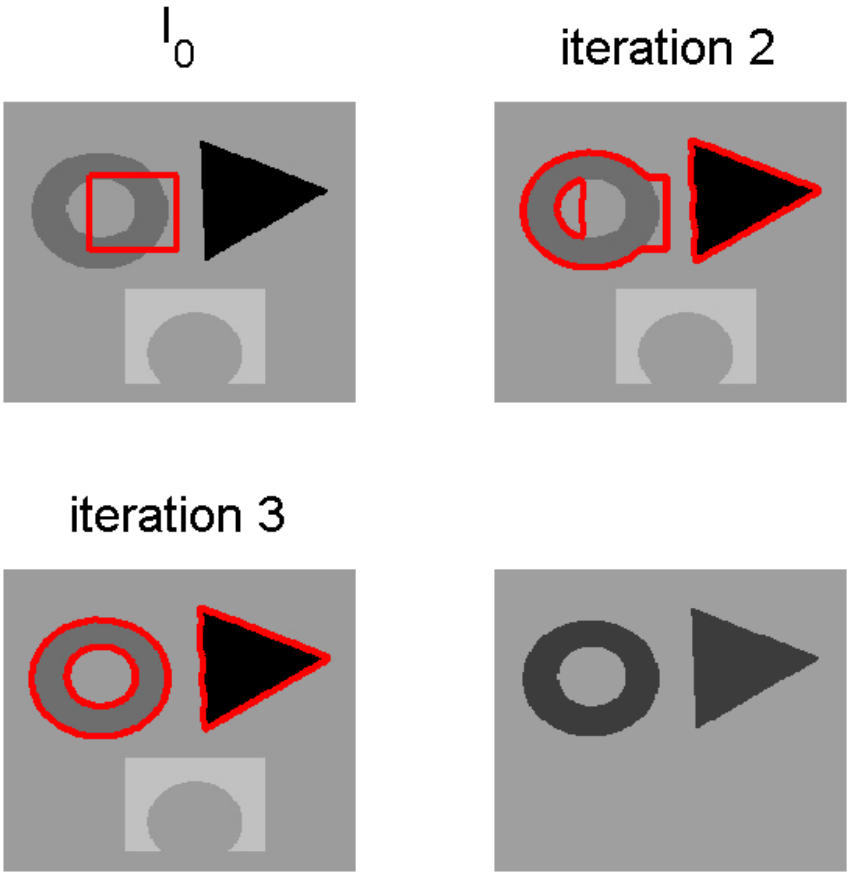


Figure 2.10: Example of limitations to using one curve – forcing the method to find only a single object as defined by intensity and "miss" an object

object and what the method deems an object.

These examples give a better idea for the motivation behind the main approach of this thesis. If we consider our ultimate goal – to find the eyeballs in a 3D brain image – we can see that we cannot simply apply the Chan and Vese algorithm to the whole volume. This would result in finding a large number of features, either separated or grouped together, due to the complex nature of a human brain. Instead, we must find a way to narrow down our search to a very small region, preferably containing no other objects or features aside from the eyeballs. Once we have this smaller section of our image we can more confidently proceed with Chan and Vese's segmentation procedure to accurately extract the eyeballs

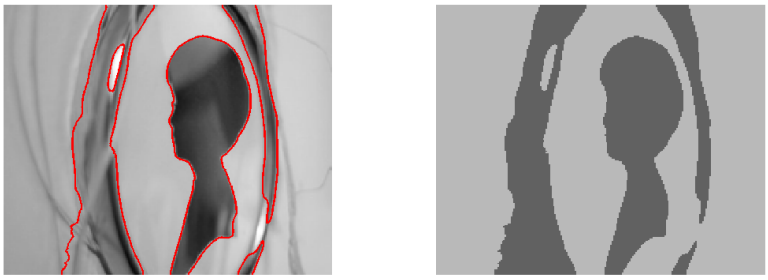

Figure 2.11: Another example of what the method considers to be one object, disagreeing with human intuition

from this isolated piece in the brain image. The next task becomes: how do we find this piece of the brain? The answer lies in using prior knowledge of human physiology, and in the particular case of the eyeballs – knowing the basic shape of a human eyeball. The next chapter explores how using this knowledge can formally lead us to the correct region and provide a good starting point for running the final segmentation procedure.

# Chapter 3

# Shape Priors

## 3.1 Introduction to Shapes

### 3.1.1 Motivation

The complex nature of the segmentation problem requires the use of high level information to guide it to success and shape priors offer just that. Our application is looking for the eyeballs in the human brain – this in itself builds an expectation for our object to be of a particular shape, so incorporating this information into the segmentation procedure is natural. Another example of how shape priors can be useful is facial recognition using feature templates. Here one could consider the specific shape of some important feature in a person's face (eg. mouth) and can use this to search through a database of face images for a specific person, by matching this precise shape (mouth) to the best candidate.

Shape priors can be used in other areas of image processing, aside from segmentation – for example, in image registration. This is the process of aligning a set of related images in a consistent manner. In fact, this thesis uses brain images that are assumed to have already gone through this procedure. Registration typically involves matching up a set of individual points amongst all the images. For example, if we had a number of images of the same object, but cropped, resized and perhaps rotated differently, we may wish to standardize the position and orientation of the object in all of these. This can be done by manually selecting points in all of the images that correspond to the same physical aspect in the object – eg.

if it were a person, specifying the location of the person's left eye, right eye and nose in all images could be sufficient to design a transformation to straighten out the person in all images, crop the image to center the person, etc.

This is a tedious process and can be inaccurate in certain applications. A large number of registration techniques have been developed – intensity based, feature based, etc. A detailed overview of approaches is available in [4]. Much like segmentation, registration is a difficult problem that benefits from application specific information. Another example in the medical field is working with functional MRI images (fMRI). These are time series images typically used to detect activity in the human brain, for example, used to assess brain functionality or aid studies in behavioral sciences. Normally in such studies a subject is exposed to some baseline conditions, followed by activation conditions which are determined by the design of the study [8]. Then, the activity in the different parts of the brain can be observed through the time series MRI images taken in the process. The difficulty of dealing with these images comes from the fact that the process takes some time, and slight movements of the subject during image acquisition are unavoidable. This results in misalignment of the images at each time snapshot, and requires registration to have an accurate compilation of the subject's reaction. Using the registration method mentioned above, as well as intensity based registration can be less than satisfactory in practice – the first due to its manual and time consuming nature, and the second due to the variations in intensity distribution from image to image. In this situation shape priors can provide a more reliable and robust method for alignment. Locating a specific feature in the brain in all of the images, we can estimate the "amount" of misalignment (eg. angle of rotation, specific translation) and correct for it. That is, by estimating the transformation parameters for the same feature in two images, we hence estimate the transformation parameters for the entire images.

Introducing shape information to existing segmentation techniques, such as Chan and Vese's method, provides for more control and flexibility in the segmentation procedure. We can create methods with the efficiency and precision of Chan and Vese, while suppressing objects that are not similar to the shape prior we use. We can also allow for the search of all objects, but require the recognition of the specific shape we are looking for. Some

flexibility is introduced through a parameter that can be chosen to specify the importance of the shape similarity – ie. we can decide how much we want our object to be defined by intensity (Chan and Vese's method), and how much we require this object to, at the same time, stay true to the shape prior. This allows us to look for objects similar to a particular shape, with the freedom to specify various levels of "similarity," depending on the type of application we are working with.

### 3.1.2 Definition of Shape

To define a shape in mathematical terms, we adopt the representation used in [7]. Using a similar idea as the definition for our segmentation $\phi$ in Chapter 2, we define the shape of an object, $S$, as a function $\psi$ that is positive inside the object and negative outside. More specifically, we use a signed distance function here – the unique regularized solution to the following problem:

$$|\nabla\psi| = 1$$

$$\psi(x,y) \begin{cases} > 0 & (x,y) \in S \backslash \delta S \\ = 0 & (x,y) \in \delta S \\ < 0 & (x,y) \in \mathbf{R}^2 \backslash S. \end{cases} \tag{3.1}$$

The nature of the signed distance function is that it is close to 0 near the boundaries of the 0 level set (the boundary of the object), gets more and more positive towards the interior of the object, and more and more negative the farther away we get outside of the object. One way to interpret the use of a distance function intuitively is to compare it to a statistical representation of a shape. Consider a similar shape positioned somewhat overtop of our shape $\psi$. The probability of a point somewhere in the interior to coincide for both shapes is high compared to the probability of a point closer to the edge to coincide. For example, if we have a shape of an apple and wish to compare it to that of a slightly different apple, we expect the differences to be mostly evident around the boundaries of the apples – hence the probability of matching in that area will be low, and its proximity to the 0 level sets combined with the definition of the distance function reflect this property. Having the values of $\psi$ be positive and negative simply allows us to easily classify points as "inside"

Figure 3.1: Two objects with the same shape: translation, scaling, and rotation invariance

or "outside" of the object, which suggests the use of a *signed* distance function.

Now we consider how to compare two objects and determine if they have the same shape. We choose to specify that shape is scale, translation and rotation invariant. That is, magnifying an object, translating it in the image domain and/or changing its orientation should not be considered to change the shape of that object. More formally, we say that two objects have the same shape if their representation functions, say $\psi_1$ and $\psi_2$, can be related by the following equation:

$$\psi_2(x,y) = r\,\psi_1\left(\frac{(x-a)\cos(\theta)+(y-b)\sin(\theta)}{r}, \frac{-(x-a)\sin(\theta)+(y-b)\cos(\theta)}{r}\right) \quad (3.2)$$

Here $(a,b)$ represents the translation, $\theta$ the rotation angle and $r$ is the scaling factor, that need to be applied to $\psi_1$ to obtain $\psi_2$. An example of two objects with the same shape is shown in Figure 3.1. Here we take a simple object, translate it by 20 units to the right, 15 units down, rotate it by 30°, and rescale it by a factor of 1.2. Note that some cropping is done at each stage to maintain same size domains for both, for comparison purposes.

### 3.1.3 Obtaining a Useful Shape Prior

In practice, we need to ensure that incorporating a shape prior will make the segmentation procedure better, not worse. It is important to select an accurate shape representation of our target object. The simplest approach would be to manually segment a single image containing the object we are looking for, and extract a shape function $\psi$ using the segmentation curve as an outline for our object boundary, and then solving the signed distance function equation. Another approach, mentioned in [8], is to construct a statistical model for shape variation across a training set of images, using sets of points in each image that correspond to each other (ie. registration points.) These points are combined and a Gaussian (or other) model can be fit to them in order to extract the boundary of the desired object.

Another approach used in practice is to use a training set of images which already have the correct segmentation (eg. in terms of level sets) and take the average of the contours. This can be done in different ways, one of which is implemented in [8]: having contours $C_1, C_2, \cdots, C_n$ representing the same shape but in different position/size/rotation, we fix $C_1$, find the transformation parameters $(a, b, r, \theta)$ so that $C_1$ can be expressed in terms of $C_2$ in the "best" way – by maximizing the area of overlap and minimizing area of no overlap between the two objects. Denote this "best" approximation of $C_1$ with $C_2$ by $C_2^{new}$. Then the same procedure is done for the remaining curves, and the average is defined as $C_{average} = (C_1 + \sum_{j=2}^{n} C_j^{new})/n$. Furthermore, if the training set contains high variation in shapes, a clustering technique can be used to extract meaningful information (details in [8]). Other approaches for obtaining a good shape prior include looking at the Fourier coefficients of the contours, or principal component representations, etc.

In the case of this thesis, the shape prior is quite simple due to the simple anatomical structure of the eyeballs. We approximate the eyes by circles in 2D and spheres in 3D. In 2D we look at four images, manually measure the approximate radius of an eye in each image (by counting pixels), and take the average. Next, we again manually measure the approximate distance between the centers of the two eyes and average. Taking advantage of the fact that this data set is already aligned in a consistent manner (registered), we also average the pixel coordinates that would be a best location for placing the initial shape

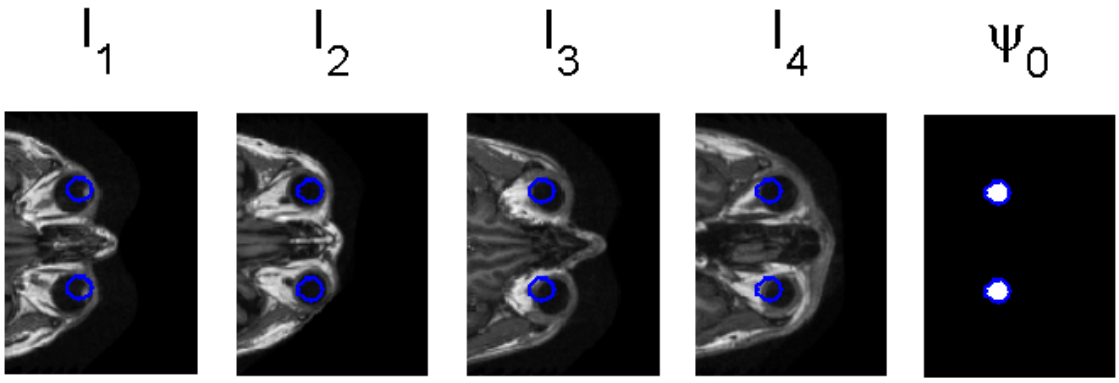


Figure 3.2: Designing a shape prior for 2D image of eyes: constructed manually using averaged data over a sample of images

prior in the images. Another benefit of the registration of the images we have is that we only need to worry about the translation parameters in shape representation – ie. we can assume no rescaling or rotation of our prior is necessary to match the shape of the eyes in all of the images. This idea will also be extended to 3D using spheres.

