# Abstract

*Active Contour Models:*

*Consistency, Stability, and Parameter Estimation*

*Tianyun Ma*

*Yale University*

*1997*

Active contour models are a very appealing method in image segmentation. In the thesis, we will point out two significant problems of them and develop our solutions.

First, the external energy of an active contours is often formulated as Euclidean arc length integrals. In this thesis, we show that such formulations are biased. By this we mean that the minimum of the external energy does not occur at an image edge. In addition we also show that for certain forms of external energy the active contour is unstable – when initialized at the location where the first variation of the energy is zero, the contour drifts away and becomes jagged. Both of these phenomena are due to the use of Euclidean arc length.

We propose a non-Euclidean arc length which eliminates this problem. This requires a reformulation of active contours where the the global external energy function is replaced by a sequence of local external energy functions and the contour evolves as an integral curve of the gradient of the local energies.

Second, all the active contour models require the user to set a smoothness parameter manually. This step has always been annoying to users. By exploiting image properties near an object boundary, we develop a principle to set the parameter to an appropriate value, and further, we develop an algorithm to set the parameter automatically

and an energy functional needed by the algorithm.

With the two pieces of work, we build a system that finds the object boundary automatically given an initial position near the boundary. Finding a good initial position, or say finding a rough position of the boundary, is a tough problem by itself, which we do not get into in the thesis.

Experimental evidence is provided in support of the theoretical claims. Possible extensions of the work is also presented.

# Active Contour Models:

# Consistency, Stability, and Parameter Estimation

A Dissertation

Presented to the Faculty of the Graduate School

of

Yale University

in Candidacy for the Degree of

Doctor of Philosophy

by

Tianyun Ma

Dissertation Directors: Hemant Tagare and Drew McDermott

May 1998

To My Grandma

# Acknowledgment

I am very lucky to have Hemant Tagare as my advisor in the past a few years. Without his invaluable guidance and great efforts, it would be impossible for me to reach this point. I am grateful to my official advisor Drew McDermott, who directed me to choose the appropriate subject of my graduate work and fulfill all the requirements in the Computer Science Department. I am thankful to Greg Hager and Gene Gindi for serving on my committee, giving me helpful comments, and tolerating many short notices of mine.

This is a joint work with the Image Processing and Analysis Group at Yale. I appreciate the help of the people affiliated with the group: James Duncan lead me to a high starting point of my career; Perry Miller and Albert Sinusas gave me kind help when I needed it badly. Pengcheng Shi wisely advised me to deal with a variety of problems. I must also thank the other members of the group: Ravi Bansal, Eric Bardinet, Amit Chakraborty, Haili Cui, Donald Dione, Christophe Leger, John Mceachen, Carolyn Meloling, Francios Meyer, Xenios Papademetris, Jim Rambo, Anand Rangarajan, Lawrence Staib, Pappu Suguna, Yongmei Wang, Larry Winn, Lei Zhang, Xiaolan Zeng. The friendly atmosphere in the group made my years at Yale enjoyable.

Many people outside Yale also helped me when I faced difficulties, and they made this thesis possible; particularly, I want to thank Anselm Blumer and Mihaly Gereb at Tufts University and Lin Shaw at Microsoft Corporation.

I will always be indebted to my family and friends for their support through my life, and I dedicate this thesis to my grandma, Xiuqin Duan, for her love.

# List of Figures

# List of Tables

# Contents

# Chapter 1

# Introduction

## 1.1 Introduction to the Problem

The goal of *image segmentation* is to identify the boundaries of objects in an image. The boundary information is useful to a variety of image analysis and computer vision applications including pattern recognition, robot vision and biomedical image processing.

To identify the boundary, one needs to distinguish an object from its surroundings. This requires not only the image data but also models as how the object distinguishes itself.

The existing work on image segmentation is divided into two classes: region-based approaches and gradient-based approaches. Both classes have seen tremendous efforts, and some researchers are developing methods to integrate the two classes.

Region-based approaches rely on the homogeneity of image features in an object, and the major variants include thresholding, region growing, Markov Random Fields, scale space filtering, and non-linear diffusion.

In many images where homogeneity is not present, one has to use gradient-based

approaches. The class of approaches rely on the brightness contrast near the boundary, and its major variants include edge detection and grouping, the Hough transform and active contour models. Edge grouping tends to be very difficult since edge detectors always produce spurious edges due to noise. The Hough transform is robust to noise, but the storage and computational complexity are large. Currently, active contour models are a very appealing approach, since it is computationally simple and always provides continuous smooth curves as answers.

There are some significant problems in the most up-to-date active contour model. Our work aims to establish a better active contour model without the problems and provide a guideline to automatically choose a smoothness parameter for active contour models.

## 1.2 Main Contributions

An active contour is an evolving curve that follows gradient dynamics until it is stationary at a minimum of a weighted sum of *external* and *internal* energies. The two energy terms are often expressed as integrals with respect to the Euclidean arc length of the curve.

It is well known that internal energy rounds off corners and produces contours that are different from the true underlying edge. This is a desirable effect, since internal energy is introduced for smoothing (or regularization).

What is poorly appreciated is that external energy can have a similar effect. A badly designed external energy term can consistently make the final curve smaller or bigger. This is not the effect of noise, but is the effect of using Euclidean arc length integrals in the energy term. One way to eliminate bias is to use a non-Euclidean metric for the arc length. We discuss this solution at length.

How serious is the bias? It depends on the scale and strength of the edge. For strong edges which have a fine scale, the bias is small. On the other hand, for weak edges or coarse edges, the bias can be significant.

In medical image processing, the results of active contour algorithms are used for statistical comparisons of normal and abnormal anatomy. For such applications, it is desirable to understand and eliminate any bias irrespective of the edge strength and scale. That is one motivation for this study.

We also investigate the effect of using Euclidean arc length integrals on the stability of classical active contours. We show that certain external energies that have been proposed in the literature actually make the contour unstable. When this contour is initialized at the exact solution (in synthetic images, for example) it drifts away and breaks up into a jagged curve.

Another significant problem with all the active-contour models is that the user has to set a smoothness parameter, which determines how much to smooth the result in a noisy image. Because the range of valid value of the parameter is large, and many active-contour users have no intuition about the function of the parameter, they often find the step annoying. Usually they do it with a trial-and-error procedure: they initially set the parameter to a very small value, and increase it if the result is too jagged; the process goes on until that looks right.

We aim to provide the user a guideline on how to choose the parameter. We first study the structure of an image near an edge. The study finds that active-contour models only use information on the curves while there is a structure of image in their neighborhood. The structure gives a principle to choose the smoothness parameter. We will present the principle in detail in the thesis.

According to the principle, we build up an algorithm to choose the parameter and a new energy functional required by the algorithm. We study the properties of the new energy functional thoroughly.

In the experiments, we demonstrate all the non-desirable effects of active contours we theoretically prove, and compare the new active contour algorithm with existing ones. We also test the parameter estimation algorithm in the experiments.

The two pieces of work – a non-biased active-contour algorithm and a parameter-estimation algorithm – constitute an automatic procedure to find the object boundary given an initial position near the boundary.

## 1.3 Overview

The thesis builds a new active contour model which does not have the problems of the current ones and provides an algorithm to automatically choose the smoothness parameter in active contours. The present chapter serves to introduce the problem of image segmentation and our achievements.

Chapter 2 reviews the relevant literature on both region-based and gradient-based segmentation methods.

Chapter 3 presents the problems with the current active contour models: the most recent model is biased, one variant of it is unstable, and all the models require the user to set a smoothness parameter manually.

Chapter 4 establishes a new active contour model, which we prove to be non-biased. We believe it is stable, though we do not have a proof yet.

Chapter 5 is on the smoothness parameter. It studies the image property near an edge. The study derives a principle to choose the smoothness parameter. After studying

the structure of the set of answers of active contour models, the chapter develops an algorithm step by step.

Both the new active-contour algorithm and the parameter-estimation algorithm are tested on synthetic and real images in chapter 6. We design two performance measures, then we compare both algorithms with existing methods.

The last chapter concludes the thesis and points to possible extensions of our work.

# Chapter 2

# Related Work in Image Segmentation

In the chapter, we briefly describe literature relevant to the work.

The goal of image segmentation is to find object boundaries in images. The problem is critical in low-level vision; therefore, researchers have done intensive work in the field and developed a number of approaches. Fu et. al. [28], Haralick et. al. [37] and Pal et. al. [74] wrote excellent reviews of the field. Books on computer vision and image processing, such as [5, 42, 34, 38, 61, 83], discuss its theory, application and relationship to other fields in details.

Image segmentation methods roughly divide into two classes: region methods and boundary methods. Methods in the first class exploit image properties of the region inside a boundary, usually homogeneity; methods in the second class focus on image features near the boundary, such as sharp change of image intensity. Recently, researchers presented different ways of integrating the two classes of methods [6, 12, 11, 36, 76, 108].

Region and integrated methods are restricted to finding objects with a well-defined

uniformity, which is not present in many images; therefore, boundary methods became one interesting field by itself. Our research focuses on one sub-class of boundary methods, while we briefly introduce both region and boundary methods in the chapter.

## 2.1 Region Methods

Region methods find object boundaries by exploiting image homogeneity of an object. As Pal et. al. [74] pointed out, the principle of this class of methods is: if $I$ is the set of all pixels and $P()$ is a uniformity predicate on groups of connected pixels, the region methods partition the set $I$ into a set of connected subsets or regions ($I_1$, $I_2$, ... , $I_n$) such that

1. $\bigcup_{i=1}^{n} I_i = I$, with $I_i \cap I_j = \phi \ \forall i \neq j$ ,
2. The uniformity predicate $P(I_i) = true$ for all regions $I_i$.
3. $P(I_i \cup I_j) = false$, when $I_i$ is spatially adjacent to $I_j$.

Region methods divide into the following sub-classes:

- Threshhoding

  Global thresholding scheme assumes that pixels whose image intensity is between two thresholds belong to one region. Researchers have developed many methods to choose the threshold [54, 68, 73, 85, 102].

  In images with high noises or uneven brightness, the basic assumption does not hold, and Yanowitz et. al. [103] devised the adaptive thresholding technique to apply to such images.

- Region Growing Methods

  Region growing methods segment an image by combining or splitting regions based

on similarity [7, 39, 71, 77]. Among them, the split and merge techniques [47, 48] are widely used, which plant seeds in the image and let regions grow out from the seeds based on homogeneity properties.

- MRF Related Methods

  With the influential work of Geman & Geman [32], researchers began to use spatial interaction models like Markov Random Field (MRF) to analyze images [24, 25, 32]. Their efforts resulted in several region-based image segmentation methods [6, 23, 24, 25, 30, 32, 33, 51, 60, 67].

- Scale Space Filtering and Non-linear Diffusion Methods

  Scale space filtering methods [3, 40, 49, 101, 104] describe an image in a multi-scale way and embed the original image in a family of derived images $I(x, y, \sigma)$ obtained by convolving the original image $I_0(x, y)$ with a Gaussian kernel $G_\sigma$:

  $$I(x, y, \sigma) = I_0(x, y) \otimes G_\sigma$$

  As researchers pointed out, the family of images can be obtained by the following the isotropic diffusion equation,

  $$\frac{\partial I}{\partial \sigma} = \triangle I \,,$$

  where $\triangle$ is the Laplacian operator.

  As noted in [75], the location of the true boundary is directly available only at the original image. To solve the problem, researchers developed anisotropic diffusion [1, 75, 80], which controls the smoothing by a non-linear function and achieves edge-preserved smoothing.

## 2.2 Boundary Methods

Boundary methods find object boundaries by focusing on the features near them. One commonly used feature is the sharp change of image intensity near an edge. Boundary methods divide into the following sub-classes.