It is important to note that this approach is by no means robust and would likely cause problems if we look for more complex anatomical features, but is sufficient for now. The reason is that we will use a two stage segmentation procedure. Only the first stage relies on this shape prior information – we use it to narrow down the search region for the eyeballs, but we do not expect the final result of this stage to be an accurate segmentation on its own. The second stage applies the Chan and Vese procedure (without a shape prior) to this newly restricted region of the image domain. Figure 3.2 shows the shape prior which resulted from the above procedure. We can see that in some of the images it already provides what we are looking for ($I_2$, for example), and the purpose of this shape prior stage of the method is exactly to ensure we have estimated the location of the eyes to the best of our abilities. Once that is achieved, we can pass the end result to the Chan and Vese algorithm to obtain the final refined segmentation.

## 3.2 Incorporating Shape Priors into the Segmentation Process

We now consider different approaches for incorporating the information that a shape prior can provide into the segmentation procedure. The simplest approach is to minimize an energy involving only the shape prior, $\psi$, by translating it over the image domain. Most common approaches aim to add an energy term involving $\psi$ directly into the Chan and Vese energy, (2.11), and alter the update equations for all variables accordingly. A new idea described in [3] builds on the shape prior framework and incorporates an appearance prior to provide not only shape information, but appearance information into the overall segmentation procedure – this can involve specifying a particular intensity we expect the object we are looking for to have, position, orientation and more. First, let us consider the simplest scenario with a purely shape prior-driven method.

### 3.2.1 Translating Prior

For our purposes, recall that we are ignoring rescaling and rotation of our shape prior. Thus, we are looking for a particularly shaped object in our image, and we only aim to find its location by translating our shape prior in the $x$ and $y$ directions. Borrowing the ideas of Chan and Vese, we begin with a similar energy:

$$E_\psi = \lambda_1 \int_\Omega (I_0 - c_1)^2 H(\psi)\, dxdy + \lambda_2 \int_\Omega (I_0 - c_2)^2 (1 - H(\psi))\, dxdy \tag{3.3}$$

Here $c_1$ and $c_2$ are the average intensities of our image $I_0$ inside $\psi$ (ie. where $\psi > 0$), and outside, respectively. The parameters $\lambda_1$ and $\lambda_2$ determine the importance of having low variation in intensity of the image inside vs. outside of the 0-level set of $\psi$. We again employ the regularized version of the Heaviside function, $H_\varepsilon$, as before, but drop the subscript from here on. Notice we do not have a length minimizing term, as the length of the 0-level set of our prior is fixed in advance. The only way to evolve $\psi$ is to translate it in both the $x$ and $y$ directions. So, the end result for $\psi$ in our method will have the same shape as our starting prior, $\psi_0$. Using the definition of shape, we can write $\psi(x,y) = \psi_0(x-a, y-b)$, for some $a, b \in \mathbf{R}$. Replacing $\psi$ with this in equation (3.3) above, we can now minimize

$E_\psi$ with respect to $c_1$, $c_2$, $a$ and $b$. For the first two, we get equations similar to those in Chapter 2:

$$\begin{aligned} c_1 &= \frac{\int_\Omega I_0\, H(\psi)\, dxdy}{\int_\Omega H(\psi)\, dxdy} \\ c_2 &= \frac{\int_\Omega I_0\, (1-H(\psi))\, dxdy}{\int_\Omega (1-H(\psi))\, dxdy} \end{aligned} \tag{3.4}$$

Now to obtain an update equation for $\psi$, we minimize the energy for the translation values $a$ and $b$. The details for $a$ are shown here:

$$\begin{aligned} \frac{\partial E_\psi}{\partial a} &= -\int_\Omega \lambda_1 (I_0 - c_1)^2\, \delta(\psi_0(x-a, y-b))\, \psi_{0_x}(x-a, y-b)\, dxdy \\ &\quad + \int_\Omega \lambda_2 (I_0 - c_2)^2\, \delta(\psi_0(x-a, y-b))\, \psi_{0_x}(x-a, y-b)\, dxdy \\ &= \int_\Omega \left[-\lambda_1 (I_0 - c_1)^2 + \lambda_2 (I_0 - c_2)^2\right]\, \delta(\psi(x,y))\, \psi_x\, dxdy \end{aligned} \tag{3.5}$$

The corresponding equation for $b$ becomes:

$$\frac{\partial E_\psi}{\partial b} = \int_\Omega \left[-\lambda_1 (I_0 - c_1)^2 + \lambda_2 (I_0 - c_2)^2\right]\, \delta(\psi(x,y))\, \psi_y\, dxdy \tag{3.6}$$

We can now employ the usual gradient descent method for solving these equations iteratively until we reach a minimum of the energy, by evolving $\frac{\partial a}{\partial t} = -\partial_a E_\psi$ and $\frac{\partial b}{\partial t} = -\partial_b E_\psi$. The time derivatives are discretized using a first order forward Euler scheme for now (although this will be changed in the final model), and spatial derivatives using centered differences. To avoid function values growing and causing numerical instability, we may wish to re-initialize $\psi$ to a signed distance function at every iteration, or at every few iterations. For the examples in this section we do this at every iteration. The procedure for minimizing $E_\psi$ is as follows:

- Specify a starting shape prior, $\psi_0$, initialize $a = b = 0$
- Compute the values for $c_1$ and $c_2$ at the current time step, using equations (3.4)

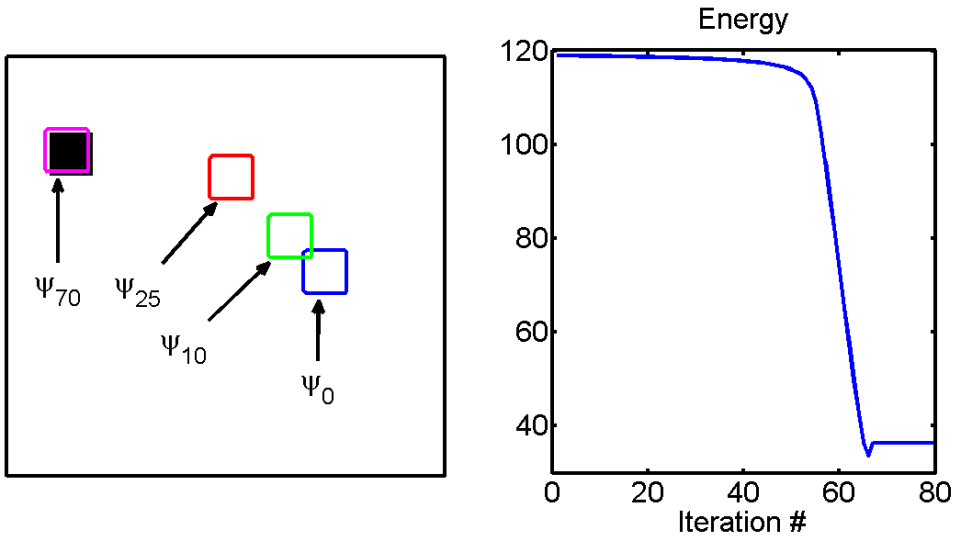


Figure 3.3: The translating prior method: evolution over a simple binary image

- Update the values for $a$ and $b$ using the corresponding discretization of equations (3.5) and (3.6)
- Transform the current shape prior as $\psi(x,y) = \psi_0(x-a, y-b)$
- Re-initialize $\psi$ to a signed distance function, by solving equation (3.1)
- Iterate the last three steps for the preset number of iterations determined by a training set of experiments

Although this is very similar to Chan and Vese's method, the key difference is the fact that with the active contours, we were able to evolve all of $\phi$'s level sets, which was important in working towards a global minimum without necessarily getting stuck at a local minimum. This also meant that the starting position for $\phi$ is not critical. However, with the shape prior, we are only moving a few pixels at a time, based on local information determined by intensities, and starting position can make a big difference. To be able to achieve our goal of roughly locating the eyes, we must be particularly careful to avoid getting stuck at one of the many undesirable local minima that brain images tend to have.

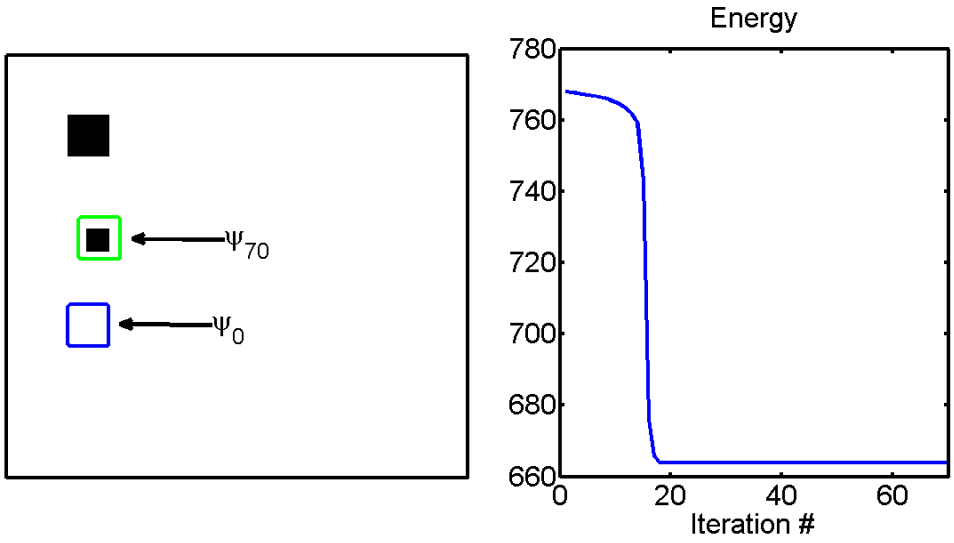


Figure 3.4: A local minimum in shape prior method (energy at global minimum $\approx 440$)

To illustrate how it works, we consider two simple cases. The first is shown in Figure 3.3 – we have one object (the black square), and we place the starting shape prior somewhere in the domain, with a shape matching the square perfectly. We let the method run for 80 iterations, using parameter values: $\Delta t = 0.1,\ \lambda_1 = 1,\ \lambda_2 = 2$. The reason for choosing $\lambda_2 > \lambda_1$ is that in most of the image domain, the value of $I_0 = c_1$ precisely and the first term in the energy is 0, so entering into the black square would cause this to change and hence $\psi$ is repelled away from the square. However, the second term is nonzero, and giving it more importance forces $\psi$ to move towards whatever is causing this inconsistency in intensities – ie. $\psi$ is attracted towards the square. It should be noted that the choice for the two parameters $\lambda_1$ and $\lambda_2$ is not particularly sensitive: we could use $\lambda_1 = 0.5,\ \lambda_2 = 1$, for example, or $\lambda_1 = 10,\ \lambda_2 = 100$, as long as we adjust $\Delta t$ accordingly, to maintain stability.

We can see that the method does what it is supposed to in this simple case. However, to illustrate the problem with getting stuck at local minima, we now add a smaller black square in close proximity to the original one, and place the starting shape prior such that it has to go through the small square in order to reach the larger one. We use the same parameter values as above. Figure 3.4 shows the result – as expect, we are stuck at the smaller square, as leaving it would increase the current energy.

For the reasons mentioned above, relying on an evolution guided only by the shape prior is less than satisfactory in most situations. To benefit from the advantages of Chan and Vese's model, we now consider adding a term to their energy that incorporates the shape prior, in hopes that we will once again have low sensitivity to initial conditions and decrease the likelihood of getting stuck at a local minimum.

### 3.2.2 Combining Chan and Vese's Method with Shape Priors

The idea here is to use the shape prior information, $\psi$, to enhance Chan and Vese's segmentation procedure. We begin with Chan and Vese's energy, omitting the length term for now, and consider adding a term which penalizes the segmentation $\phi$ for being different from the prior, $\psi$. The straight forward thing to do here is to have this term be of the form $\int_\Omega (\phi - \psi)^2 \, dxdy$, but this would only make sense if both $\phi$ and $\psi$ are signed distance functions. This condition is not required for $\phi$ and to avoid unnecessary computational expense, we must use a different term for comparing the two. We note that we only care about the region (area) that our object occupies and again employ the use of the Heaviside function in order to compare only the region where $\phi > 0$ with the region where $\psi > 0$ – in the optimal scenario, we would like these to coincide. In light of this, we construct the energy term $E_\psi = \int_\Omega (H(\phi) - H(\psi))^2 \, dxdy$ [7]. Including this into the existing Chan and Vese energy (without length term), we arrive at the following:

$$\begin{aligned} E_{CVSP} &= E_{CV} + E_\psi \\ &= \lambda_1 \int_\Omega (I_0 - c_1)^2 \, H(\phi) \, dxdy + \lambda_2 \int_\Omega (I_0 - c_2)^2 \, (1 - H(\phi)) \, dxdy \\ &+ \lambda_\psi \int_\Omega (H(\phi) - H(\psi))^2 \, dxdy \end{aligned} \tag{3.7}$$

This energy will allow for the evolution of both $\phi$ and $\psi$ in a coupled way, taking advantage of both Chan and Vese's idea about intensity-based segmentation, and combining it with the shape fidelity term to narrow down the search for objects. The parameter $\lambda_\psi$ can be chosen to represent the importance placed on the shape prior, whereas $\lambda_1$ and $\lambda_2$, as before, represent the importance of minimizing the within region pixel variation inside and

outside the object, respectively.