### 2.2.1 Edge Detection and Grouping

An edge detector is a *local* operator to find edges formed by sharp change of image intensity. Researchers have presented a number of edge detectors [34, 38], two of which are very influential – Marr et. al. [62] developed one based on the zero-crossings of the Laplacian of the Gaussian of an image, which produces continuous closed boundaries but is very noise sensitive; Canny [8] developed one based one the gradient of an image convolved with a Gaussian kernel, which often results in broken edges.

Since edge detectors rely on image properties in a local neighborhood of pixels and ignore higher order organization of the image, they make premature decisions and generally produce many broken edges and spurious edges. To solve this problem, researchers have developed algorithms to group edge elements together [27, 70, 82, 100]. But, it is still often difficult to identify and discount spurious edge segments.

### 2.2.2 Hough Transform

The Hough transform is a mapping from the image space to a parameter space. It is originally used to detect straight lines and parameterizable curves such as circles [26], and Ballard [4] extended it to general shapes.

The Hough transform adopts a voting method to find instances of a shape in an image [42]. The algorithm identifies likely boundary points in the image, then each of the

points votes for several combinations of parameters. The combinations getting a majority of votes win, while each winning combination corresponds to one possible instance of the shape,

We illustrate the idea by an example in *Machine Vision* [42]. The problem is to fit a straight line to several image points in image space.

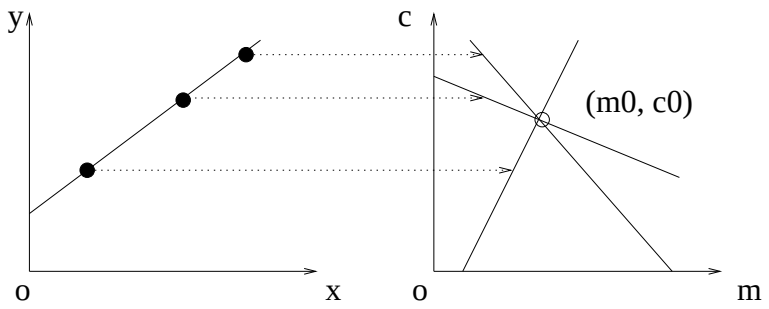



Figure 2.1: Hough Transform

First, we construct the parameter space. Consider the equation of a straight line: $y = mx + c$ , where $(x, y)$ is the coordinate of one point in the image space, and $(m, c)$ represents one combination of the parameters of a straight line. We call the space of $(m, c)$ the *parameter space*. As shown in figure 2.1, one straight line in the image space corresponds to one point in the parameter space, and one point in the image space corresponds to a straight line in the parameter space. Suppose that, in the image space, we have several points on a straight line $y = m_0x + c_0$, as marked by "•" in figure 2.1. Each point is consistent with an infinite set of $(m, c)$ pairs, all of which lie in a straight line in $(m, c)$ space. According to the voting method, each image point votes

for every combination of $m$ and $c$ on its corresponding parameter-space line. Since each image point votes equally, the intersection of all the lines, which is $(m_0, c_0)$, wins the most votes. As desired, the winning combination – $(m_0, c_0)$ – corresponds to the original straight line that fits the image points. To apply the voting algorithm in practice, we need to discretize the parameter space because of noise and round-off errors.

The Hough transform is noise-insensitive, but the storage and computational complexity are large, especially when the curve is complicated.

### 2.2.3 Active Contours

We introduce the rough idea of active contours here. We will use the next whole chapter to discuss the details and problems of the methods, since our work focuses on this sub-class.

In 1988, Kass et. al. [44, 96] introduced *active contour models (snakes)*, which became a widely-used technique to find boundaries and features in images. Considering the image as a spatially varying potential field, the active-contour approach deforms an elastic curve from an initial contour towards the object boundary, and finally fits the curve to the valley of the potential field to provide a continuous boundary to the image features. The curve evolves according to a speed function to minimize an *energy* functional, which consists of two parts – an image-dependent term called *external energy* and a smoothing term called *internal energy*. Both terms are formulated in the form of an integration of some function along the curve with respect to a parameter in $[0, 1]$.

Two major problems exist in the classical approach: (a) according to the speed function, the curve may evolve along its tangential direction; consequently, the points on the curve can keep moving while the shape of the curve does not change; (b) the approach is non-intrinsic – the behavior of the curve depends on how it is parameterized.

Some researchers replace the $[0, 1]$ curve parameter in the energy formulation of

the classical approach by the arc-length parameter [9, 16, 45, 59, 69, 78, 89, 99, 108]. We will call the new model *Euclidean arc-length active contours* or simply *arc-length active contours*. Recently several authors have remarked that the arc-length active contour formulation fits well within a Riemannian framework [9, 45, 59]. The minimizing contour with respect to the energy functional can be interpreted as a geodesic with respect to the Riemannian metric.

The new model avoids the two problems with classical active contours: the curve evolves only along its normal direction and the evolution is independent of parameterization. but as we will point out in next chapter, it incurs new problems.

#### 2.2.3.1 Level Sets

In the original in active contour model, a curve is represented by a list of points, and the curve evolution is described by the change of coordinates of the points. Heuristic procedures have to be used to handle change of topology [9, 94]. Recently, Caselles et. al. [9] and Malladi et. al. [59] implemented curve evolution on a *level set* scheme, which handles change of topology automatically. The idea is following.

Consider figure 2.2. For a closed curve $c$ in the plane $\Phi = 0$, one can construct a function $\Phi(x, y)$ such that $\Phi(x, y) < 0$ inside $c$, $\Phi(x, y) > 0$ outside $c$, and $\Phi(x, y) = 0$ on $c$. The curve $c$ is call the level set of $\Phi$. Two $\Phi$ functions ($\Phi_1$ and $\Phi_2$) and their levels sets ($c_1$ and $c_2$) are drawn in figure 2.2. Suppose that, in the plane of $\Phi = 0$, a curve evolves from $c_1$ to $c_2$ according to some speed function. Instead of evolving the curve, we can achieve the same result by evolving a corresponding function $\Phi$: first, we set $\Phi = \Phi_1$, whose level set is $c_1$; then, we gradually lower value of $\Phi$ to $\Phi_2$, and the level set of $\Phi$ expands gradually from $c_1$ to $c_2$. We can recast desired properties of the evolution from $c_1$ to $c_2$ as properties of the evolution from $\Phi_1$ to $\Phi_2$.

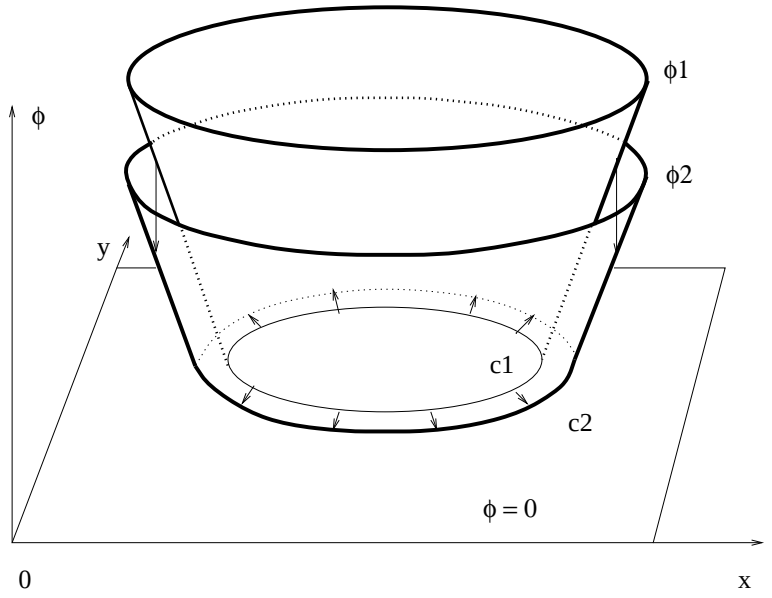



Figure 2.2: Level sets

The advantages of the level-set approach is that it can systematically handle the change of topology and development of shocks. Malladi et. al. [59] gave implementation details in their paper. Kichenassamy et. al. [45] presented the 3-D extension of this framework.

Away from the change of topology and development of shocks, there is a one-to-one correspondence between active-contour evolution and level-set evolution. Since our work is to derive an accurate evolution equation and our work does not involve the change of topology and development of shocks, we will not go into the details of level sets in the thesis. Interested readers will find detailed discussion in [80].

# Chapter 3

# Details and Problems of Current Active Contour Models

In the previous chapter, we briefly reviewed different image-segmentation methods; in this one, we will fix notations, describe different active contour models in details, and point out the problems.

## 3.1 Details of Current Active Contour Models

In this section, we review the energy functional and speed function of different active contour models – classical active contours, arc-length active contours, and geodesic active contours.

### Notations

Let $x, y$ be the image coordinates, so that the image is the function $I(x, y)$. When seeking edges in images, we often need to compute the magnitude of image gradient, which we

will call *gradient magnitude* in the thesis,

$$u(x, y) = || \bigtriangledown I|| \,. \tag{3.1}$$

Given an original image $I_0$, we can compute its gradient magnitude ($u$) by either setting $I = I_0$ or setting $I = G \otimes I_0$ in equation 3.1. $G$ is a Gaussian function to smooth the original image since the gradient operator enhances noise.

A curve $c(q)$ is function from $[0, 1]$ to the image plane. Thus, $c(q) = (c_x(q), c_y(q))$ where $c_x(q)$ and $c_y(q)$ are the coordinate functions of the curve. The arc length parameterization of a curve is $c(s) = (c_x(s), c_y(s))$, where $s$ is the arc-length parameter. Curves with identical coordinate functions after arc length parameterization are considered to be the same curve.

The set of all closed plane curves with finite curvature is the space $\mathcal{C}$. Since plane curves can continuously deform into each other, $\mathcal{C}$ is a manifold. Further, because plane curves which are infinitesimally close to each other can deform into each other by displacement along the curve normals, the tangent space at $c \in \mathcal{C}$ is isomorphic to the set of vector fields on $c$ given by $f(s)n(s)$, where $n$ is the unit normal to the curve $c$ and $f$ is a continuous real function.

We write an evolving curve as $c(q, t)$, where $t$ is time. A continuously evolving curve is itself a curve in the manifold $\mathcal{C}$. The parameter of this curve is $t$, the evolution time.

### 3.1.1 Classical Active Contours and Their Problems

According to Kass et. al., the classical active contour "is a controlled continuity spline under the influence of image forces and internal constraint forces. The internal spline

forces serve to impose a piecewise smoothness constraint. The image forces push the snake toward salient image features like lines, edges, and subjective contours" [44]. In classic active contour algorithms, the energy $E(c)$ associated with the curve is a sum of external and internal energy terms:

$$E = E_{\text{external}} + E_{\text{internal}} \tag{3.2}$$

The external energy term is constructed by creating an image potential $P(x, y)$ and integrating it along $c$ with respect to the curve parameter $q$

$$E_{\text{external}}(c) = \int_0^1 P(c(q))\ dq. \tag{3.3}$$

For classical active contour algorithms that seek edges, the image potential is the negative gradient magnitude. That is, $P(x, y) = -u$ [1]

The most general form of internal energy term is a weighted sum of energy terms in the first and second order derivatives of $c(q)$: $E_{\text{internal}}(c) = \{\alpha||\frac{dc}{dq}||^2 + \beta||\frac{d^2c}{dq^2}||^2\}dq$. However, the $\beta$ term provides a "bending force" [44], which is not desirable in many images; also, adjusting both $\alpha$ and $\beta$ is quite complicated in practice. Many authors have pointed out that using only the energy term in the first order derivative is efficient [9, 46, 59, 88]. Thus, the commonly used internal energy term is

$$E_{\text{internal}}(c) = \alpha||\frac{dc(q)}{dq}||^2 dq\,. \tag{3.4}$$

[1] As we defined in equation 3.1, $u = ||\bigtriangledown I||$.

Adding external and internal energy terms together, we get

$$E_0(c) = \int_0^1 \{P + \alpha||\frac{dc}{dq}||^2\}dq\,. \tag{3.5}$$

Using gradient descent method to lower the energy, we derive the curve evolution equation as the following:

$$\frac{\partial c}{\partial t} = -\lambda\frac{dE_0}{dc} = -\lambda(\bigtriangledown P + \alpha\frac{d^2c}{dq^2})\,, \tag{3.6}$$

where $\lambda$ is a constant to adjust the step size in evolution.

According to equation 3.6, the curve evolves to minimize potential function $P$ (to maximize gradient magnitude) while smoothing itself. The stationary point of evolution (the condition under which the curve stops evolving) is given by $\frac{\partial c}{\partial t}(q) = 0$.

Two major problems exists in the classical approach:

- When we re-parameterize the curve on the same image, $\bigtriangledown P$ will remain the same while $\frac{d^2c}{dq^2}$ will change for some or all points on the curve; therefore, $\frac{\partial c}{\partial t}$ depends on the parameterization. As a consequence, when a curve gets to a stationary point, it may evolve again after we re-parameterize it.

- $\bigtriangledown P$ has both normal and tangential components to the curve. Suppose that there is a smooth valley in $P$, and $P$ is not a constant along the bottom of the valley; then $\bigtriangledown P$ at the bottom is always tangential to the bottom. After we initialize the curve at the bottom of the valley and set $\alpha = 0$. the points on the curve will keep on drifting along the bottom (the tangential direction of the curve); thus, the shape of curve does not change while the evolution goes on forever. To avoid the

problem, we need to set $\alpha$ to a positive value even if the image is noiseless. The internal energy plays a double role – it smoothes the curve and it stops the curve from drifting along itself forever.

### 3.1.2 Euclidean Arc-length Active Contours

Many researchers have proposed a simple alternative approach which solves the two problems with classical active contours. We call the new approach *Euclidean arc-length active contours* or simply as *arc-length active contours*, because the main idea is to replace the arbitrary parameter $q$ in the energy formulation by the arc-length parameter of the curve [9, 16, 45, 59, 69, 78, 89, 99, 108]. The substitution leads to a new energy functional:

$$E(c) = \int_0^l \{P(c(s)) + \alpha||\frac{dc}{ds}||^2\}ds\,,$$

Since $s$ is the arc-length parameter, $||\frac{dc}{ds}|| = 1$; the internal energy term is just proportional to the total arc-length of the curve:

$$E(c) = \int_0^l \{P(c(s)) + \alpha\}ds\,, \tag{3.7}$$

When this is used to seed edges, different researchers define the potential function $P$ in alternate ways. Cohen and Kimmel[16], Neuenschwander and Fua et al. [69], Sapiro[89], and Williams and Shah[99], define $P()$ to be the negative gradient magnitude

$$P(x, y) = -||\bigtriangledown I||\,. \tag{3.8}$$

In a sense, this is the most intuitive image potential that can make the contour cling to edges. On the other hand, Caselles, Kimmel and Sapiro[9], Malladi, Sethian and Vemuri [59] and Kichenassamy et al.[45] propose

$$P = \frac{1}{1 + || \bigtriangledown I||^n} \tag{3.9}$$

where, $n = 1 \text{or } 2$. Finally, Malladi, Sethian and Vemuri [59] also propose

$$P = \exp\left(-|| \bigtriangledown I||\right). \tag{3.10}$$

In all three definitions, $P$ is in the form of $P(x, y) = P(u(x, y))$ [2] :we can consider $P$ as a function of $x$ and $y$ and write it as $P(x, y)$; we can also consider $P$ as a function of $u$ and write it as $P(u)$. The three definitions in the later form are:

$$P(u) = -u \tag{3.11}$$

$$P(u) = \frac{1}{1 + u^n} \tag{3.12}$$

$$P(u) = \exp(-u) \tag{3.13}$$

They all have the following properties:

- The value of $P(u)$, for $u > 0$, is either always greater than zero or always less than zero. We will use this property in our analysis of stability.
- $P' = \frac{dP}{du} < 0$ :$P$ is a monotonically decreasing function of $u$.

[2] As we defined in equation 3.1, $u = || \bigtriangledown I||$.

- $\bigtriangledown P = (\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}) = P' \bigtriangledown u$ :The direction of the gradient of $P$ is opposite to the direction of the gradient of $u$. When the curve evolves to lower $P$, it is evolving to higher $u$, which is the gradient magnitude.

For computation convenience, we express the energy $E(c)$ in terms of an arbitrary parameterization of the curve. If $c(q)$ is an arbitrary parameterization of the curve for $q \in [0, 1]$, then

$$\begin{aligned} E(c) &= E_{\text{external}} + E_{\text{internal}} \\ &= \int_0^1 \{P(c(q)) + \alpha\} || \frac{dc(q)}{dq} || dq. \end{aligned} \tag{3.14}$$

Again, the evolution of the arc-length active contour follows gradient dynamics: the curve evolves in a direction that achieves the maximum rate of decrease in its energy:

$$\frac{\partial c(q,t)}{\partial t} = \omega(q) n(q) = -(\bigtriangledown P \cdot n - P\kappa - \alpha\kappa) n. \tag{3.15}$$

In this equation, the term $\bigtriangledown P \cdot n$ is the inner product of the vector $\bigtriangledown P$ with the normal $n$, and $\kappa$ is the curvature function of $c$.

The stationary point of evolution (the condition under which the curve stops evolving) is given by $\omega(q) = 0$. The function $\omega()$ is sometimes called the *speed function* of the arc-length active contour.

### Geodesic active contours

Recently several authors have remarked that the active contour formulation fits well within a Riemannian framework [9, 45, 59]. This is based on the following observation:

Let $\phi(x,y)$ be a positive function. Then, $ds' = \phi(x,y)\ \sqrt{(dx)^2 + (dy)^2}$ is a Riemannian metric defined in the image plane. The Riemannian arc length of a curve $c(q)$ is

$$\begin{aligned} L_\phi(c) &= \int_0^1 \phi(c(q))\ \sqrt{\left(\frac{dc_x(q)}{dq}\right)^2 + \left(\frac{dc_y(q)}{dq}\right)^2}\ dq \\ &= \int_0^1 \phi(c(q))\ ||\frac{dc(q)}{dq}||\ dq, \end{aligned}$$

where, $||\frac{dc(q)}{dq}||dq$ is the normal Euclidean arc length $ds$ of the curve. By setting

$$\phi(x,y) = P(x,y) + \alpha, \tag{3.16}$$

we obtain

$$L_\phi(c) = E(c). \tag{3.17}$$

Thus, the minimizing contour with respect to $E(c)$ can be interpreted as a geodesic with respect to the Riemannian metric.

The appeal of this interpretation is that the theory of geodesics becomes available for analyzing active contours. This framework is sometimes called *geodesic active contours* [9].

Since geodesic active contour and arc-length active contour have the same form of objective function 3.17. our analysis of arc-length active contour in the thesis covers geodesic active contours as well. Therefore, we will not consider geodesic active contours separately.

## 3.2 Problem 1 of Arc-Length Active Contours: Bias due to Euclidean Arc-Length Integration

In the section, we will describe the bias problem with arc-length active contours.

### 3.2.1 Examples to Illustrate the Bias

We describe the bias intuitively with the example in figure 3.1, which contains a uniform white disc on a gray background. Image intensity $I$ along line $AB$ is shown in the figure too. It doesn't matter whether $P$ is defined as $P(u) = -u$, $P(u) = \frac{1}{1+u^n}$ or $P = e^{-u}$; in any case, its shape looks like a valley along the radius of the disc near the boundary of the disc, as shown in the figure.

In this image, the true boundary is the circle surrounding the white disc, which lies in the bottom of the valley of $P$. Suppose we initialize a curve at the correct position, and run it according to the evolution equation for arc-length active contours (equation 3.15). Since this image is noiseless, we set $\alpha = 0$. Depending on the formulation of $P$, the curve will drift away from the correct position, either inside to a smaller circle or outside to a larger circle. We explain this phenomenon in two aspects.

First, we can explain it in terms of speed function. Since the curve is at the bottom of $P$, the value of $P$ is minimal along the normal direction, so $\bigtriangledown P \cdot \vec{N} = 0$; while $P \neq 0$ on the curve in all three definitions. The curve evolves according to equation 3.15, so the absolute value of its speed function is

$$||\frac{\partial c}{\partial t}|| = |-\lambda(0 + P\kappa\vec{N} + 0)| \neq 0\,,$$

therefore the curve drifts away. Whether it drifts inside or outside depends on the sign

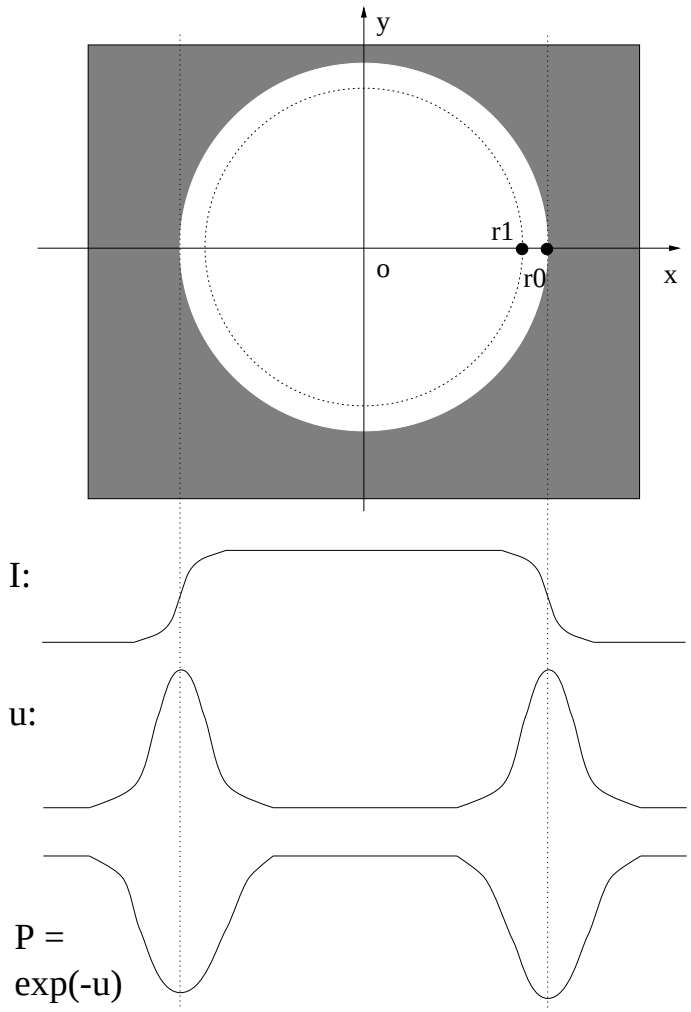



Figure 3.1: A white disc on gray background

of $P$ at the bottom.

Second, we can explain it in terms of energy. Since this image is noiseless, we ignore the internal energy term by setting $\alpha = 0$, so that the behavior of active contours

is determined by the external energy alone. Because this simple image is circularly symmetric, we get $E = \int_0^L P(c(s))ds = Pl$ from equation 3.7, where $l$ is the total arc length of the circle, $P$ is the value of potential function on the curve. Without loss of generality, let us consider the case $P = e^{-u}$, which is always positive. Imagine the active contour shrinks itself uniformly to the inside circle shown in the image; as a consequence, $P$ increases, which causes the external energy to go up; but the length of the curve gets shorter, which causes the energy to go down. Depending on the actual value of $P$ and $l$, the external function can decrease. So, the active contour can find a position in the neighborhood of the true boundary with lower external energy. The curve drifts inside to that position. If $P$ is defined as $P = -u$, which is negative near the bottom, the curve will drift outside. From the analysis above, we can see the source of bias: when the curve evolves along its normal direction, the length of the curve changes, and the change has effects in the external energy.