Recall that the only evolution permitted for $\psi$ is translation, with translation parameters $a$ and $b$. Once again we can replace $\psi(x,y)$ by $\psi_0(x-a, y-b)$ and minimize the above energy with respect to $c_1$, $c_2$, $a$, $b$, and $\phi$. Employing Euler-Lagrange and gradient descents, as usual, we arrive at equations for $c_1$ and $c_2$ as in Chapter 2, equations (2.17), and (2.18). For $a$ and $b$ we have:

$$\begin{aligned} \frac{\partial a}{\partial t} &= \int_\Omega (H(\psi) - H(\phi))\ \psi_x\ \delta(\psi)\ dxdy \\ \frac{\partial b}{\partial t} &= \int_\Omega (H(\psi) - H(\phi))\ \psi_y\ \delta(\psi)\ dxdy \end{aligned} \tag{3.8}$$

These provide for the evolution of $\psi$. The Euler-Lagrange equation for $\phi$ now becomes:

$$\frac{\partial \phi}{\partial t} = -\{\lambda_1\ (I_0 - c_1)^2 - \lambda_2\ (I_0 - c_2)^2 + 2\ \lambda_\psi\ (H(\phi) - H(\psi))\}\ \delta(\phi) \tag{3.9}$$

These equations are discretized in the same way as before. However, the nature of this energy is to simultaneously aim to segment the image based on intensities and attempt to keep the shape of the segmentation similar to the shape prior. This results in more iterations to achieve a stationary solution. However, as suggested in [10], there is a faster approach available for updating $\phi$. Recall that we only care about the 0-level set of $\phi$, and spending time on updating all values of $\phi$ can become inefficient. The information essential to us is only the sign of $\phi$ – this will give us all the information necessary to locate the 0-crossing. To this effect we also consider the following update equation for $\phi$ instead of 3.9:

$$\phi = \text{sign}[-\{\lambda_1\ (I_0 - c_1)^2 - \lambda_2\ (I_0 - c_2)^2 + 2\ \lambda_\psi\ (H(\phi) - H(\psi))\}] \tag{3.10}$$

This is also employed in [7], where a proof is provided to show that in the case of Chan and Vese without length term, using the update procedure based on the $c_1$ and $c_2$ equations (2.17, 2.18), and the original Chan and Vese equation for $\phi$ without length or area term ( 2.19 with $\mu$ and $\nu = 0$) is equivalent to using the same equations for $c_1$ and $c_2$, and the corresponding fast version of the equation for $\phi$ using the sign function (3.10 with $\lambda_\psi = 0$).

This could be particularly important for our method, as it will be extended to 3D where computational efficiency is critical.

However, for our purposes we omit the use of the sign function and stick to the update equation (3.9), since the sign function is very sensitive to the naturally noisy images we are dealing with, and we would like to keep as much information from the original image as possible. Also, the use of the sign function can cause some issues in determining when a stationary solution is reached. This is because numerically, even very small values, close to 0, can switch between positive and negative, misleading us to believe that $\phi$ is still rapidly changing. For this reason, it may be preferred to use the translation parameters $a$ and $b$ to determine when the shape prior has found its final position – ie. consider when these parameters become stationary as a stopping criterion.

An important point to consider is the fact that we are omitting the length term from Chan and Vese's method. The length term was meant to keep the 0-level contour of $\phi$ smooth by minimizing its length. Including this term would undermine the goal of increasing computational efficiency, but at the same time we expect that the restriction imposed by the shape energy term, $E_\psi$, will provide enough regularization for the 0-level contour of $\phi$. This is because the boundaries of our shape prior are already smooth, and the shape comparison term forces $\phi$ to look like $\psi$. Thus, the procedure employed to minimize this energy is as follows (with appropriate discretization):

- Initialize $\psi$ and $\phi$ to $\psi_0$ and $\phi_0$ – these could be identical
- Calculate the values for $c_1$ and $c_2$ using (2.17) and (2.18)
- Update the values for $a$ and $b$ using (3.8)
- Compute $\psi$ by translating the initial prior $\psi_0$ using the values for $a$ and $b$, and redistance by solving the signed distance equation
- Find the current $\phi$ using equation (3.9)
- Repeat the last four steps until a stationary solution is achieved, or for a preset number of iterations

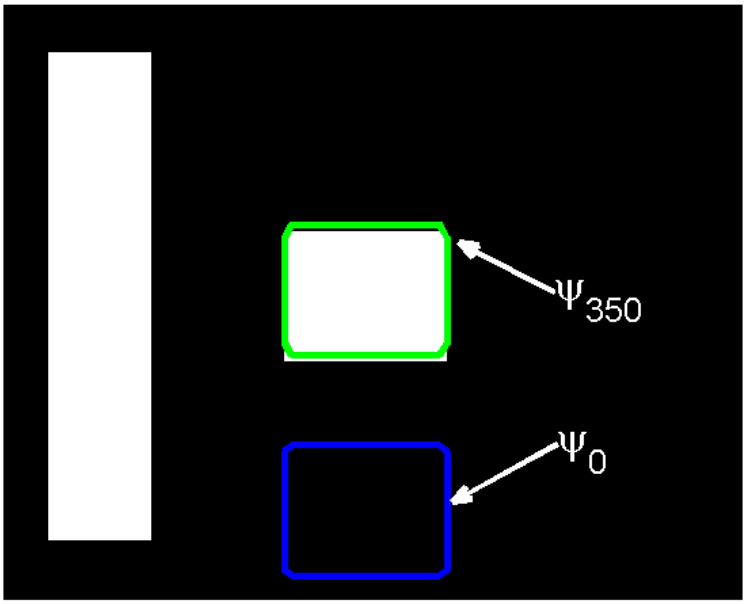


Figure 3.5: Adding a shape prior to Chan and Vese: find only the object we are interested in

To demonstrate how this method works, we first look at a simple case shown in Figure 3.5. Here we have a simple binary image containing two objects: two rectangles of different dimensions. The shape prior (and $\phi$) are initialized to have the same shape as the smaller rectangle, but different position. The desired outcome is to suppress the larger rectangle, in spite of its intensity being the same as that of the smaller rectangle. The parameter values for this experiment are $\lambda_1 = 0.01$, $\lambda_2 = 1$, and $\lambda_\psi = 5$. We take $\Delta t = 0.3$ and run the algorithm for 350 iterations. The reason for choosing $\lambda_1$ so small is the same as in the previous subsection and is due to the simplicity of the binary image causing the first term in our energy to be 0 in most of the image domain.

Now consider a more complicated image: a 2D brain slice. The issue with using the shape prior only in the previous subsection was that we can easily get stuck at the first local minimum we encounter. Here, we must consider the many local minima present due to the nature of the image itself. These images have plenty of possible positions for $\psi$ that are local minima, resulting once again in sensitivity to initial conditions. A couple of things we

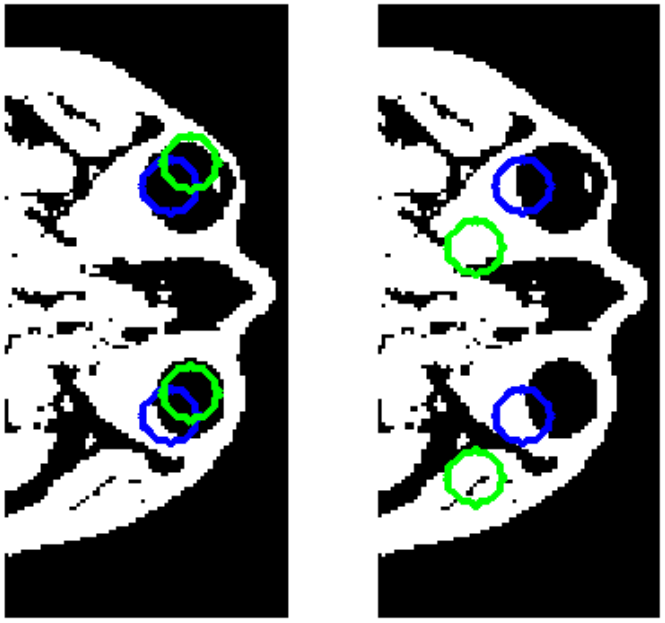

Figure 3.6: Chan and Vese with shape prior: a 2D brain image with a "good" and a "bad" result

can do in an attempt to combat this issue is to first roughly crop the image to the relevant region (where we expect to see the eyes), and then turn the grayscale image into a black and white image. We do this by thresholding: we select a "nice" intensity value using Otsu's method [11], and map the grayscale image to a binary one, by turning pixels of intensity $\leq$ this thershhold value to 0 (black), and pixels with intensity $>$ that value, to 1 (white). Otsu's method is described in more detail in the next chapter, as this technique is also used in 3D. Even with this simplified version of the image, choosing the parameters to obtain a satisfactory result is difficult, and starting position is still critical.

This problem is illustrated in Figure 3.6. The initial position of $\psi$ is shown in blue, and its final position is in green. Parameter values for both images are: $\lambda_1 = \lambda_2 = 1$, $\lambda_\psi = 1.5$, $\Delta t = 0.3$, and 50 iterations. In the first image, we can see that starting close enough to the eyes results in the desired final position. The reason is that if we consider the intensity inside $\psi_0$ (which is the same as $\phi_0$), we note that the pixels are mostly black. So, moving towards a region that is black will be beneficial – ie. moving all the way inside the eyes. In

the second image, however, the intensity inside our starting position is mostly white. This results in ending up at another local minimum which is completely in the wrong region. It is important to realize that this is the wrong region for us, but is perfectly reasonable for $\phi$ and $\psi$. To combat this issue we wish to somehow incorporate the information that our final object is expected to be of black (0) intensity. This would attract our shape prior towards black regions as opposed to white regions. This idea will be developed later on in this chapter.

### 3.2.3 Selective Shape Prior using Static and Dynamic Region Labelling

Some of the issues that still remain, even with the shape prior information, come from the fact that all of the above shape terms are defined on the whole image domain, which may contain many objects. Assuming we are only interested in one of these objects, of a particular shape, we can make an adjustment to the segmentation with shape prior model (3.7), in order to further narrow down the region in the image where our object of interest is located. This can be done by means of a labelling function, $L$. This function defines a region of the image domain where the shape prior is enforced. The idea of a labelling function is introduced in [9] and developed further in [7].

The simplest way to incorporate this idea into the Chan Vese with shape prior model is to define $L$ to be $+1$ inside a region containing our object, and $-1$ outside. Using this definition, we start with the Chan and Vese energy and add a modified shape energy term to it. Cremers et. al. in [9] have the following resulting energy to be minimized:

$$E = E_{CV} + \lambda_{\psi} \int_{\Omega} (\phi - \psi)^2 (L+1)^2 \, dxdy \tag{3.11}$$

Note that their shape term does not impose the Heaviside function to compare the segmentation $\phi$ with the shape prior $\psi$. This, however, imposes the necessity for both of these to be signed distance functions. To avoid this, [7] incorporates the labelling function as the following:

$$E = E_{CV} + \lambda_{\psi} \int_{\Omega} (H(\phi)\, H(L) - H(\psi))^2 \, dxdy \tag{3.12}$$

Note that their definition of the shape term differs slightly from [9], but the important point still holds: this forces the shape prior to only be applied to the region where $L > 0$ and $\phi > 0$. We expect the region where $L > 0$ to include the region where $\psi > 0$, otherwise our labelling is incorrect, as it does not include the object we are looking for. For this idea of the labelling region, $\psi$ is assumed stationary (ie. placed in the correct position to start with.) This shape term allows for the model to find the object inside this region that resembles our shape prior, but still find the correct segmentation of other objects outside of this region, without being restricted with information from the shape prior. However, in this definition of the labelling function we assume that the location of this region of interest is known in advance and constant throughout the evolution of $\phi$ and $\psi$. This is called static labelling. This may not be the case in all applications, but in many medical image processing problems it is reasonable. Similarly to a physician looking at medical images, we can use our general knowledge of anatomy to be able to know the relative location of some anatomical feature – such as the eyeballs. Furthermore, in our case we are working with registered images, so that we have a reasonable assumption for the cropping and orientation of the brain. Because of this, we are able to make a good guess as to how to define our labelling function $L$.

To this end, we may in fact incorporate this idea into our model, but in the form of cropping. Instead of running our segmentation algorithm on the whole domain, we will consider a cropped version of the image, such that we are certain that the eyeballs are completely contained inside the resulting image. The reason we will crop instead of explicitly defining a labelling function and running the segmentation procedure on the whole domain is that we do not require a segmentation of all objects in the brain – we only want one. The more unrelated information we can ignore, the faster and more efficient our final method will be. As well, this means that we can use our translating prior model as is, on the already cropped image. We will do this in 2D and 3D.