The example in figure 3.1 has one disadvantage. It suggests that bias originates from total change in the arc length of the contour. This is not the case. Bias is a local property of Euclidean arc length formulations. An infinitesimal version of the above example shows this clearly (figure 3.2). If an infinitesimal piece of a curve $c$ has Euclidean arc length $ds$, then its infinitesimal external energy is the product $Pds$. If we push this infinitesimal piece in the normal direction by the amount $dv$, then $P$ changes by the amount $P' \frac{\partial ||\bigtriangledown I||}{\partial v} du$ while $ds$ changes by the amount[3] $-\kappa ds dv$. Therefore, up to first order

Change in the contribution to external energy

[3]The change in the infinitesimal arc length is explained concisely by Morgan [65].

$$= \text{First term} \times \text{change in second term} + \text{Second term} \times \text{change in first term}$$

$$= P' \frac{\partial || \bigtriangledown I||}{\partial v} dvds - P\kappa dsdv.$$

Suppose the curve was initially located exactly over an edge in the image. Since the gradient magnitude is locally maximized in the direction normal to the edge, $\frac{\partial ||\bigtriangledown I||}{\partial v}$ is zero. Therefore, the first term in the above expression is zero; and the change is $-P\kappa dsdv$. Hence, the rate of change of external energy with respect to $v$ is $-P\kappa ds$, which is of the same order as the external energy of the infinitesimal piece. Therefore, the external energy of the curve can decrease when it is moved in the normal direction away from the edge.

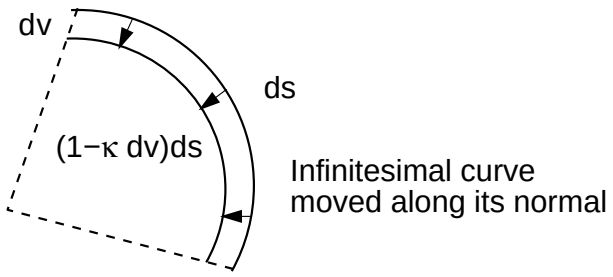



Figure 3.2: The change in arc length of an infinitesimal curve.

This argument shows that bias due to the external energy arises as a result of the change in Euclidean arc length when a curve is moved normal to itself. The argument is made more precise in the next sub-section.

### 3.2.2 Bias and Consistency: the Rigorous Formulation

In this sub-section, we define when an external energy is consistent and when it is biased. Then we prove that external energy functions with Euclidean arc length integrals are biased.

Let $c^*(q)$ be a plane curve with unit normals $n(q)$. An image $I(x, y)$ has an *edge* on the curve $c^*$ if for all $q$

$$\frac{\partial}{\partial n(q)} \, || \bigtriangledown I(x,y)|| \quad = \quad 0, \tag{3.18}$$

and,

$$\frac{\partial^2}{\partial n^2(q)} \, || \bigtriangledown I(x,y)|| \quad < \quad 0, \tag{3.19}$$

The symbols $\frac{\partial}{\partial n(q)}$ and $\frac{\partial^2}{\partial n^2(q)}$ denote the first and second order partial derivatives in the direction of the normal vector $n(q)$. Note that this definition does not assume that the contrast of the edge (i.e. $|| \bigtriangledown I||$) is uniform along $c^*$.

Let $\mathcal{I}_{c^*}$ be the set of images which have an edge on $c^*$. This is the set of images that we will be concerned with.

The external energy of an active contour is consistent when it makes the contour stationary at an edge. More formally:

**Definition:** The external energy $E(c)$ of an active contour is *consistent* if the first variation of $E(c)$ is zero at $c = c^*$ for any $I \in \mathcal{I}_{c^*}$. Otherwise, the active contour is *biased*.

Where the first variation is defined as the following: suppose that a functional $E$ is defined in an open subset $Y$ of a normed linear space $S$, then $\delta E(h)$ is called the first

variation of $E(y)$ at $y = y_0$ if, for $t \in R$,

$$\delta E(h) = \frac{d}{dt} E_{|t=0}(y_0 + th)$$

exists for all $h \in S$ [84].

We can establish the following result

**Proposition 1:** $p(u)$ is a real function. If for any $u$, $p$ is either always greater than zero or always less than zero, then any external energy of the type

$$E(c) = \int_0^1 p(|| \bigtriangledown I(c(q))||) \; ||\frac{dc(q)}{dq}|| \; dq,$$

is biased.

**Remark:** The $p(u)$'s proposed in the literature are either greater than or less than zero. Therefore, this theorem applies to all of them.

**Proof of Proposition 1:** The proof is an evaluation of the first variation of of $E(c)$ at $c = c^*$ for any $I \in \mathcal{I}_{c^*}$. Letting $v = f(q)n(q)$ be an element of the tangent space of $\mathcal{C}$ at $c^*$, the first variation $\partial E(c^*)[]$ is given by $\partial E(c^*)[v] = \int_0^1 g(q)f(q)dq$, where,

$$\begin{aligned} g(q) &= ||\frac{dc^*(q)}{dq}||\{\frac{dP(|| \bigtriangledown I(c(q))||)}{d|| \bigtriangledown I||} \; \frac{\partial || \bigtriangledown I(c^*(q))||}{\partial n(q)} \\ &\quad -P(|| \bigtriangledown I(c(q))||) \; \kappa(q)\} \\ &= ||\frac{dc^*(q)}{dq}|| \left\{-P(|| \bigtriangledown I(c(q))||) \; \kappa(q)\right\}, \end{aligned}$$

where, $\kappa(q)$ is the curvature of $c^*$, and we have used the fact that $\frac{\partial || \bigtriangledown I(c^*(q))||}{\partial n(q)} = 0$ because

$I \in \mathcal{I}_{c^*}$.

The first variation $\partial E(C^*)[]$ is zero if and only if $g(p)$ is the zero function. However, $g(p)$ cannot be a zero function because: (1) $||\frac{dc^*(q)}{dq}||$ cannot be zero for a regular parameterization, (2) $P$ is either greater or less than 0 and hence $P \neq 0$, and (3) $c^*$ is a closed curve and hence $\kappa(q)$ cannot be zero for all $q$.

This completes the proof.

## 3.3 Problem 2 of Arc-Length Active Contours : Instability with Certain Formulation

We now turn to the issue of stability of active contours whose external energy is an Euclidean arc length integral. The issue of stability is the issue of deciding if a stationary point of the energy functional is a true minimum.

### 3.3.1 Mathematical Reasoning

Determination of a true minimum in a variational problem is a complicated issue and to resolve it one has to take into account various pathologies of infinite dimensional function spaces. Such investigation is beyond the scope of this paper. Instead we focus on the *Legendre condition*. This is a necessary (but not sufficient) condition for the existence of a (weak-) local minimum. That is, when the Legendre condition is violated, we may be sure that the solution of the Euler-Lagrange equation is not a minimum. However, when the condition holds, we cannot conclude that the solution is a minimum.

The Legendre condition states that if the functions $\phi(q)$ and $\psi(q)$ make the inte-

gral

$$E(\phi,\psi)=\int_0^1 F(q,\phi,\psi,\frac{d\phi}{dq},\frac{d\psi}{dq})\ dq$$

minimal, then the matrix

$$\mathcal{F}=\begin{pmatrix} F_{\frac{d\phi}{dq}\frac{d\phi}{dq}} & F_{\frac{d\phi}{dq}\frac{d\psi}{dq}} \\ F_{\frac{d\psi}{dq}\frac{d\phi}{dq}} & F_{\frac{d\psi}{dq}\frac{d\psi}{dq}} \end{pmatrix}$$

evaluated at $\phi(q),\psi(q)$ is positive semi-definite for all $q$ ([20], 214-216). The term $F_{\frac{d\phi}{dq}\frac{d\phi}{dq}}$ is the partial second derivative of $F$ with respect to its argument $\frac{d\phi}{dq}$, the term $F_{\frac{d\phi}{dq}\frac{d\psi}{dq}}$ is the mixed partial derivative of $F$ with respect to its arguments $\frac{d\phi}{dq}$ and $\frac{d\psi}{dq}$, and so on.

To evaluate the Legendre condition, we write the Euclidean arc length integral energy function of equation (3.14) as

$$E(c)=\int_0^1 F(q,c_{x_1},c_{x_2},\frac{dc_{x_1}}{dq},\frac{dc_{x_2}}{dq})\ dq, \tag{3.20}$$

where,

$$F=(P(||\bigtriangledown I(c_{x_1},c_{x_2})||)+\alpha)\sqrt{(\frac{dc_{x_1}}{dq})^2+(\frac{dc_{x_2}}{dq})^2}.$$

The resulting $\mathcal{F}$ matrix is

$$\mathcal{F}=\frac{P(||\bigtriangledown I(c_{x_1},c_{x_2})||)+\alpha}{\left((\frac{dc_{x_1}}{dq})^2+(\frac{dc_{x_2}}{dq})^2\right)^{3/2}}\begin{pmatrix} (\frac{dc_{x_2}}{dq})^2 & -\frac{dc_{x_1}}{dq}\frac{dc_{x_2}}{dq} \\ -\frac{dc_{x_1}}{dq}\frac{dc_{x_2}}{dq} & (\frac{dc_{x_1}}{dq})^2 \end{pmatrix}.$$

The two eigenvalues of $\mathcal{F}$ are

$$\begin{aligned}\lambda_1 &= 0, \text{ and} \\ \lambda_2 &= \frac{P(||\bigtriangledown I(c_{x_1}, c_{x_2})||) + \alpha}{\sqrt{(\frac{dc_{x_1}}{dq})^2 + (\frac{dc_{x_2}}{dq})^2}}.\end{aligned}$$

Thus, $\mathcal{F}$ is positive semi-definite if and only if

$$P(||\bigtriangledown I(c_{x_1}, c_{x_2})||) + \alpha \geq 0$$

for all $q$.

Hence, we reach the following conclusion:

**Proposition 2:** An active contour with the energy function of equation (3.20) is unstable at its stationary position $c(q)$ if

$$P(||\bigtriangledown I(c_{x_1}, c_{x_2})||) + \alpha < 0 \tag{3.21}$$

for some $q$.

We can use this result to evaluate the effect of different $P$ functions on the stability of the active contour.

[1] $\mathbf{P}(\mathbf{u}) = -\mathbf{u}$: This is the most interesting case. The condition (3.21) reduces to

$$-||\bigtriangledown I(c_{x_1}, c_{x_2})|| + \alpha < 0$$

which can certainly hold for some $q$ if $\alpha$ is not large enough. Therefore active contours with this external energy functions are unstable if $\alpha$ is not numerically greater than the largest value of $||\bigtriangledown I(c_{x_1}, c_{x_2})||$. This is surprising in light of the common belief that

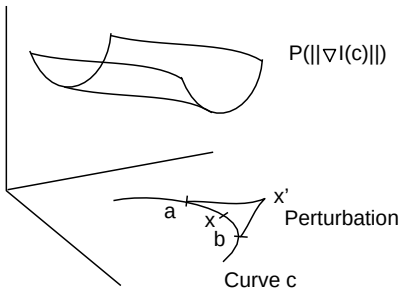
$P(u) = -u$ is the intuitively simplest function which can make an active contour cling to an edge.

We can try to make the contour stable by increasing $\alpha$, but the Legendre condition does not give the value to which $\alpha$ must be increased.

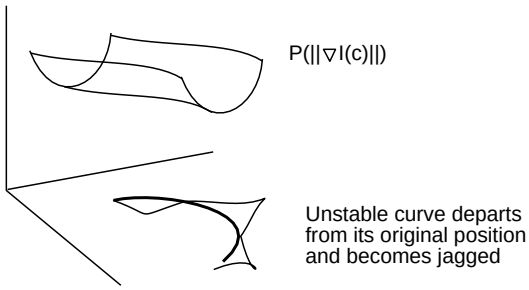
[2] $\mathbf{P(u)} = \frac{\mathbf{1}}{\mathbf{1+u^n}}$ or $\mathbf{P(u)} = \exp(-\mathbf{u})$ : In either case, $P(|| \bigtriangledown I(c_{x_1}, c_{x_2})||)$ is greater than zero and the condition of proposition 2 is not satisfied. As we emphasized above, this does not prove that active contours with these $P$'s are stable. It only means that the Legendre condition is unable to resolve the issue of stability and a more refined analysis is required.

To get further insight into the instability of the active contour for $P(u) = -u$ consider figure 3.3. The figure shows the function $P(|| \bigtriangledown I||)$ and a curve $c$ which satisfies the Euler-Lagrange equation. Let $[a, b]$ be a closed curve segment of length $\lambda$ and let $x$ be the mid point of $[a, b]$. Perturb the segment $[axb]$ as shown in the figure by moving $x$ normal to the curve to $x'$ in the direction opposite to the osculating circle at $x$. Let the length of the perturbed segment $[ax'b]$ be $\lambda'$. Since the perturbation is in the opposite direction to the osculating circle, $\lambda' > \lambda$.

Consider what happens to the energy of the curve due to this perturbation. Set $\alpha$ to zero for now. The external energy of the curve outside the perturbed segment remains the same. If $\lambda$ is small enough, the initial external energy of $[axb]$ is approximately $-\lambda|| \bigtriangledown I(x)||$. After perturbation the Euclidean arc length of $[ax'b]$ increases to $\lambda'$ while the value of $|| \bigtriangledown I||$ decreases to $|| \bigtriangledown I(x')||$. As we argued before, it is easy to create a profile for $\bigtriangledown I$ such that the ratio $|| \bigtriangledown I(x')||/|| \bigtriangledown I(x)||$ is greater than the ratio $\lambda/\lambda'$. In that case, the external energy of the perturbed contour $(= -\lambda'|| \bigtriangledown I(x')||)$ is lower than the external energy of the initial contour.




Perturbation around the Euler−Lagrange solution




The shape of an unstable solution

Figure 3.3: Instability and the Legendre condition.

If the second derivative of the energy is negative as the contour is moved from $[axb]$ to $[ax'b]$, then the contour is unstable at $c$. An unstable contour will drift away from $c$ by becoming more jagged as it acquires a number of "triangular" perturbations of the sort shown in figure 3.3, each perturbation decreasing the external energy by increasing arc length.

The Legendre condition is the evaluation of the second derivative for certain "triangular" perturbations that are similar to the one in figure 3.3 (for details, see Courant and Hilbert ([20],214-216)). Thus, equation (3.21) tells us precisely when the second derivative of the energy with respect to the perturbation becomes negative, even if $\alpha$ is not zero.

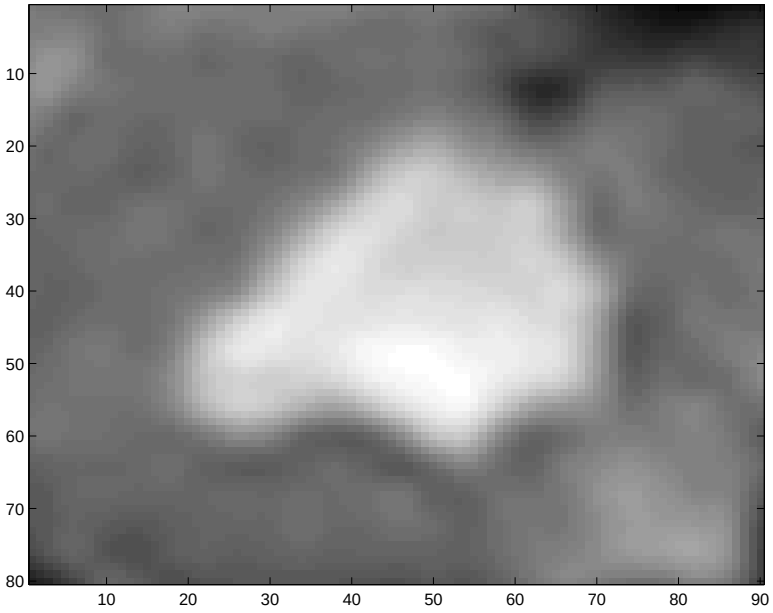
This argument makes it intuitively clear that the instability is due to the increase in Euclidean arc length from $\lambda$ to $\lambda'$ as the curve is deformed. Intuitively speaking, replacing the Euclidean arc length with the non-Euclidean arc length of section should eliminate this problem. Indeed, the local energy function we propose in next chapter does not satisfy condition (3.21). However, we do not yet have a complete proof of the stability of these active contours. It remains an open problem.

### 3.3.2 Instability Demonstrated on a Real Image

The instability of $P(u) = -u$ can be observed in practice. The telltale sign of the instability is that the active contour becomes jagged as it tries to decrease its external energy by cramming ever greater arc length into the "ravine" formed by $P(||\bigtriangledown I||)$. Figure 3.4.(a) is one MRI image of a dog heart, and figure 3.4.(b) shows the potential image ($P$) of the MRI image.

An active contour was initialized to the true edge of the image. The initial position is shown in figure 3.4(b). To check that the initial position was close to the true edge,

(a) MRI image




(b) $P = -|| \bigtriangledown I||$

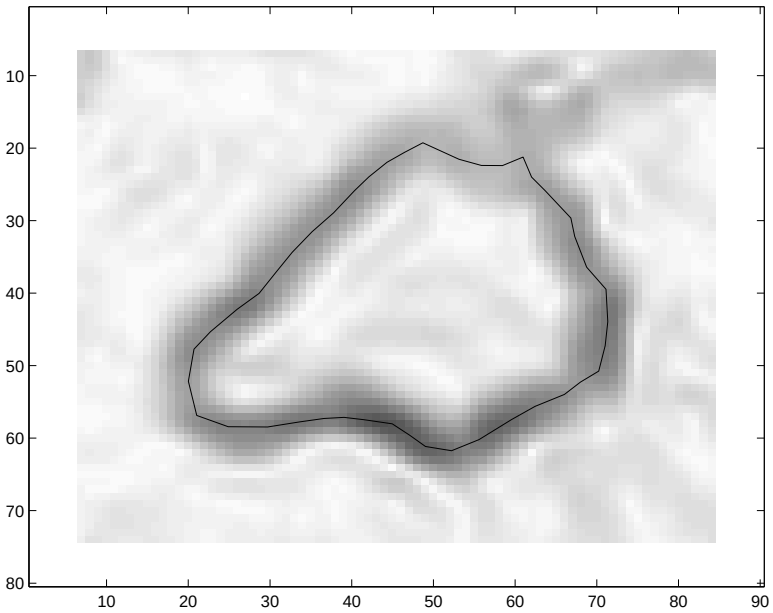



Figure 3.4: normals

we plotted the variation of $P = -|| \bigtriangledown I||$ along 10 evenly placed normals to the contour. as shown in figure 3.5. The middle point in each mini-figure represents one point on the curve. We can see that the value of $P$ at each point on the curve is lower than the value of $P$ at its neighbor points along the normal. Since $P = -|| \bigtriangledown I||$, this means that the contour sits on the edge.

From this initial position, the contour was evolved along the gradient of its energy for $\alpha = 0.1, 5, 10$ and 15. The external energy was the Euclidean arc length integral with $P = -|| \bigtriangledown I||$. The results are shown in figure 3.6. We observed that for $\alpha = 0.1$ and 5 the contour was unstable. It kept evolving into a more jagged curve thereby increasing its arc-length and decreasing its energy. As expected, the tendency to become unstable and jagged decreases with increasing $\alpha$, and at $\alpha = 20$ the contour appears to have stabilized.

This example not only illustrates the instability in the active contour but also the difficulty in finding the stabilizing $\alpha$.

## 3.4 Parameter Estimation

While using active contour models, the user has to set some smoothness parameters; for example, the user needs to set an appropriate $\alpha$ in equation 3.15 to evolve the curve in a desired way. If the user sets it too small, the results will be more jagged than the correct answers because of noise in the image; if the user sets it too large, the results will be overly smoothed, and in some cases they will even collapse to a point. Because the range of the valid value of the parameter is very large and the appropriate value varies across different images, the user usually has difficulty finding the right parameter. One common approach is trial-and-error: the user begins with a small parameter, then he or she increases it gradually until the result curve looks good.

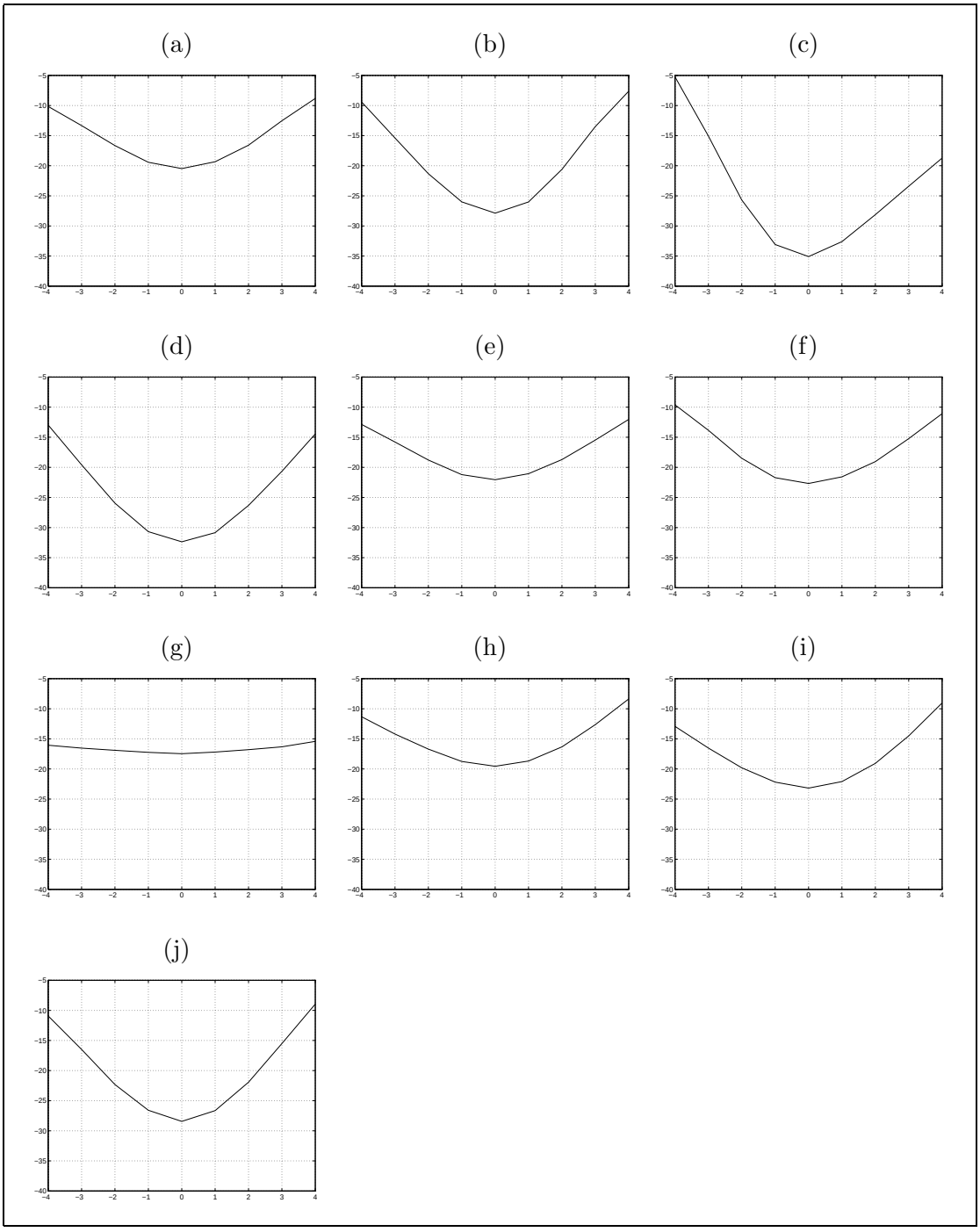



Figure 3.5: $P$ value along ten normals of the curve.