For some applications, even in a medical image setting, a static labelling is hardly

useful. An extension to this idea, known as dynamic labelling, allows for the labelling region itself to evolve in an unsupervised manner. This involves adding more terms to the energy which will help guide the function $L$ naturally. In Cremers et. al. [9], two more terms are added, resulting in the following:

$$E = E_{CV} + \lambda_\psi \int_\Omega (\phi - \psi)^2 (L+1)^2 \, dxdy + \int_\Omega \lambda^2 (L-1)^2 \, dxdy + \gamma \int_\Omega |\nabla H(L)| \, dxdy \tag{3.13}$$

To understand what this model does, first consider the last two terms. The first of those is meant to maximize the area of $\{L > 0\}$. This is because we want to ensure that the labelling region does not become so small that it no longer contains our target object. The last is a regularizing term, minimizing the length of the separation curve for the region where the labelling function is active and the region where it is not. Qualitatively, if we hold $\phi$ constant and consider the area term, we can see the meaning of the parameter $\lambda$:

$$\begin{aligned} L &\to +1 \quad \text{if} \quad |\phi - \psi| < \lambda \\ L &\leftarrow -1 \quad \text{if} \quad |\phi - \psi| > \lambda \end{aligned}$$

Alternatively, in [7], employing the Heaviside function again, the energy has a similar interpretation but slightly different form:

$$\begin{aligned} E = E_{CV} &+ \lambda_\psi \int_\Omega (H(\phi)H(L) - H(\psi))^2 \, dxdy \\ &+ \mu_1 \int_\Omega (1 - H(L)) \, dxdy + \mu_2 \int_\Omega |\nabla H(L)| \, dxdy \end{aligned} \tag{3.14}$$

However, the parameter choice for $\mu_1$ and $\mu_2$ is difficult using only the above. To help alleviate that issue, in [7] another term involving $\psi$ is added to the energy, in hopes that the interaction between $\psi$ and $L$ in the first part of the shape term above will provide more information for the desired evolution of $L$. This term comes from the expectation that at the desired segmentation, the shape prior $\psi$ should approximately segment the image in a reasonable manner. Then the term added to 3.14 becomes $\nu \int_\Omega (I_0 - c_1)^2 H(\psi) + (I_0 - c_2)^2 (1 - H(\psi)) \, dxdy$, where all parameters above are greater than or equal to 0, and $c_1$ and

$c_2$ are the same Chan and Vese constants as before.

This dynamic labelling component is able to separate the image domain actively into regions containing objects similar to the shape prior, from those containing background and possibly other objects. This can be useful in a situation where simply cropping the image is not reliable – ie. if we are looking for an object in a large database of images, but we do not have a reliable registration, or the anatomical nature of the object is such that a good guess for the cropping is not possible. Still, even in our case, the variety of brain images and richness of features present force us to use a cropping region that does not guarantee that the eyeballs are the only object left. In fact, there is some area of approximately constant intensity just behind the eyes that may confuse our shape prior if we do not start mostly inside the eyes. This problem is shown in Figure 3.6, and cannot be solved reliably using cropping as the only approach. The final piece to consider adding to our segmentation procedure is introduced next – the appearance prior.

### 3.2.4 Appearance Prior

Even with the help of shape priors and a labelling function, the nature of medical images still makes the segmentation problem difficult. The presence of many local minima can be difficult to overcome without requiring supervision. In the particular application of finding the eyeballs in a brain MRI image, we look for two circles (in 2D) of a certain radius and distance apart, such that the intensity values of the image inside and outside the shape prior boundary have minimum variance. However, there is some tissue behind the eyeballs of higher intensity that is a local minimum for our shape prior model. Since this tissue is directly behind the eyes, our model encounters an ambiguity, since as far as the energy is concerned, both "objects" are simply two local minima to settle for.

The main difference between the two minima: eyes vs. tissue behind eyes, is the difference in intensity. The eye balls in an MRI image appear with black intensity, while the tissue has a light gray intensity. This important distinction is just the right thing to add to our model, in the form of an appearance prior. An appearance prior incorporates knowledge beyond the target shape – one could specify expected intensity, orientation, scaling,

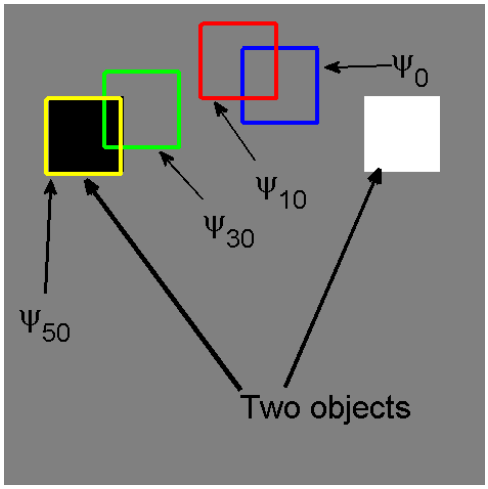


Figure 3.7: Using an appearance prior to match both shape and intensity for an object

etc. of the object in the image. For example, if we have a picture containing a white square and a black square on a gray background, and we have a shape prior matching both squares, we could specify exactly which kind of square we would like to look for by incorporating the information about our desired intensity. This simple scenario is illustrated in Figure 3.7.

This idea is developed in [3] by incorporating information about the expected intensity of our object. However, it is possible to specify more information to alleviate ambiguity arising in the particular application of image segmentation. For our case, intensity is all we need. Formally, we build on the energy models above by having Chan and Vese's segmentation energy, adding some shape prior energy term chosen from earlier in this section, and add a final appearance term. First, suppose we have an image defined on the prior shape domain, $I_p$. Define the embedded prior image as follows:

$$I_\psi = \begin{cases} I_p(x,y) \text{ if } \psi \geq 0 \text{ (inside object)} \\ c \in [0,1] \text{ if } \psi < 0 \text{ (outside object).} \end{cases} \tag{3.15}$$

Here $I_p$ can be obtained from similar images that have been manually segmented, in the same way that the shape prior $\psi$ is obtained. The constant value $c$ is an intensity value, used to define $I_\psi$ over the whole image domain. If we have no expectation as to what the average intensity outside of our object should be, we can set this value to Chan and Vese's

$c_2$ defined by the boundary of our shape prior. This way we are really only comparing the inside of the object in our image with that of the appearance prior. In fact, since there is no need to compare the image outside the object with this value, we will use a Heaviside function to ensure we are only looking inside the object. In the case of our application, we already have our shape prior, and we know that we are looking for an object which matches the shape prior but also has intensity close to 0. Therefore, in the above formulation for $I_\psi$, we can use a constant value of 0 for $I_p$. Note that as the shape prior $\psi$ evolves by translating, so does the appearance prior.

It should be noted that in [3], the underlying segmentation method used is based on an edge detector using a function based on the image gradient. This method is described in some detail at the beginning of Chapter 2. Regardless, any segmentation idea can be combined with the shape prior and appearance prior ideas by forming a linear combination of the appropriate energy terms. The term we add as an appearance similarity measure is:

$$\begin{aligned} E_{Ap} &= \int_\Omega (I_0 - I_\psi)^2 \, H(\psi) \, dxdy \\ &= \int_\Omega (I_0 - I_{\psi_0}(x-a, y-b))^2 \, H(\psi_0(x-a, y-b)) \, dxdy \end{aligned} \tag{3.16}$$

Keeping in mind that this term is to be added to an already existing shape prior model, we consider minimizing it with respect to the translation parameters $a$ and $b$, to be able to update our gradient descent equations for these. Let us minimize the above with respect to $a$ by first writing $I_\psi$ as $I_\psi = c_2(1 - H(\psi_0(x-a, y-b)))$:

$$\begin{aligned} E_{Ap} &= \int_\Omega (I_0 - c_2(1 - H(\psi_0(x-a, y-b))))^2 \, H(\psi_0(x-a, y-b)) \, dxdy \\ \frac{\partial E_{Ap}}{\partial a} &= -2 \int_\Omega (I_0 - c_2(1 - H(\psi_0(x-a, y-b)))) \\ &\quad \cdot \delta(\psi_0(x-a, y-b)) \, \psi_{0x}(x-a, y-b) \, H(\psi_0(x-a, y-b) \, dxdy \\ &\quad - \int_\Omega (I_o - c_2(1 - H(\psi_0(x-a, y-b))))^2 \, \delta(\psi_0(x-a, y-b)) \, \psi_{0x}(x-a, y-b) \, dxdy \\ &= \int_\Omega [-2(I_0 - I_\psi) \, H(\psi) - (I_0 - I_\psi)^2] \, \delta(\psi) \, \psi_x \, dxdy \end{aligned} \tag{3.17}$$

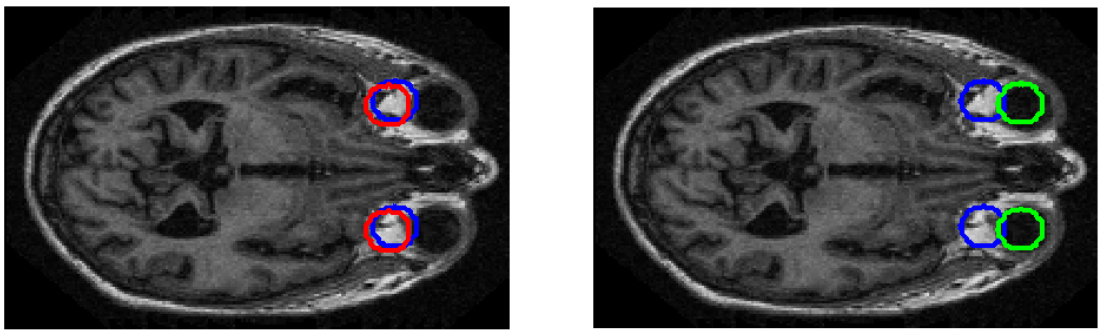

Figure 3.8: Comparing the translating prior model with and without an appearance term in a 2D black and white brain image.

This equation and its equivalent for $b$ can simply be added to the shape prior model. Note that if we choose to use a model that combines all three – segmentation, shape prior and appearance prior – the last of these does not involve $\phi$, so no modification is necessary to its update equation. This simple yet powerful addition to the model gives much more robust results compared to a shape prior only model. To demonstrate how this term affects the segmentation process, consider the simple situation with the two squares, in Figure 3.7. Note that the shape prior is repelled away from the white square thanks to using an appearance prior of constant intensity 0 (black).

Now to test this on a brain image. Consider the problem illustrated in Figure 3.6: depending on the initial position, the shape prior either moves towards the black region (eyes), or the light gray tissue behind the eyes, as both of these positions satisfy the shape matching. By adding the 0 (black) appearance prior the shape prior should be repelled away from the light gray region and towards the black – the eyes. The model implemented here is the translating shape prior – the simplest shape prior model discussed in the beginning of this chapter, with parameters $\lambda_1$, $\lambda_2$, and $\lambda_\psi$, the last one corresponding to the weight of the appearance prior. Figure 3.8 shows what happens with the same initial condition, with $\lambda_1 = \lambda_2 = 1$, but $\lambda_\psi = 0$ in the first picture and $\lambda_\psi = 2$ in the second.

To better compare the shape model with and without appearance prior, let us do an experiment. We start with a grayscale 2D brain slice image and place the shape prior where

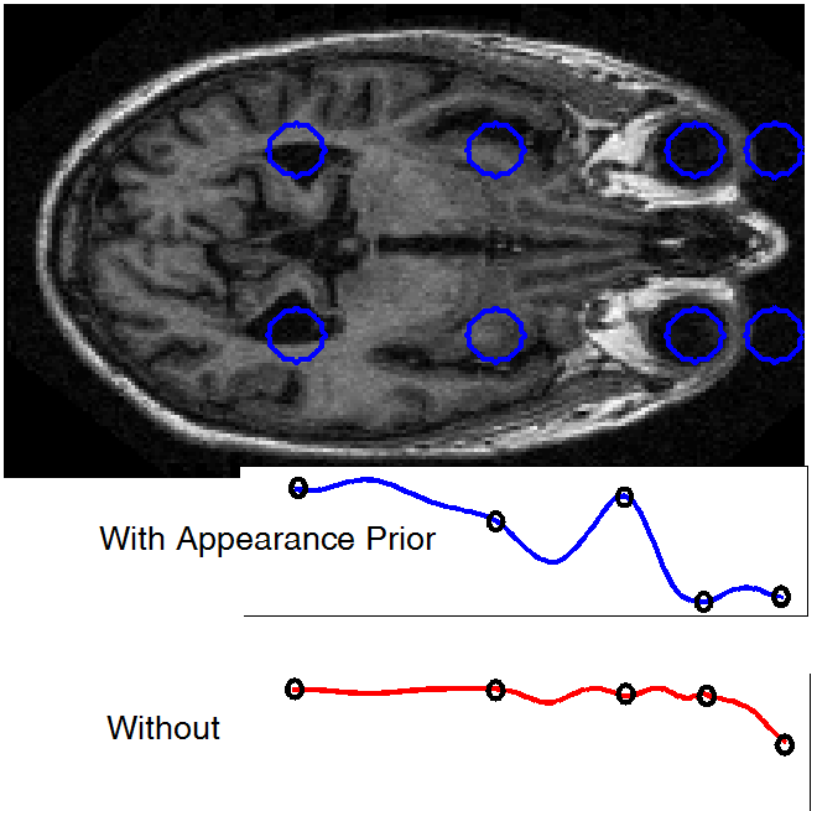


Figure 3.9: Comparing the translating prior model with and without an appearance term: Energy

the eyes are. Now we move the shape prior along the image domain and evaluate the energies for both models at the different positions of the shape prior. In order to be able to compare these in a simple way, we only move the prior in the $x$ direction. Figure 3.9 shows a few of the positions considered for the prior, with the first plot (in blue) showing the energy of the model including the appearance prior, and in the second plot (red) the energy of the model without. We can see that this makes a great difference when it comes to the gray tissue behind the eyes: adding the appearance prior has indeed successfully eliminated that region as a local minimum.