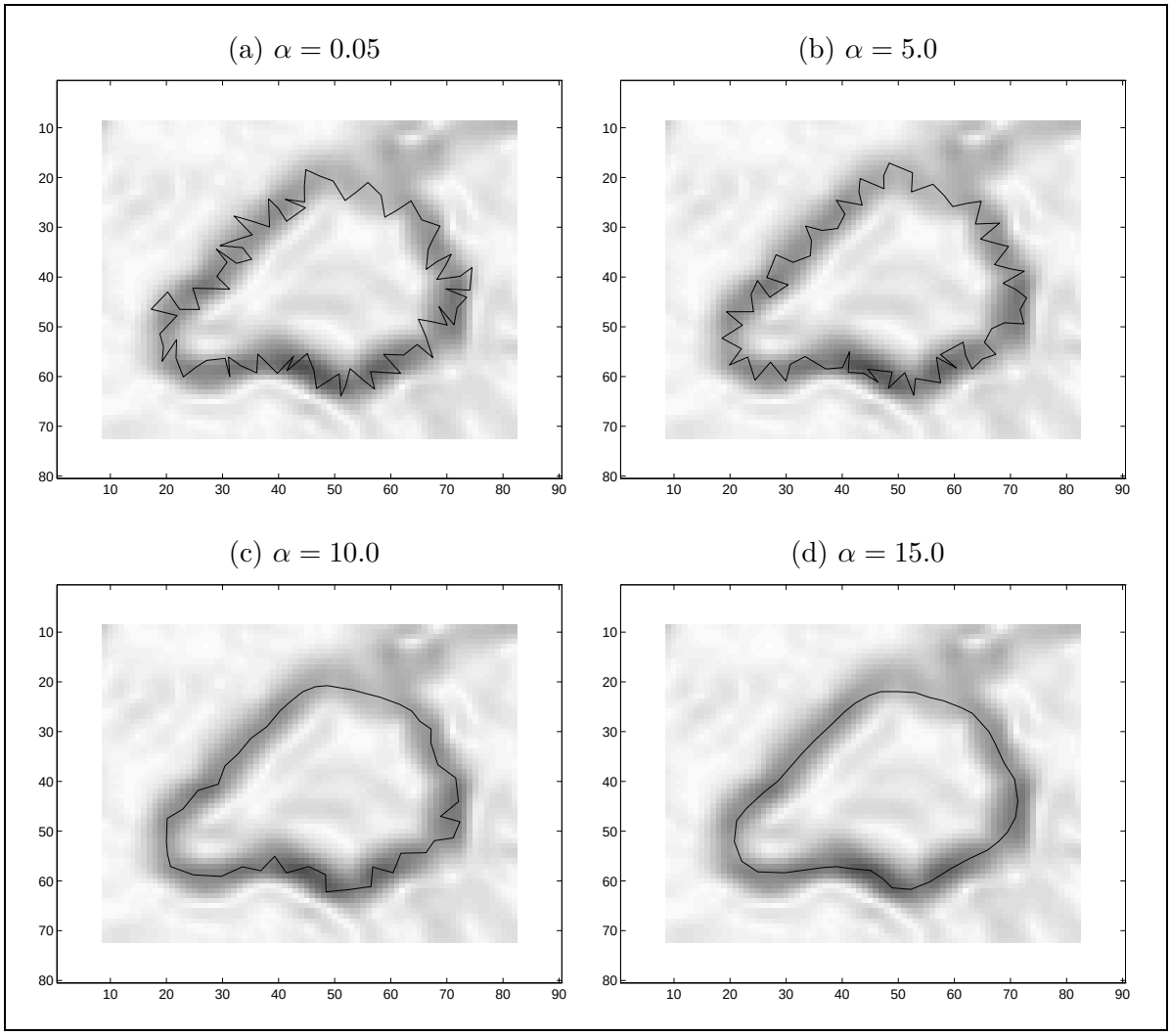



Figure 3.6: $P = -|| \bigtriangledown I||$ with different $\alpha$'s

### 3.4.1 Related Work

Researchers have tried to systematically find appropriate smoothness parameters in different ways. Samadani [86] presented a method to dynamically estimate and adjust the parameters to avoid instability in the deformation process. Larsen et. al. [52, 53] established bounds of optimal parameters by extracting information from the object model. Davatzikos et. al. [21, 22] established a special kind of active contour – the ribbon model, then they studied the uniqueness and fidelity of the results, and their study gave bounds on the parameters.

All the proposed methods only gave bounds of the parameter, and they usually apply to a specific class of images. In the thesis, we will present a more general approach to set the smoothness parameter, which exploits the image structure in the neighborhood of an object contour and establishes a principle to find the optimal smoothness parameter.

# Chapter 4

# A New Active Contour

In the previous chapter, we described that the external energy term of arc-length active contour is biased and some forms of it are unstable; In this one, we will define a consistent external energy functional. How to achieve it? Since the origin of the bias is due to the use Euclidean arc lengths, one possibility is to use non-Euclidean arc lengths.

## 4.1 A New Energy Functional

Getting rid of the bias requires a reformulation of active contours. The new formulation can be explained in two steps:

1. Given a curve $c$, we change the definition of arc length in its neighborhood such that when any infinitesimal piece of $c$ is moved normal to itself, its (new) arc length is unchanged.

   The new arc length metric depends on $c$ and is valid only in a small neighborhood of $c$ (figure 4.1). It might appear that this definition is of limited use because it only applies to curves close to $c$. But this is not really a problem. We observe that

active contours evolve smoothly, so that at any instance, the next contour is always in a small neighborhood of the current contour and the new definition of arc length can be applied.

More precisely, every curve $c \in \mathcal{C}$ has a neighborhood in which the arc length definition based on $c$ is valid. Using this definition we can formulate a "local" external energy function and decide the best direction in which to evolve $c$. This gives a vector in the tangent space of $c$ (figure 4.1). Repeating this for every $c \in \mathcal{C}$ defines a vector field on $\mathcal{C}$. The evolution of a specific contour $c$ is simply the integral curve of the vector field passing through $c$. The evolution stops when it reaches a stationary point of the field.

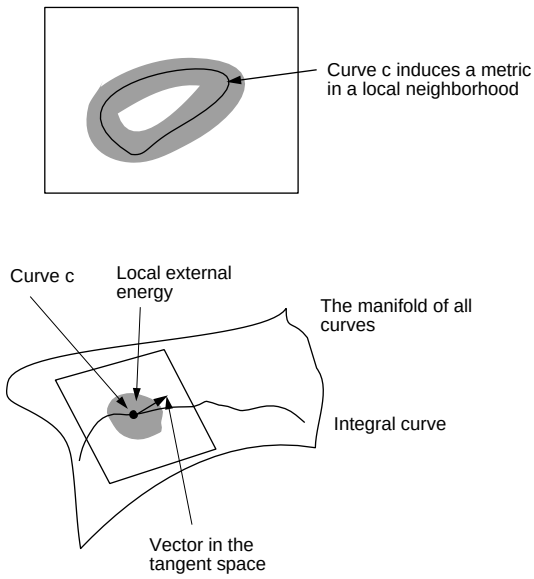



Figure 4.1: Evolving a curve by a local external energy.

2. Internal energy can be added as before without any further modifications. We simply set each "local" energy to be a weighted sum of the new external energy and the old internal energy. As above, for any curve $c$, we compute the direction that gives the fastest decrease in the local energy. This gives a vector field on $\mathcal{C}$ and the integral curve of the vector passing through $c$ gives the evolution of $c$.

This formulation of active contours is fundamentally different from earlier formulations in that it does not contain a globally defined external energy, but instead has a vector field induced from a locally defined energy. The active contour no longer seeks an minimum of a function; but instead seeks a stationary point of a vector field.

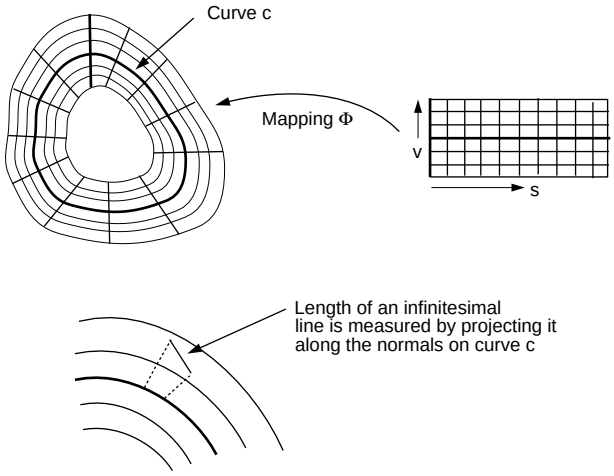



Figure 4.2: Non-Euclidean arc length.

We proceed with the details of the each of the above steps. Let $c(s)$ be an arc length parameterized curve with normal $n(s)$ and total Euclidean arc length $L$ (figure 4.2). If $v$ is the distance along the normal, then for $-D \leq v \leq D$, and $0 \leq s \leq L$ the points $c(s) + v\ n(s)$ belong to a neighborhood of the curve, say $\Omega_c$. In fact, we can choose $D$ and $\Omega_c$ small enough that the map $\Phi : [0, L] \times [-D, D] \to \Omega_c$ given by

$$\Phi(s, v) = c(s) + v\ n(s),$$

is a diffeomorphism. The map $\Phi$ takes a rectangular grid in $[0, L] \times [-D, D]$ and conformally maps it onto $\Omega_c$ such that the horizontal lines of the grid are mapped parallel to $c$ and the vertical lines are mapped perpendicular to $c$.

Evaluating the inverse map $(s, v) = \Phi^{-1}(x_1, x_2)$ corresponds to projecting the point $x_1, x_2$ on to the curve $c$ along the normal of the curve. The distance along the normal is the variable $v$ and the arc length parameter of the projection is $s$.

The new arc length metric is defined as follows: Given any infinitesimal vector in $\Omega_c$ we project it on $c$ and obtain the infinitesimal arc length parameter $ds$ of its projection. The length of the vector is the absolute value of $ds$.

If $\begin{bmatrix} a\ dx_1 \\ b\ dx_2 \end{bmatrix}$ is the infinitesimal vector in in $\Omega_c$ at $x_1, x_2$ then its image in the $s, v$ space is

$$\begin{bmatrix} ds \\ dv \end{bmatrix} = J_{\Phi^{-1}} \left( \begin{bmatrix} a\ dx_1 \\ b\ dx_2 \end{bmatrix} \right),$$

where, $J_{\Phi^{-1}}()$ is the Jacobian of $\Phi^{-1}$. By the new definition, the length of $\begin{bmatrix} a\ dx_1 \\ b\ dx_2 \end{bmatrix}$ is the absolute value of

$$\begin{bmatrix} 1 & 0 \end{bmatrix} J_{\Phi^{-1}} \left( \begin{bmatrix} adx_1 \\ bdx_2 \end{bmatrix} \right).$$

If $\hat{c}(s) = c(s) + \tau f(s)n(s)$ is a curve in $\Omega_c$, then $\Phi^{-1}(\hat{c}) = c(s)$ so that

$$J_{\Phi^{-1}}(\hat{c}'(s)\ ds) = \begin{bmatrix} ds \\ 0 \end{bmatrix},$$

and, the length of $\hat{c}'(s)\ ds$ is

$$\begin{aligned} & | \begin{bmatrix} 1 & 0 \end{bmatrix} J_{\Phi^{-1}} (\hat{c}'(s)ds) | \\ = \ & ds. \end{aligned}$$

Thus, when an infinitesimal piece of the curve $c(s)$ is moved normal to itself (to $c(s) + \tau f(s)n(s)$), its (new-) arc length does not change.

The new (local) external energy of $\hat{c}$ can be written as

$$E = \int_0^L P(|| \bigtriangledown I(c(s) + \tau f(s)n(s))||)ds. \tag{4.1}$$

## 4.2 The New Speed Function

The rate of change of $E$ with respect to $\tau$ at $\tau = 0$ is

$$\frac{dE}{d\tau}_{\tau=0} = \int_0^L \frac{dP(|| \bigtriangledown I||)}{d|| \bigtriangledown I||} \frac{\partial|| \bigtriangledown I||}{\partial n(s)} f(s)\, ds,$$

where, as before, $\frac{\partial||\bigtriangledown I||}{\partial n(s)}$ is the directional derivative of $|| \bigtriangledown I||$ along the direction $n(s)$. As expected, the rate of change of $E$ is a linear function of the tangent vector $f(s)n(s)$. The maximum rate of increase in $E$ is achieved when $f(s)$ is proportional to

$$\frac{dP(|| \bigtriangledown I||)}{d|| \bigtriangledown I||} \frac{\partial|| \bigtriangledown I||}{\partial n(s)}.$$

The maximum rate of decrease occurs when $f(s)$ is the negative of this function. Therefore, we can set the preferred direction of evolution from $c$ to be the vector

$$v(s) = -\frac{dP(|| \bigtriangledown I||)}{d|| \bigtriangledown I||} \frac{\partial|| \bigtriangledown I||}{\partial n(s)} n(s). \tag{4.2}$$

This is the vector field mentioned in the step 1 above.

## 4.3 The New Active Contour is Consistent

An active contour evolving as an integral curve of this vector field stops at a point where $v$ is zero. Thus, the vector field is *consistent* if it is zero when the active contour is over an edge:

**Definition:** An image dependent vector field in $\mathcal{C}$ which is used for an active contour is

*consistent* if its value is zero at $c = c^*$ for $I \in \mathcal{I}_{c^*}$.

**Proposition 3:** The vector field in equation (4.2) is consistent.

**Proof:** If $I \in \mathcal{I}_{c^*}$, and $c = c^*$, then $\frac{\partial||\bigtriangledown I||}{\partial n(s)}$ is zero. Thus the vector field at $c = c^*$ is zero, showing that the vector field is consistent.

We proceed to step 2 discussed above. After adding internal energy, the local energy becomes

$$E = \int_0^L P(||\bigtriangledown I(c(s) + \tau f(s)n(s))||)ds + \alpha\ \hat{L}, \tag{4.3}$$

where, $\hat{L}$ is the Euclidean arc-length of $c^*$. As above, the preferred direction of evolution from $c$ is the vector in the tangent space of $\mathcal{C}$ at $c$ which decreases the energy the most. Repeating the calculation from above, this vector is:

$$v(s) = -\frac{dP(||\bigtriangledown I||)}{d||\bigtriangledown I||}\ \frac{\partial||\bigtriangledown I||}{\partial n(s)}\ n(s) + \alpha\ \kappa(s),$$

which, in terms of an arbitrary regular parameterization of $c$ is

$$v(q) = -\frac{dP(||\bigtriangledown I||)}{d||\bigtriangledown I||}\ \frac{\partial||\bigtriangledown I||}{\partial n(q)}\ n(q) + \alpha\ \kappa(q).$$

Curve evolution under this vector field is given by

$$\frac{\partial c(q,t)}{\partial t} = v(q). \tag{4.4}$$

We will evaluate this form of the active contour in the experiment chapter.

# Chapter 5

# Choosing the Optimal $\alpha$

In the chapter, we present a principle and algorithm to choose the appropriate $\alpha$ in the curve evolution equation (equation 4.4).

## 5.1 Principle

In the section, we study the structure of an image near an edge and establish a principle to choose the appropriate value for the smoothness parameter.

### 5.1.1 Image Properties in a Curve's Neighborhood

An edge is usually associated with a discontinuity in image intensity. "If we take a cross section of the image brightness along a line at right angles to an edge, we might hope to see a step discontinuity. In practice, the transition will not be abrupt because of blurring and limitations of the imaging device." [38] Usually we will see a *ramp*, and we demonstrate the fact by one MRI image of a dog heart in figure 5.1.

Figure 5.1.(a) contains the image of a dog heart, in which we can see a clear contour surrounding the white region in the middle. We draw a line segment perpendicular

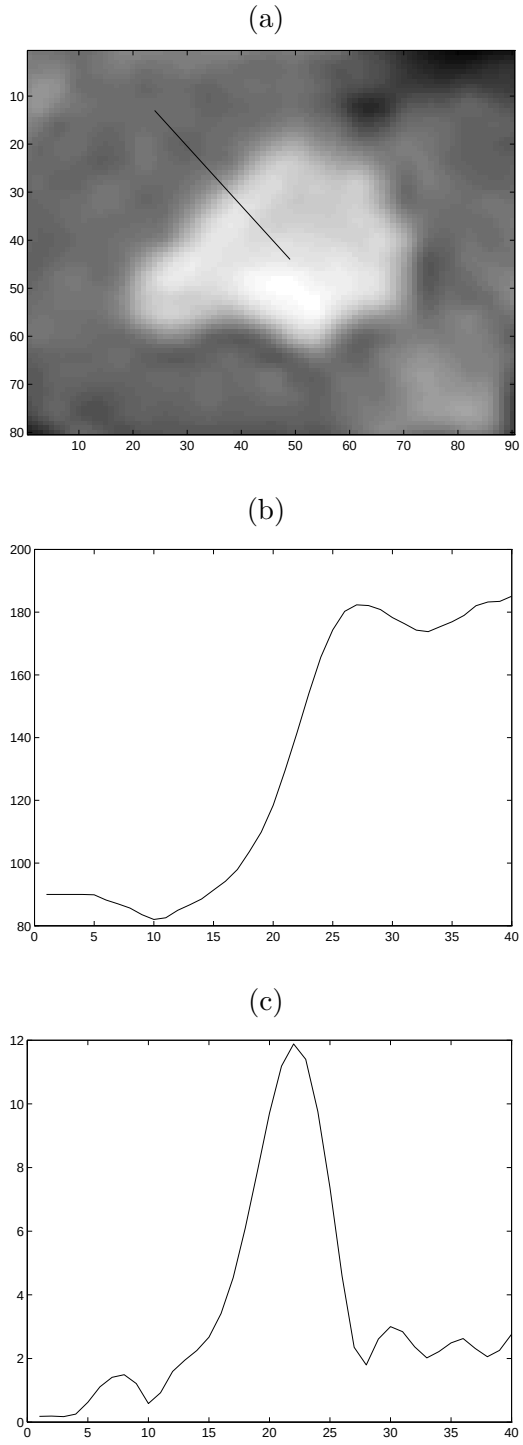



Figure 5.1: Image structure

to the contour, as shown in the figure. Then, we sample the image intensity at 45 points on the line segment and plot the sample in figure 5.1.(b). The sample has the shape of a ramp.

Further, the gradient magnitude $u = ||\bigtriangledown I||$ along the line segment roughly forms a Gaussian function, as shown in figure 5.1.(c). We explain this property by analyzing two edge models – *step model* and *smoothed-step model*.

In figure 5.2.(a), the wide curve in the center separates two regions of contrast brightness. A straight line ($v$-axis) lays along the normal direction of the boundary. We denote the image intensity along the $v$-axis as $I(v)$, the gradient magnitude along the line as $u(v)$, and the step function as $S(v)$:

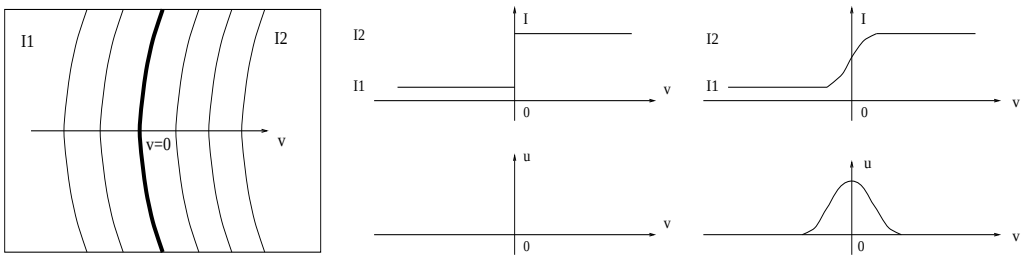
$$S(v) = \begin{cases} 0, & \text{for} v < 0; \\ 1/2, & \text{for} v = 0; \\ 1, & \text{for} v > 0. \end{cases}$$

In the two simple models, we assume that image intensity $I$ is constant along each curve parallel to the wide one in figure 5.2.(a). Under this assumption, the gradient of $I$ perpendicular to $v$-axis is 0 on $v$-axis, so $u(v) = ||\bigtriangledown I||(v) = |\frac{dI}{dv}(v)|$.

In the step model, $I(v) = I_1 + (I_2 - I_1)S(v)$, and therefore $u(v) = \frac{dI}{dv} = (I_2 - I_1)\delta(v)$, where $\delta(v)$ is the unit impulse function. Figure 5.2.(b) illustrates $I(v)$ and $u(v)$. The location of the peak of $u$ is the location of the edge. That is $v = 0$.

In the smoothed-step model, we model the image intensity along the line as a step function smoothed by a Gaussian function $G$:

$$I(v) = I_1 + (I_2 - I_1)(S \otimes G)(v)\,, \tag{5.1}$$




(a) (b) (c)

Figure 5.2: Edge models

and thus,

$$u(v) = \frac{dI}{dv} = (I_2 - I_1)G(v)\,. \tag{5.2}$$

Figure 5.2.(c) illustrates $I(v)$ and $u(v)$. In this model, the edge is of a certain scale. We define the position of the edge to be $v = 0$. As we can see in the figure, $u$ has two properties near the edge according to the model.

- **Property 1** $u$ along the normal direction is maximal on the edge.
- **Property 2** $u$ along the normal direction forms a Gaussian function centered at the edge. We can model the edge in different ways. Generally, $u$ along the normal direction is "Gaussian like".

Because of the property 2, $u$ along the edge forms a ridge, as shown in figure 5.3.

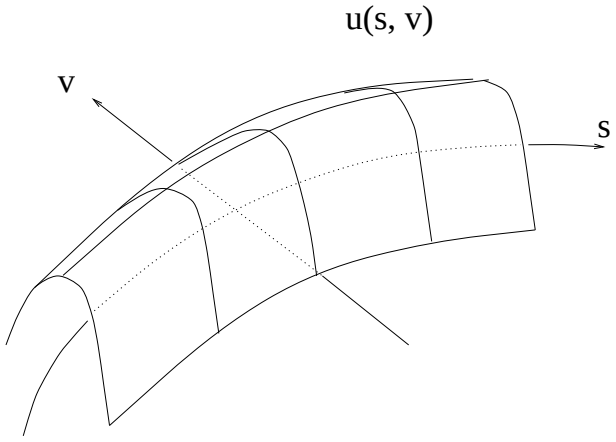



Figure 5.3: $u$ forms a ridge

### 5.1.2 How the External Energies Work

When researchers design external energies for an active contour to seek edges, they all use the first property of $u$. As we discussed in Chapter 2, while minimizing the external energy, an active contour evolves to minimize a potential function $P$. In edge-seeking tasks, researchers have proposed the following three formulations of $P$:

$$P = -u\,,$$

$$P = \frac{1}{1+u^n}\,,$$

and

$$P = exp(-u)\,.$$

In all of the three, $P(u)$ is a monotonically decreasing function of u and the minimum of $P$ happens at the maximum of $u$, so the active contours evolve to maximize $u$. With

this kind of evolution, an active contour will move closer to an edge because maxima of $u$ happen at the edges.

### 5.1.3 The Functions of Internal Energy Terms

External energy leads active contours to maximize gradient magnitude $u$. In noiseless images, since $u$ is locally maximal at edges, the external energy alone will lead the curve to edges. However, there is always noise in image, so the maxima of image gradient no longer necessarily happen at edges. Thus, if we evolve the active contour by external energy alone, it will end up with a very jagged shape.