Now that we have found a way to drive the shape prior towards black regions over gray, we can use a black and white version of the image with more confidence that we will end

up in the black region. Using black and white for the shape and appearance prior stage of our procedure will speed up the process – the gray version of the image has too much detail which causes the shape prior to move slowly compared to only two levels of intensity. However, for the final stage of the segmentation, the Chan and Vese model, we would like to have as much accuracy as possible, and for this reason we will keep the original gray version of the image to retain as much detail as possible.

## 3.3 Final Shape and Appearance Model

Now that we have all the pieces, we consider building the model best suited to our application. Recall that the full segmentation procedure is done in two stages: first, we find a rough location for the eyeballs using a shape model, and second, we send this information to Chan and Vese's active contours model to get the final eyeball segmentation. There are many options for building the shape portion of the model, as discussed in this chapter. We have to choose between a simple translating prior model, a model which couples Chan and Vese's method with the shape prior, a model with a dynamic labelling function, and possibly adding an appearance prior.

The method taken here, for the shape stage of the segmentation process, is a linear combination of the translating prior and the appearance prior. This is a straight forward model for finding an object of a particular shape in an image, and is sufficient for the purpose of this thesis. The benefits of choosing this over a method which involves both $\phi$ and $\psi$ at the same time are few. One is that our parameter selection becomes easier. In a method involving both $\phi$ and $\psi$ it is often difficult to find a suitable parameter to define the trade-off between the two. In many situations the intensity-driven segmentation, $\phi$, is pulling the overall process in one direction, while the shape prior $\psi$ disagrees with that direction, and depending on the trade-off parameter, both move in a similar direction but very slowly. This is not robust and many experiments are necessary to select the "correct" parameter values in such a model.

Another reason for choosing the simple translating prior model is that we do not expect a highly accurate segmentation to result from this stage of the overall process. We only

need an approximate location for the eyes to be used as an initialization for $\phi$ in the active contours without edges method. Just as any alternative approach, this translating prior model is prone to getting stuck at local minima. This issue is addressed by adding the appearance prior discussed in the previous section. Combining these two ideas together, we turn our problem into trying to find two circular regions some distance apart that have a low intensity variance inside and outside the object boundary, and the intensity inside should be close to 0 (black). More formally, the energy we are minimizing becomes:

$$\begin{aligned} E_{shape} = \lambda_1 \int_\Omega (I_0 - c_1)^2 \, H(\psi) \, dxdy + \lambda_2 \int_\Omega (I_0 - c_2)^2 \, (1 - H(\psi)) \, dxdy \\ + \lambda_\psi \int_\Omega (I_0 - I_\psi)^2 \, H(\psi) \, dxdy \end{aligned} \tag{3.18}$$

Here $I_\psi$ is defined as in (3.15) – 0 inside the object described by $\psi$. This appearance prior moves with $\psi$ by translation only, with translation parameters $a$ and $b$. Rewriting the shape prior $\psi(x,y) = \psi_0(x-a, y-b)$ as before, we can minimize the above with respect to $a$ and $b$ in order to obtain the associated gradient descent equations that we use to evolve $\psi$. The final resulting equations are:

$$\begin{aligned} \frac{\partial a}{\partial t} &= \int_\Omega \lambda_1 \, (I_0 - c_1)^2 - \lambda_2 (I_0 - c_2).^2 + \lambda_\psi (I_0 - I_\psi)^2) \, \delta(\psi) \psi_x \, dxdy \\ &\quad + 2 \int_\Omega \lambda_\psi (I_0 - I_\psi) \, H(\psi) \, \delta(\psi) \, \psi_x \, dxdy \\ \frac{\partial b}{\partial t} &= \int_\Omega \lambda_1 \, (I_0 - c_1)^2 + \lambda_2 (I_0 - c_2).^2 + \lambda_\psi (I_0 - I_\psi)^2) \, \delta(\psi) \psi_y \, dxdy \\ &\quad + 2 \int_\Omega \lambda_\psi (I_0 - I_\psi) \, H(\psi) \, \delta(\psi) \, \psi_y \, dxdy \end{aligned} \tag{3.19}$$

In order to ensure that when solving these equations we are moving as fast as we can in a way that gives opportunity for both directions to evolve, we employ a quasi-adaptive time stepping. The size of the images we are dealing with restricts us as to how fast we can move, and to ensure that we do not push our prior out of the domain by taking large steps, the following scheme is used to update $a$ and $b$:

$$a^{new} = a^{old} + \Delta t \cdot \frac{A}{\max(A, B, 1)}$$
$$b^{new} = b^{old} + \Delta t \cdot \frac{B}{\max(A, B, 1)} \tag{3.20}$$

Here $A$ is the magnitutde of the right hand side of the descent equation for a, and B – the equivalent for b. We ensure our prior is moving as fast possible without flying out of the domain by scaling our update step by the largest "speed," and we need to compare these with the number 1 in case our $A$ and $B$ are $\ll 1$, to avoid speeding ourselves up artifically. In this way we can more comfortably choose a small but fixed $\Delta t$ to maintain stability in most scenarios ($\Delta t$ is usually set to 0.005). The rest of the terms are discretized as before, making use of the regularized versions of the Heaviside and delta functions.

The overall shape and appearance model then becomes the following:

- Define an initial shape prior and appearance prior using one of the methods listed in the beginning of the chapter. For our application, we take four 2D images, find the approximate radius of an eye in each image, the approximate distance the eyes are apart, and the approximate location of the center of one eye. We then average all of this data and create a shape prior based on this information. The appearance prior is simply set to intensity 0 (black) in the region where the shape prior function $\psi_0 > 0$.

- The image domain along with the shape and appearance functions is roughly cropped to a smaller size containing the eyes. The image is also transformed to a black and white (binary) image using Otsu's thresholding method.

- The update procedure is initialized and $\psi$ is translated over the domain by the update equations (3.20) until a stationary solution is reached (ie. until the translation parameters $a$ and $b$ become stationary). This concludes the shape stage of our method.

Passing this resulting shape prior to Chan and Vese's segmentation algorithm should greatly aid in the final segmentation of the eyes. At this final stage, we take circles of a larger radius than the shape prior's, centered at the current shape prior location, and set the intensity of the image outside to $c_2$. This way, the only region we expect to have intensity 0

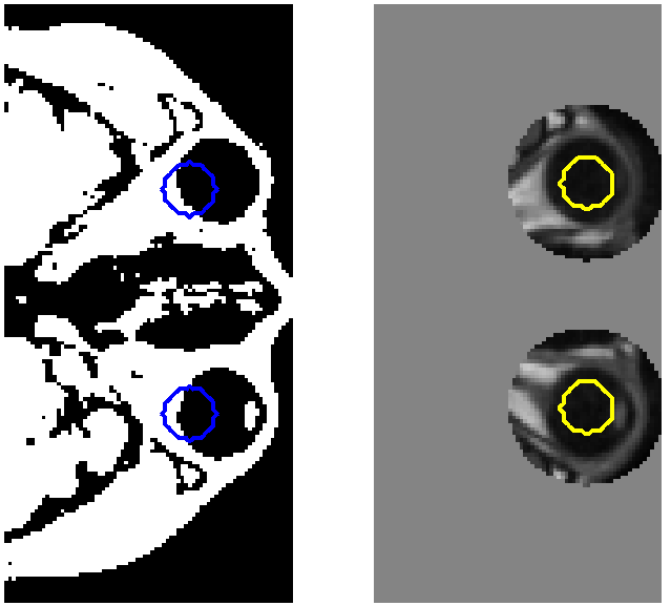

Figure 3.10: 2D brain image: starting position of shape and appearance prior on black and white image, and resulting shape prior used as initialization for Chan and Vese's method, on a grayscale image

is the eyes, and the gray tissue right behind the eyes should now blend in with the "outside" region. At this stage of the procedure, $c_2$ is our best approximation to the average intensity in the image outside of the eyes. This $c_2$ is the same Chan and Vese value, but calculated on the grayscale image based on the current position of the shape prior. As well, the nose cavity (which is also approximately black in an MRI image, located between the eyes) is eliminated by the use of these bigger circles. Hence, the result from this stage should be the final product we are looking for.

Figure 3.10 shows this link between the shape stage and the Chan and Vese stage of our model. The first image is the black and white version of the original image, and the blue corresponds to the initial shape prior position. The second image shows the transformation of the grayscale image by the use of the circles of radius twice as large as that of our shape prior, with the outside region set to intensity $c_2$. In yellow, we have the final shape prior position after the shape and appearance procedure, and this is the image to be passed to Chan and Vese's algorithm with initial segmentation $\phi_0$ defined by this yellow prior.

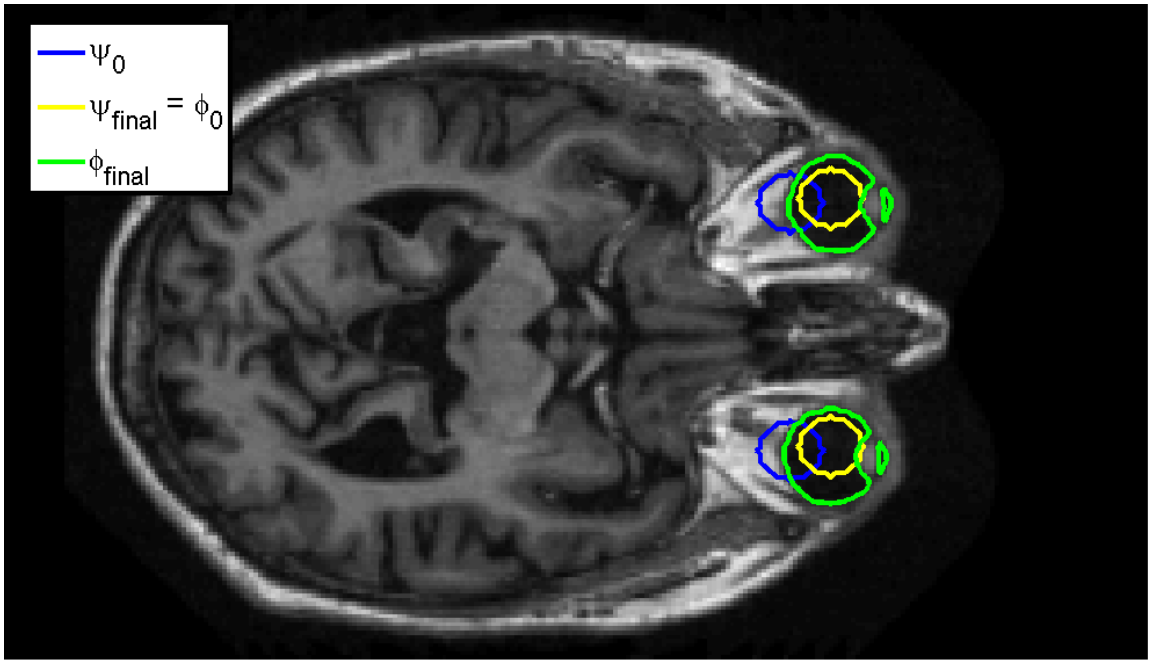


Figure 3.11: Demonstrating the full shape, appearance, and active contour model for a 2D brain image

Putting this together, Figure 3.11 shows, in blue, the starting position of a shape prior ($\psi_0$), the final result of the shape and appearance model that we use (yellow), and having passed this information to initialize the segmentation ($\phi_0$), the final Chan and Vese method result is shown in green. We now need to translate the above procedure to generalize to 3D and apply it on a set of 3D images.

# Chapter 4

# Extension to 3D

The application which motivated this research needs the final segmentation method to be implemented in 3D – so, we start with the 2D model described in the previous two chapters and extend the ideas to 3D. Results will be presented at the end of this chapter for the final information extracted from some brain MRI images, describing the eyeballs. These results are meant to demonstrate a proof of concept, and the conclusion to be drawn from this is that the equations used to solve this particular segmentation problem indeed achieve their goal. The numerical approach for solving these equations can be improved and further analyzed in the future; however, for this thesis we leave all discretization explicit without much analysis, to be able to test the concept and conclude that it is worth developing further and refined in terms of numerical set-up.

## 4.1 Chan and Vese's Model Revisited

All of the ideas in the full shape prior, appearance prior, and active contours without edges segmentation model naturally generalize to any dimension. The nature of the energy functionals involved in building the model makes this straightforward. The interpretation of individual pieces of the model also hold, leaving our intuition about how the model works unchanged. First, recall the full Chan and Vese energy arising from trying to find the best piecewise constant approximation to the original image, while maintaining some desirable regularization conditions (re-written in 3D):

$$
\begin{aligned}
E_{CV}(c_1,c_2,\phi) = \mu \int_\Omega \delta(\phi(x,y,z))|\nabla\phi(x,y,z)|\, dxdydz \\
+ \nu \int_\Omega H(\phi(x,y,z))\, dxdydz \\
+ \lambda_1 \int_\Omega |I_0(x,y,z) - c_1|^2\, H(\phi(x,y,z))\, dxdydz \\
+ \lambda_2 \int_\Omega |I_0(x,y,z) - c_2|^2\, (1 - H(\phi(x,y,z)))\, dxdydz
\end{aligned} \tag{4.1}
$$

The only difference from 2D is that instead of a two dimensional image with pixels, where $I_0(x,y)$ denotes the intensity of the image at pixel coordinates $(x,y)$, we now have a 3D volumetric image with voxels, such that $I_0(x,y,z)$ denotes the intensity of the image at voxel location $(x,y,z)$. As well, when evolving the segmentation $\phi$, instead of a 0 level set curve, we now have a 0 level set surface (isosurface). This still defines the boundary of our object, and we still define $\phi > 0$ inside the object and $< 0$ outside. The first term in the energy aims to minimize the surface area of the segmentation $\phi$, while the second minimizes the volume. With this simple adjustment in mind, we now look at the corresponding Euler-Lagrange equations for $c_1$, $c_2$, and $\phi$. These essentially remain unchanged:

$$
\begin{aligned}
c_1 &= \frac{\int_\Omega I_0\, H(\phi)\, dxdydz}{\int_\Omega H(\phi)\, dxdydz} \\
c_2 &= \frac{\int_\Omega I_0\, (1 - H(\phi))\, dxdydz}{\int_\Omega (1 - H(\phi))\, dxdydz}
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\partial\phi}{\partial t} = \delta_\varepsilon(\phi)\, \left[\mu\, \mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \nu - \lambda_1\, (I_0 - c_1)^2 + \lambda_2\, (I_0 - c_2)^2\right]\, , t > 0 \\
&\phi(x,y,z,t=0) = \phi_0(x,y,z)\ , \text{in } \Omega \\
&\frac{\delta_\varepsilon(\phi)}{|\nabla\phi|}\, \frac{\partial\phi}{\partial\vec{\eta}} = 0\ \ , \text{on the boundary of } \Omega
\end{aligned} \tag{4.2}
$$

We employ the same regularized versions of the Heaviside and delta functions, and we still set the parameter controlling the volume of our object (area in 2D), $\nu$, to 0. Looking at the first two equations above, for $c_1$ and $c_2$, we see that we can once again evaluate these in the same way as in 2D. The boundary conditions remain as zero Neumann, and are implemented as in 2D. The discretization of the update equation for $\phi$ is also a natural extension

to 3D: we employ the same notation for forward difference $\Delta^x_+ \phi_{i,j,k} = \phi_{i+1,j,k} - \phi_{i,j,k}$, and backward difference $\Delta^x_- \phi_{i,j,k} = \phi_{i,j,k} - \phi_{i-1,j,k}$, with the index $k$ corresponding to the $z$-direction now. The numerical scheme for updating $\phi$ then becomes (with superscript $t$ denoting the current time level):

$$
\begin{aligned}
\frac{\phi^{t+1}_{i,j,k} - \phi^t_{i,j,k}}{\Delta t} = \delta_\varepsilon(\phi^t_{i,j,k}) \cdot \\
\Bigg[ \mu\, \Delta^x_- \cdot \left( \frac{\Delta^x_+ \phi^t_{i,j,k}}{\left( (\Delta^x_+ \phi^t_{i,j,k})^2 + \frac{1}{4}(\phi^t_{i,j+1,k} - \phi^t_{i,j-1,k})^2 + \frac{1}{4}(\phi^t_{i,j,k+1} - \phi^t_{i,j,k-1})^2 \right)^{\frac{1}{2}}} \right) \\
+ \mu\, \Delta^y_- \cdot \left( \frac{\Delta^y_+ \phi^t_{i,j,k}}{\left( \frac{1}{4}(\phi^t_{i+1,j,k} - \phi^t_{i-1,j,k})^2 + (\Delta^y_+ \phi^t_{i,j,k})^2 + \frac{1}{4}(\phi^t_{i,j,k+1} - \phi^t_{i,j,k-1})^2 \right)^{\frac{1}{2}}} \right) \\
+ \mu\, \Delta^z_- \cdot \left( \frac{\Delta^z_+ \phi^t_{i,j,k}}{\left( \frac{1}{4}(\phi^t_{i+1,j,k} - \phi^t_{i-1,j,k})^2 + \frac{1}{4}(\phi^t_{i,j,k+1} - \phi^t_{i,j,k-1})^2 + (\Delta^z_+ \phi^t_{i,j,k})^2 \right)^{\frac{1}{2}}} \right) \\
- \lambda_1 \left( I_{0,i,j,k} - c_1(\phi^t) \right)^2 + \lambda_2 \left( I_{0,i,j,k} - c_2(\phi^t) \right)^2 \Bigg]
\end{aligned}
\tag{4.3}
$$

Notice once again that Chan and Vese use an implicit scheme, while we use a fully explicit one, in order to be able to easily implement the full model in 3D and validate our concept. This particular discretization of the divergence term is similar to the one proposed in [19], and a convergence analysis for another similar numerical scheme is available in [1] (for the implicit scheme in 2D). With the scheme 4.3 in mind for the Chan and Vese portion of the full energy model, we now move on to consider how to extend the shape prior to 3D.

## 4.2 Shape and Appearance Prior Model in 3D

Recall that we rely on Chan and Vese's segmentation method to provide a final accurate segmentation of the eyeballs in a brain image. Incorporating shape and appearance information into the first stage of our full procedure is simply meant to provide a good starting

position for $\phi$. In 3D we still need a good definition for an initial shape prior. This can be obtained in the same way as in 2D: using a previously segmented image, or an average of existing segmentations of the desired object shape. Another possibility is again segmenting an image by hand. In 3D this can be done by segmenting individual 2D slices of a 3D volume, and compiling the slices back together.

In the case of searching for eyeballs in a 3D brain image, we are able to simply use two spheres. We once again look at a few images and average the radius for each sphere and the distance the two spheres should be placed apart. Our images are still registered, and hence aligned and cropped in a consistent manner, so we can ignore rotation and scaling for the evolution of the shape prior $\psi$. To aid in removing the many local minima present in these images, we again perform a rough cropping of the full image to a smaller portion containing the eyeballs – and hopefully not much else.

The problem with gray tissue directly behind the eyes providing an unwanted local minimum is addressed by the appearance prior. In the same way as before, we can specify that we expect our desired object to have the same shape as the shape prior $\psi$, but translated by $(a, b, c)$ units in the $x$, $y$, and $z$ directions, respectively, and with intensity of approximately 0 (black) inside. We expect this to be sufficient to push our shape prior towards the black region defined by the eyeballs, and to speed this process up we once again use a black and white version of the image.

The procedure we use to turn a grayscale image into a black and white (binary) one is based on Otsu's method [11]. As mentioned in the previous chapter, transforming a grayscale image to binary involves defining a threshold value which separates all the pixels in the image into two classes: those that map to 0 (black) and those that map to 1 (white). Consider a grayscale image $I_0$ with gray level intensities between 0 and 1. Selecting a value that is between 0 and 1 to define this class separation can be difficult. What Otsu's method does is try to define this threshold value in a way that the between-class variance is maximized.

The method operates directly on the histogram of an image, so it easily generalizes to

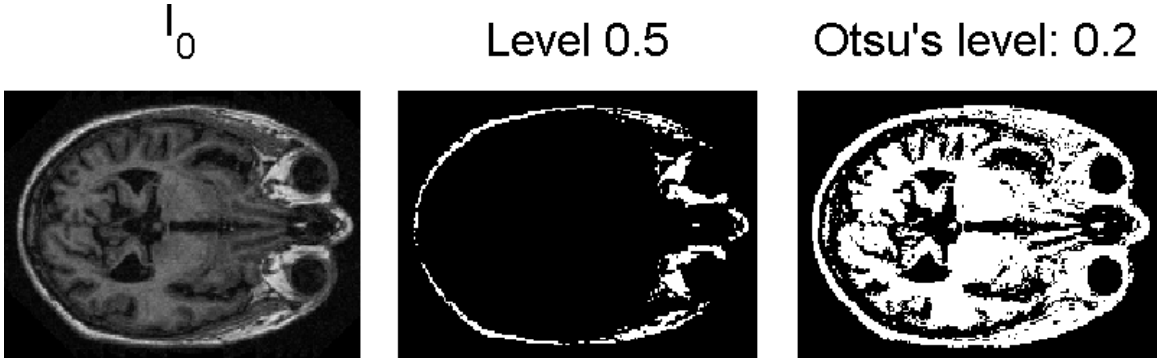


Figure 4.1: Turning a grayscale image to binary using Otsu's thresholding method

any dimension. The idea is to find a threshold value, $c$, and classify all pixels with intensity $\leq c$ to 0 and the rest to 1. Otsu's method uses statistical tools to find the value of $c$ which most clearly separates the pixels into these two classes – ie., we want the least amount of pixels that are on the border of belonging to one or the other class. We want the mean of the original intensities of the pixels which get mapped to 0 to differ as much as possible from the corresponding mean of the pixels which get mapped to 1. This should then provide a black and white version of the original image which would be most recognizable by the human eye.

To illustrate this point, consider Figure 4.1. We can see a 2D slice of a brain image: the first instance is the original grayscale image, the second is a blind thresholding at level 0.5 (medium gray) and the third is the binary image obtained using Otsu's method. Notice in particular that for us, this gives a very nice separation of the eyes from the gray tissue behind the eyes which was causing problems as a local minimum. An important point to mention is that this threshold value $c$ should be obtained based on the entire image, not the cropped region containing the eyes only – the full histogram provides the optimal value, specifically for our application.

As in the 2D model, we use this black and white image in the shape prior stage. Recall the energy:

$$E_{shape} = \lambda_1 \int_\Omega (I_0 - c_1)^2 \, H(\psi) \, dxdydz + \lambda_2 \int_\Omega (I_0 - c_2)^2 \, (1 - H(\psi)) \, dxdydz$$
$$+ \lambda_\psi \int_\Omega (I_0 - I_\psi)^2 \, H(\psi) \, dxdydz \tag{4.4}$$

We again use the regularized versions for the Heaviside and delta functions, and all terms can be computed as in 2D. We minimize this energy with respect to the translation parameters $(a,b,c)$ by re-writing the shape prior $\psi$ at any given iteration of our method as $\psi(x,y,z) = \psi_0(x-a, y-b, z-c)$ as before, and end up with the same gradient descent equations for each parameter (with one extra for the $z$-direction). We also use the same time stepping scheme as in the 2D case to ensure we translate the prior at a reasonable speed without letting it move out of our domain.

After obtaining the final position of $\psi$, we expect to have found roughly the location of the eyeballs. Now we pass this information to Chan and Vese's procedure in 3D to initialize $\phi_0 = \psi$. Similarly to 2D, we now return to using the original grayscale version of the image in order to obtain as accurate a final segmentation as Chan and Vese can give us, by using all the information we have in the image. To simplify this and further eliminate possible local minima, we define two spheres, each centered at one of the two spheres from $\psi$, but with radii twice as large. Everything in the image outside of these spheres is set to its average intensity, and we expect to be left with the eyeballs alongside pieces of the gray tissue behind them and some other elements contained in the brain image. Now we are ready to use the active contours without edges model to arrive at the final product. Some examples of our results in 3D are shown next.

## 4.3 Results

Visualizing results in 2D is straightforward, but there are a few options in 3D. In 2D, the final segmentation curve is easily shown using a contour plot overtop of a 2D image, where $\phi = 0$. For 3D, the final segmentation is represented by the isosurface $\phi = 0$. We can extract this data and create a surface plot showing the eyeballs in 3D, and this way we can ensure no irrelevant pieces of noise/objects have been grouped together with the eyeballs as our

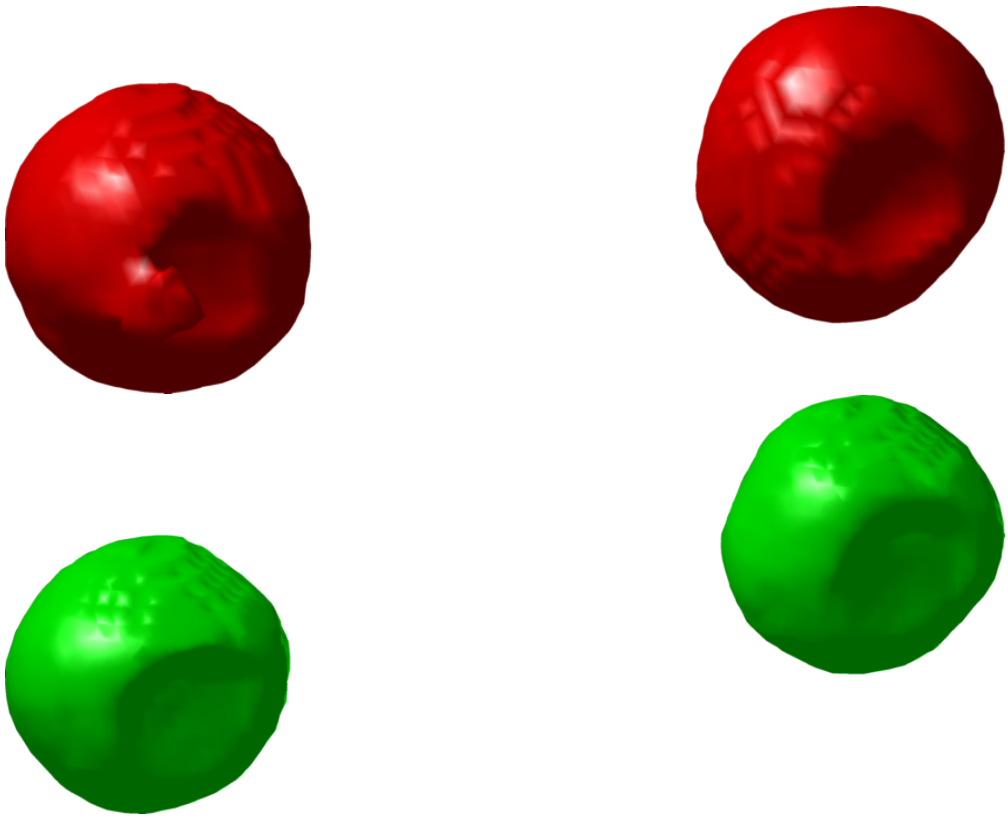

Figure 4.2: Two sets of extracted eyeballs. Note: the voxelization is due to how the pictures are made, not $\phi$ (see slices below)

final object. However, this does not provide convincing information that indeed the segmentation is "correct." To be able to compare carefully the segmentation with the original image, we can take 2D slices of our volumetric brain image and superimpose, just as in 2D, a 0 level contour plot of the corresponding slice of $\phi$. Finally, to see things in perspective, we can render a "see-through" version of the original image of the brain together with the eyeballs, as discovered by our procedure.

First, let us compare two surface plots of the resulting eyeballs. In Figure 4.2 we see the results in two different images. Note that the surface of the eyes appears rough – this is an image quality problem. The size of the cropped region our computations take place on is roughly 100x60x40 voxels, and the eyeballs appear as spheres of radius approximately

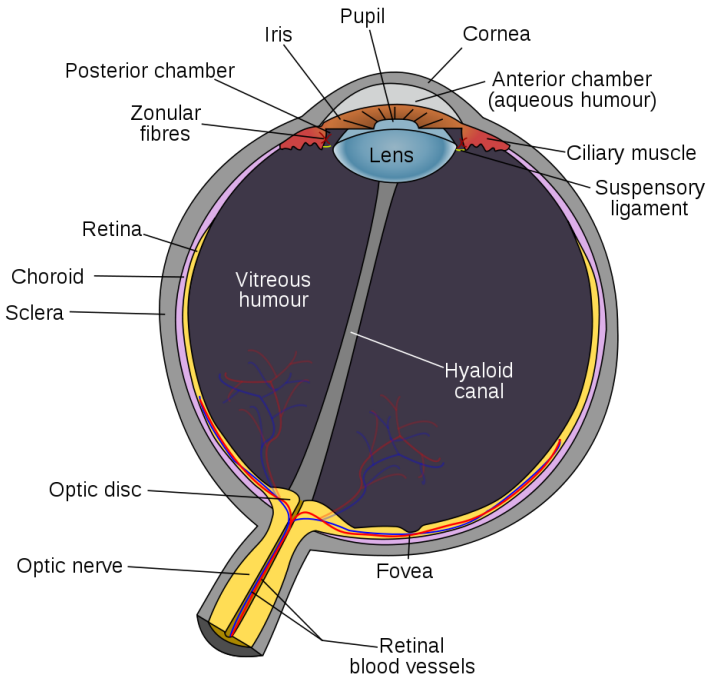


Figure 4.3: The anatomy of an eyeball

8 voxels. We can see that the two individuals have slightly different eyeballs, as expected, and there is some interesting features to be noticed. Firstly, there seems to be a piece of the eyeball "missing." This is actually a natural anatomical feature of the eye – the lens. The nature of MRI imaging causes the lens to appear with different intensity from black, so it is no surprise that our method does not find it. To compare these results to a diagram of an eyeball, we include Figure 4.3. In the left-hand eye of the first set of eyeballs we have so far, there is a piece of tissue sticking out and wrapping around the front of the eye. Looking again at the diagram, we can see that this is likely the anterior chamber. Depending on what specific purpose the full eyeball segmentation is used for, we can fine-tune the parameters in our model to either capture this feature or dismiss it. As well, note that in this thesis, although we refer to our desired object as the eyeballs, it is actually the vitreous humour that we find.

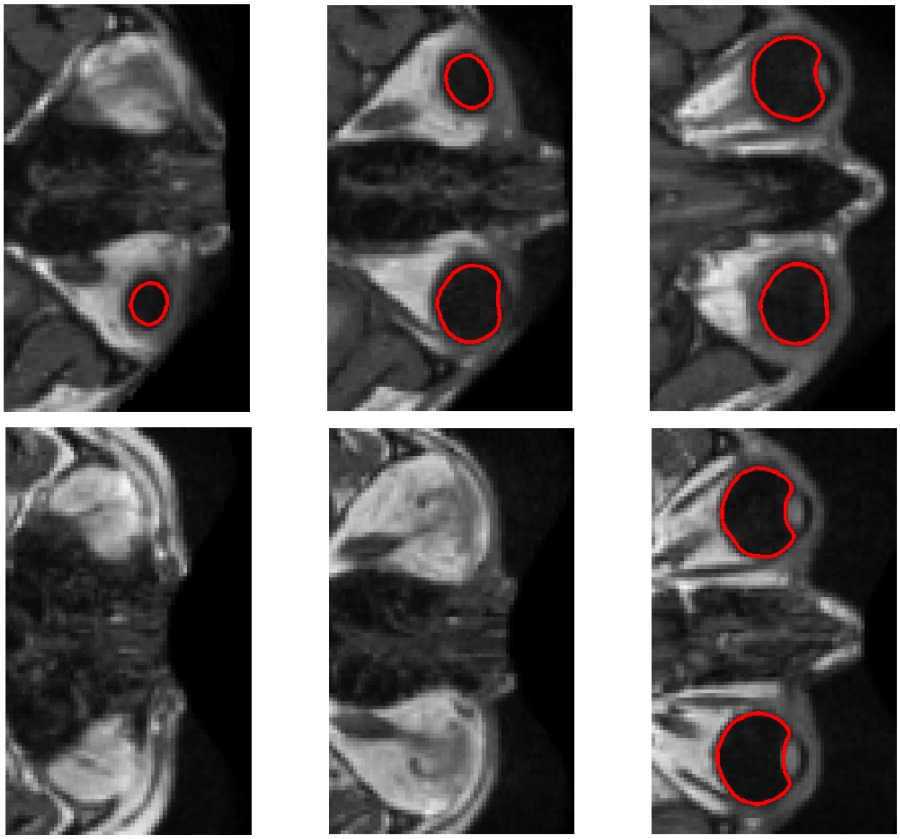

Figure 4.4: 2D slices of two brain images with their corresponding segmentations of the eyes, in the neck-up direction

To be able to view the two segmentations alongside their original images, we consider taking a few slices through the full image and placing the 0 level contour of the corresponding slice of $\phi$. Figure 4.4 shows, on the first row, three slices of the image corresponding to the first set of eyeballs above, and the second row corresponds to the second set of eyeballs. The direction along which the slices are extracted is going from the neck towards the top of the head: ie. horizontally slicing the brain from ear to ear, essentially. The specific slices correspond to the same values for both images, and we can see that the location of the eyeballs is slightly "higher" in the second image. We also see in the first image that the eyes are not perfectly level – this could either be indeed the case for this individual, or an imperfection in the registration of this particular image.

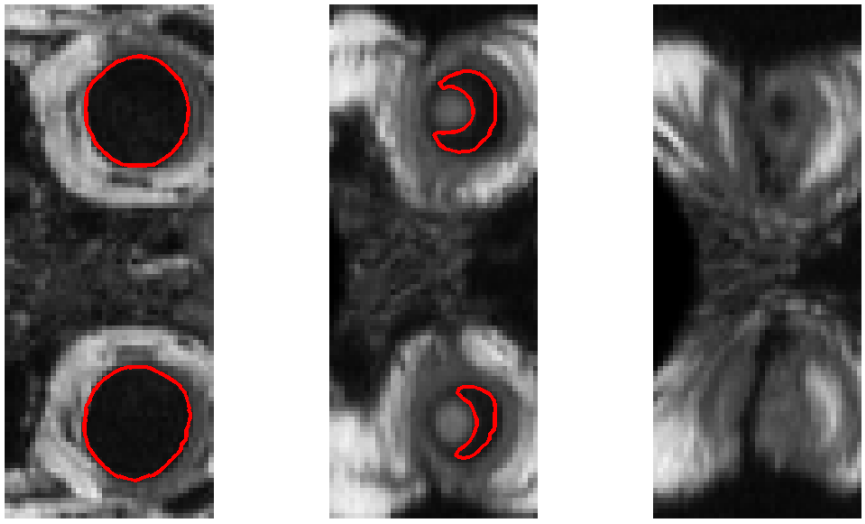

Figure 4.5: 2D slices of a brain image with its corresponding segmentation of the eyes, in the direction going from the back of the head to the front

Next, let us examine a few slices in the other two directions. These will only be shown for the second image. Figure 4.5 shows three slices coming from the back of the head towards the front. The first slice is somewhere in the middle of the eyeball, and we can see from the contour that the boundary appears to be found very well, as in the 2D case. The second slice is at the very front of the eyeballs and they appear as half moon shapes. This is the same phenomenon as in the surface plots of Figure 4.2. The last slice is just past the front of the eyeballs, and we can see in the middle region, both in the second and this third slices, the nasal cavity – it appears black. If our procedure did not include the stage of using two larger spheres to define our active domain for Chan and Vese's segmentation, this region would have caused some issues due to its intensity, and would likely have required some post-processing or extensive search for the "correct" parameters.

Finally, Figure 4.6 shows some slices taken along the direction connecting both ears. The first column shows two slices of one of the eyes, while the second column shows two slices of the second eye. It is possible to create these kinds of 2D plots at every slice in the image, for all three directions (neck to top of head, back of head to front, one ear to the other), and to compile these and view them as a movie. This visualization provides a good

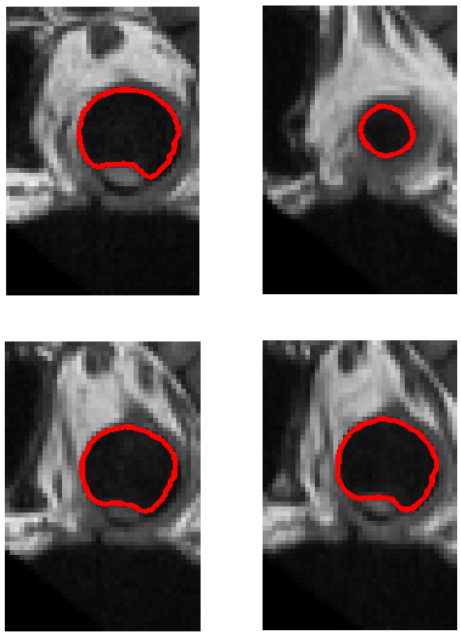

Figure 4.6: 2D slices of a brain image with its corresponding segmentation of the eyes, in the direction going from one side of the head/ear to the other

option for visually validating the results of the segmentation.

The above results let us take a close look at how our method performs on 3D images, and the results are encouraging. To put all of this in perspective, we consider the full 3D volume of the brain. To be able to see what is going on inside, we give every voxel in the image some transparency, and display here a 2D projection of the 3D data viewed from some angle. Since the brain has much detail, it is difficult to define a particularly clear color scheme to display the final result, so we include two versions of the rendering: one using a blue-to-red colour map, and a black and white map. The blue and the black both correspond to values of low intensity, while the red and white correspond to high intensity. These are shown in Figures 4.7 and 4.8. The first image in both those figures is the transparent rendering of the original brain MRI image. The second image corresponds to the original MRI image with the eyeballs found by our method inserted, set to the highest intensity only for the purposes of visualization. We can see that in the original image alone, since the MRI process does not capture any information for the eyeballs, it is difficult to

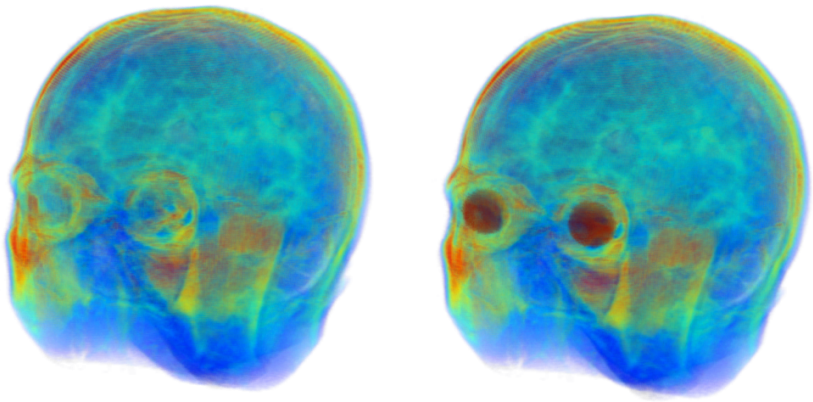

Figure 4.7: A transparent 3D volume image: finding the eyeballs

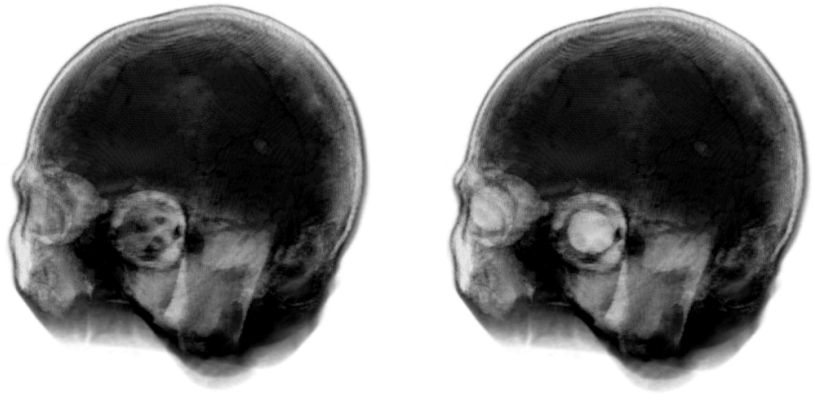

Figure 4.8: A transparent 3D volume image: finding the eyeballs, a black and white version

look at this image and decide exactly where they belong and how much space they take up. In contrast, the results we have obtained fit well with our intuition about what we expect to find.

# Chapter 5

# Conclusion and Future Research

## 5.1 Method Summary and Generalization to Other Applications

Image segmentation is a difficult problem to solve. Building a successful model is heavily reliant on knowledge about the particular application at hand. The method in this thesis incorporates both shape information and appearance information into a segmentation procedure that is appropriate for the types of images used.

The two stage procedure proposed here aims to find the eyeballs in a 3D brain MRI image. We use Chan and Vese's active contours without edges segmentation algorithm, combined with shape information about the object we wish to find – the eyeballs. The way in which the model was built can apply to other segmentation problems where a good estimate for the location of the target object and its shape are known in advance – this is particularly common for medical images, where anatomical knowledge can be very helpful in this regard. This information is used in the first stage of our procedure, to help better specify the location of the object we look for. This information is then used in the second stage: using Chan and Vese's method to get an accurate final segmentation. With this final segmentation, medical researchers are able to extract useful information to be utilized in various ways, depending on the particular application.

At the heart of our method lies the splitting of the procedure into two stages: a search for an object of a particular shape, and a full segmentation based on intensity. However, any different segmentation procedure can be used in substitution for the Chan and Vese procedure. For example, the gradient based edge detector algorithm described in the beginning of Chapter 2 can be used instead. Depending on the application at hand, it may be more suitable to use a method other than Chan and Vese's.

Recall that in our method, the shape prior $\psi$ was only allowed to translate over the image domain. This can easily be extended to rotation and scaling of the shape prior. Consider scaling parameter $r$ and rotation angle $\theta$. For 2D, we can rewrite:

$$\psi(x,y) = r\psi_0\left(\frac{(x-a)\cos(\theta)+(y-b)\sin(\theta)}{r}, \frac{-(x-a)\sin(\theta)+(y-b)\cos(\theta)}{r}\right)$$

Now the shape energy can be minimized with respect to the four-tuple $(a,b,r,\theta)$ in a similar fashion as before. Then, at every iteration the current shape prior is updated by transforming $\psi_0$ using the corresponding parameters. The same approach holds in 3D. In fact, different transformations can also be incorporated – eg. one could allow for different scaling in the $x$ and $y$ directions.

For our application and many others the shape prior may not be enough on its own to find the target object. For this reason we introduced the appearance prior into the first stage of our method. For the eyeballs this was simple – we specified that the intensity inside the object is expected to be 0 (black). It is possible to incorporate much more information into the appearance prior. Recall that in the full shape energy, the appearance prior is compared to the original image in the region inside the shape prior's current position. In some applications it may be the case that the object we are looking for is expected to have a particular distribution of intensities. Instead of comparing the intensity of the original image to some constant value incorporated into our appearance prior, we could compare the histograms of the image in the relevant region to that of a previously defined appearance prior. For example, if we are dealing with a mostly medium gray image but expect our object to contain high contrast (many white and black pixels, but few gray ones), we could define an appearance prior with a similar amount of white and black pixels. However,

comparing these directly to the image will be misleading – we do not expect these pixels to coincide precisely, but using a histogram comparison instead should achieve our goal. In this way, our shape prior should favour regions of high contrast over regions of mostly gray pixels. This is only one example of how appearance information can be incorporated into the model.

## 5.2 Extracting Useful Information

Once the final segmentation is obtained, medical researchers can decide on what kind of information they want to extract depending on what goals they want to achieve. With the eyeballs, for example, the volume of individual eyeballs can be calculated from the segmentation $\phi$. Then the volume of one eyeball can be compared to the other, within the same person, or size distribution can be estimated based on a large data set of segmentations. To aid in diagnosis, for example, physicians could track changes in a particular anatomical feature in a patient – change in size, position, even appearance.

For the eyeballs specifically, recall that the final segmentation shows the eyeballs as approximate spheres, but with a "missing" region in the front – the lens. This information can be used to determine the direction in which the person is looking. This may be useful in studies that rely on fMRI data (time series MRI). Any kind of feature can be tracked over time, provided that it does not change position significantly. In some cases it may even be possible to only require the shape stage of the method to be applied to one image, in particular if the various images taken over time are of the same person.

Another area of research in image processing could also benefit from this – image metrics [15]. Defining an image metric involves finding some suitable way to compare images by using a measure of "distance". Some existing image metric approaches start with one segmented object (template) and calculate (in various ways) how much this curve needs to be deformed to fit to another object. Using this information, the distance between these two objects can be defined. Our transformation parameters could aid the definition of some measure of distance between images, with respect to a particular anatomical feature. This can then be used to sort a large set of images by their proximity to some base image that

physicians have deemed to be "normal". In this way images that end up being classified as "far" from normal can be flagged, requiring the immediate attention of a medical professional. Using the information from the segmentation in this way can help increase efficiency and add some objectivity in current practices for diagnosis – it is often subjective for a medical professional to look at an image and decide whether it is normal or not, although the value of a human being's opinion will likely never be replaced in medical applications.

## 5.3 Method Design Improvements

The purpose of this thesis was to provide a proof of concept for the specific procedure for segmentation used here. The results suggest that it is worthwhile to develop this method further, and there is much room for improvement. In particular, the method would greatly benefit from a more automatic cropping procedure for the images. Recall the black space present on the right-hand side of a brain image. With a large number of images, it is possible that the starting position of the shape prior could land exactly in that region, and will stay there, as it is a local minimum. One way this issue could be addressed in a more automatic way that takes the variation across images into account, is to turn the image to binary (using Otsu's method) and locate the right-most white pixel in the image. Now the image can be cropped to there, and we expect that this will eliminate the problem.

Another issue that comes up in many segmentation procedures is parameter selection. In particular, the parameter $\mu$ in Chan and Vese's algorithm is meant to control the length of the segmentation curve (surface area in 3D), and has notable influence on the size of the objects found. We can avoid noise by increasing this parameter, but we run the risk of missing our object entirely if it happens to have small features that are important to us. One idea to address this issue is, for a particular application or type of image, to find some critical value or values of $\mu$ to define some state space based on the largest size of object ignored by the method. This would also improve automation and help any segmentation algorithm to be applied comfortably to a large set of images.

Going back to our choice of shape model, we chose to have this stage not use any guidance from Chan and Vese's method. However, there were a few options mentioned for methods which combine both stages of our procedure into one, by introducing a shape energy term into the Chan and Vese energy, and minimizing the resulting full equation. In [3], a model is built using gradient based segmentation combined with a shape and appearance prior. A similar approach can be constructed with Chan and Vese's segmentation algorithm. All of these changes are potential improvements to our method so far.

## 5.4 Numerical Considerations

As mentioned before, not much emphasis has been placed on choice of numerical scheme. There may be a better choice for minimizing our energy numerically, and the corresponding implementation can be optimized to greatly improve the performance of our method. Improving the stability by implementing a different scheme, or our scheme in its implicit form as in [5], would provide more reliable results. As well, this could go as far as allowing us to use the sign function for updating $\phi$ instead of the traditional gradient descent with higher accuracy. This would be a significant improvement in speed, allowing the method to be applied to a large set of 3D images easily.

One idea for speeding up Chan and Vese's method on its own arises from asymptotics [2]. Recall that Chan and Vese's update for $\phi$ has two different pieces: one is based on the image intensity, and the other is a regularization. In some applications it may be possible to rescale the problem in a way that ends up requiring a small $\mu$ to obtain satisfactory results. In such a case, an asymptotic analysis can be performed on the equation for $\phi$ where $\mu \to 0$, and the final update equation can be split into two pieces: within the iteration loop, we first update $\phi$ according to intensity information only, and then we perform a small motion by mean curvature step to ensure $\phi$'s 0 level curve stays somewhat smooth. Some further analysis on this point could lead to yet another way to speed up the overall process.

Even without these possible improvements, the method at its current stage performs fairly well on small sets of images – even in 3D processing a single image takes seconds. The final segmentation results, however, tend to be rough due to the low quality of the

image – one eyeball has a radius of approximately 8 voxels, not allowing us to resolve the detail of the eyeball very well. In order to make the method able to handle high quality images, however, improvements as suggested above (or otherwise) would be necessary to achieve a final usable product, able to handle many of these high quality images, so that it may become useful in practice for medical research purposes.

# Bibliography


[1] G. Aubert and L. Vese. A variational method in image recovery. *SIAM J. Numer. Anal.*, 34(5):1948–1979, Oct. 1997.

[2] D. Auroux. From restoration by topological gradient to medical image segmentation via an asymptotic expansion. *Mathematical and Computer Modelling*, 49(11-12):2191–2205, 2008.

[3] R. Ben-Ari and D. Aiger. Geodesic active contours with combined shape and appearance priors. *ACIVS*, LNCS 5259:494–505, 2008.

[4] L. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, 1992.

[5] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277.

[6] T. Chan and L. Vese. A multiphase level set framework for image segmentation using the Mumford and Shah model. *Int. J. Comput. Vis.*, 50(3):271–293, 2002.

[7] T. Chan and W. Zhu. Level set based shape prior segmentation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:1164–1170, 2005.

[8] Y. Chen, H. Tagare, S. Thiruvenkadam, F. Huang, D. Wilson, K. Gopinath, R. Briggs, and E. Geiser. Using prior shapes in geometric active contours in a variational framework. *Int. J. of Comp. Vis.*, 50(3):315–328, 2002.

[9] D. Cremers, N. Sochen, and C. Schnorr. Towards recognition-based variational segmentation using shape priors and dynamic labeling. *L. Griffith, M. Lillholm, editors, Int. Conf. on Scale Space Theories in Computer Vision, LNCS*, 2695:388–400, 2003.

[10] F. Gibou and R. Fedkiw. A fast hybrid k-means level set algorithm for segmentation. *4th Annual Hawaii International Conference on Statistics and Mathematics*, pages 281–291, 2005.

[11] R. Gonzalez and R. Woods. *Digital Image Processing*. Prentice Hall, 3 edition, 2008.

[12] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. Comput. Vis.*, 1:321–331, 1988.

[13] M. Leventon, W. Grimson, and O. Gangeras. Statistical shape influence in geodesic active contours. *Comp. Vision and Patt. Recon.*, CVPR, 2000.

[14] G. Maso, J. Morel, and S. Solimini. A variational method in image segmentation: existence and approximation results. *Acta Matematica*, 168:89–151, 1992.

[15] M. Miller, A. Trouve, and L. Younes. On the metrics and euler-lagrange equations of computational anatomy. *Annu. Rev. Biomed. Eng.*, 4:375–405, 2002.

[16] J. Morel and S. Solimini. Segmentation of images by variational methods: a constructive approach. *Revista Matematica Universidad Complutense de Madrid*, 1:169–182, 1988.

[17] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math*, 42:577–685, 1989.

[18] D. Nain, S. Haker, A. Bobick, and A. Tannenbaum. Shape-driven 3D segmentation using spherical wavelets. *Lecture Notes in Computer Science*, 4190:66–74, 2006.

[19] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60:259–268, 1992.

[20] M. Rumpf and B. Wirth. A nonlinear elastic shape averaging approach. *SIAM J. Imaging Sciences*, 2(3):800–833, 2009.

[21] H. Tek and B. Kimia. Image segmentation by reaction–diffusion bubbles. *Proc. Int. Conf. Computer Vision*, pages 156–162, 1995.

[22] C. Xu, D. Pham, and J. Prince. Medical image segmentation using deformable models. *SPIE Handbook on Medical Imaging*, 3:129–174, 2000.

[23] N. Yip. Stochastic motion by mean curvature. *Arch. Rational Mech. Anal.*, 144:313–355, 1998.

[24] H. Zhao, S. Osher, B. Meriman, and M. Kang. A variational level set approach to multiphase motion. *J. Comput. Phys.*, 127:179–195, 1996.