Furthermore, if we evolve an active contour by external energy alone, the speed function becomes

$$\frac{\partial c}{\partial t} = -(\bigtriangledown P \cdot n)n \ .$$

Nothing prevents the curve to develop singularities; in other words, the curve may not stay smooth, and thus $n$ may not be well defined.

To solve the first problem, researchers added an internal energy term to smooth (regularize) the curve. To minimize the internal energy, an active contour lowers the curvature at all parts of itself, and therefore smoothes itself. With the added term, the energy functional of an active contour embeds two goals of the curve – it evolves to maximize $u$ and to smooth itself. Setting the parameter $\alpha$ to different values, the user will smooth the curve at different levels, and he/she can choose an appropriate value to achieve the best smoothing effect.

The internal term also solves the second problem: with the new term, the speed

function is

$$\frac{\partial c}{\partial t} = -(\bigtriangledown P \cdot n - \alpha\kappa)n \ ,$$

when $\kappa > |\bigtriangledown P \cdot n|/\alpha$, the point evolves to lower the curvature. Since $|\bigtriangledown P \cdot n|$ has a upper bound in an image, the curvature of an active contour can not increase forever, and thus it will not develop singularities.

In theory, the user can never set $\alpha = 0$ in equation 4.4, otherwise the curve may not have well-defined normals. When he/she does not want the smoothing effect, he/she can only set $\alpha$ to a very small value, so that the internal term prevents singularities but not smooth the curve much. In the rest of the thesis, we will denote the very small value by $\alpha_0$:

$$\alpha_0 \equiv \text{a small}\, \alpha \tag{5.3}$$

and

$$c_0 \equiv \text{the equilibrium position of equation 4.4 with}\, \alpha = \alpha_0 \ . \tag{5.4}$$

This is not a rigid definition for $\alpha_0$, while we can define $\alpha_0$ rigidly by some heuristic rules in practice. For example, we can determine it adaptively: we can decrease $\alpha$ by a factor of 10 every time, until the maximal internal force is less than the average external force. In our experiments, when we set $\alpha$ to a value below 0.1, the results look no different to the result with $\alpha = 0.1$. So we can set $\alpha_0 = 0.1$.

### 5.1.4 Optimal-$\alpha$ Principle

Setting $\alpha$ to the right value in active contour algorithms is critical, and we illustrate the consequences of setting $\alpha$ to inappropriate values in figure 5.4.

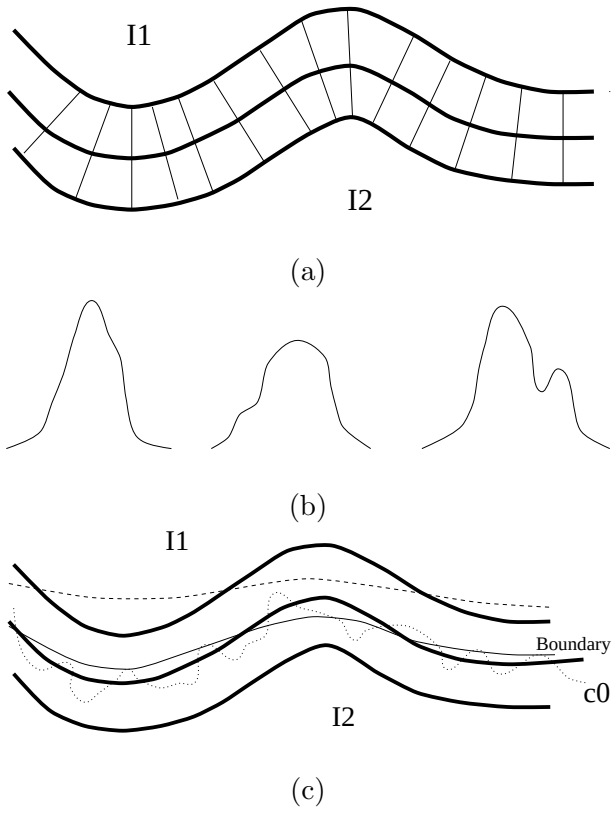



Figure 5.4: Image property along normals.

The center wide curve in figure 5.4.(a) stands for the boundary of two regions with contrast brightness, and the other two wide curves show where the edge vanishes – image intensity is basically a constant $I_1$ above the upper curve and is basically a constant $I_2$

below the lower curve. We draw normals of the boundary as grids in the figure. Figure 5.4.(b) shows several samples of $u$ along the normals at different positions. From figure 5.4.(a) and 5.4.(b), we can see that $u$ forms a ridge along the true boundary. Figure 5.3 illustrates this more clearly for a noiseless image.

While evolving an active contour according to equation 4.4, if we set $\alpha$ to $\alpha_0$, the resulting curve $c_0$ will be more jagged than the true boundary because of noise, and we draw the possible result as the dotted curve in figure 5.4.(c). In practice, the user will increase $\alpha$; as a consequence, the curve gradually gets smoother and smoother, while most points on the curve are getting closer and closer to the center of the ridge. We draw one possible smooth result as the thin curve in figure 5.4.(c). If the user continues to increase $\alpha$, the curve will get even smoother, and some points on the curve will move farther away to the center; eventually, the whole curve moves out of the ridge and collapse into a single point. We draw one possible over-smoothed result as the dashed curve in figure 5.4.(c). To find the right answer, the user will choose a number of $\alpha$'s, find their corresponding equilibrium positions, and find the one that achieves the best smoothing effect compared with curve $c_0$.

From the above analysis, we can see that the smoothing effect is closely related to the distance between the curve and the center of the ridge, we will use the property to choose $\alpha$, and our principle follows.

**Optimal-$\alpha$ Principle:** we choose an $\alpha$ which results an answer that is the closest to the center of the ridge.

Of course we do not know the center when we run the active contour algorithm, so we need a functional to compare curves as how well they are centered on a ridge, without knowing the center.

### 5.1.5 Overview of the Rest of the Chapter

In the next section, we will study the structure of the equilibrium position set of active contours. The structure determines that we need a special algorithm, which we will present in the section after. The algorithm specifies the requirements of a comparison functional, which we define in two steps. First in section 5.4, we will define a measure function $\tilde{P}$ on how well one point on a curve fits to the center of an ridge, and we discuss the properties of it. Then in section 5.5, we will define an comparison functional $E_2$ to compare curves as a whole. In the last section, with all the pieces we will have developed, we will present the algorithm in detail.

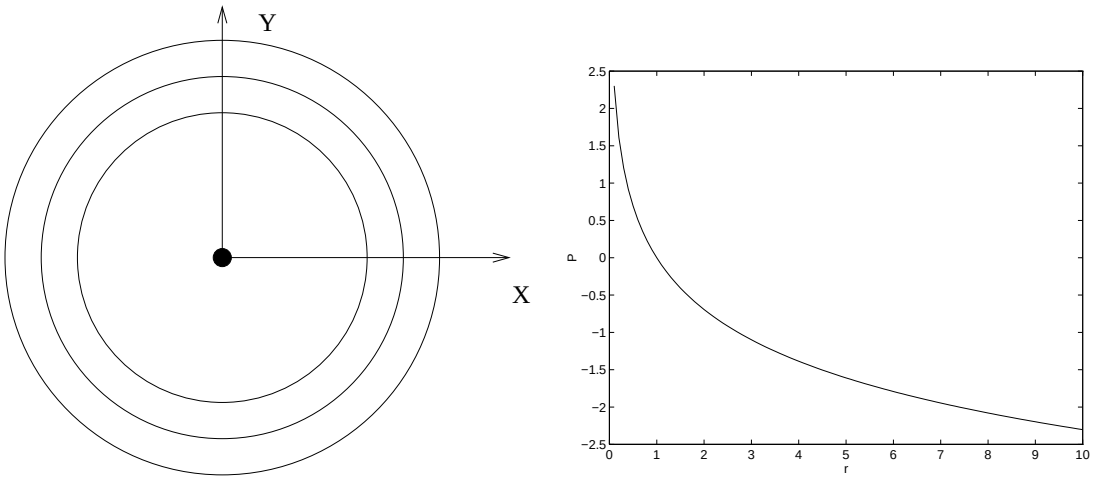
## 5.2 The Structure of the Result Set

To choose an appropriate $\alpha$, we need to know the structure of the set of all equilibrium positions of equation 4.4; for instance, if the equilibrium positions form a continuous function of $\alpha$, we might be able to find the right $\alpha$ by gradient descent method. That is the motivation for us to study the structure of the result set before constructing an algorithm to choose $\alpha$.

### 5.2.1 On Circles

Again we conduct our study on circularly symmetric images for computational convenience. We will discuss the case of a generic curve after this.

Figure 5.5.(a) contains a group of concentric circles, whose normals point inward. We assume that $P$ is constant on each of the circles: $P(x, y) = P(\sqrt{x^2 + y^2}, 0)$. Thus, if we initialize the curve on one of the circles, every point on the curve should behave in the same way – each point moves along the radius with the same speed. As a consequence,




(a) Concentric circles (b) $P$ against radius

Figure 5.5: Circles

the curve will always remain a circle. In this setup, the internal force always points inwards, while we design the external force to always point outwards. We found some special images in which active contours behave in very interesting ways:

- **In some images, there are more than one connected equilibrium positions for one $\alpha$; in other words, equilibrium positions are not a function of $\alpha$ in some images.**

  Figure 5.5.(b) shows the sample of a special $P$ ( $P(r,0) = -\ln r$ ) along $x$ axis. We set the curve at any of the circles, whose radius is $r$, and we set $\alpha = 1$; then, $\bigtriangledown P \cdot \vec{N} = \frac{1}{r}$, $\kappa = \frac{1}{r}$, and

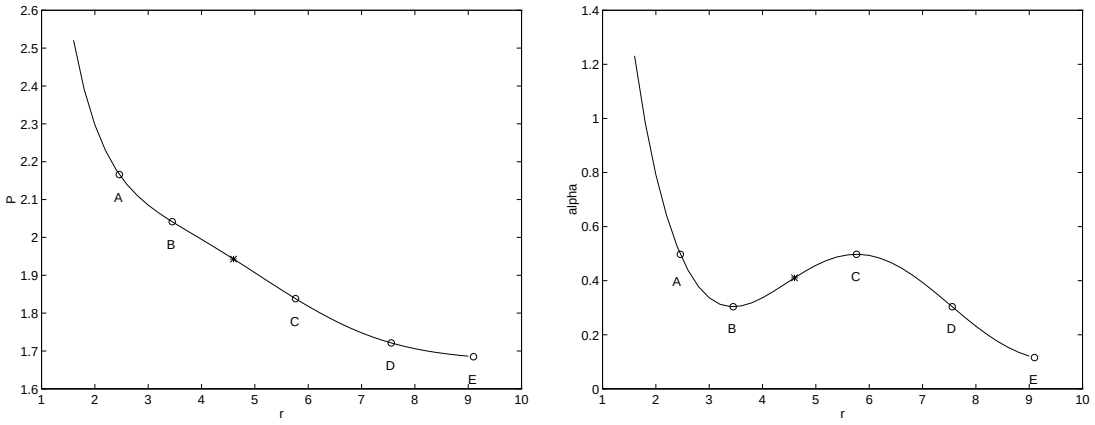
  $$\frac{\partial \vec{c}}{\partial t} = -\lambda(\bigtriangledown P \cdot \vec{N} - \alpha\kappa)\vec{N} = 0\,,$$

  which means the curve is at an equilibrium position. Since this is true for any $r$,

any of the infinite number of connected circles is an equilibrium position for $\alpha = 1$.

In the framework of arc-length active contours, the curve evolves according to equation 3.15 instead of equation 4.4. We can find a similar example: if $P(r,0) = \frac{1}{r} - 1$, any of the circles will be an equilibrium position for arc-length active contours with $\alpha = 1$.

- **In some images, the equilibrium position is a discontinuous function of $\alpha$.**




(a) $P$ against radius (b) $\alpha$ against radius

Figure 5.6: A special $P$

Figure 5.6.(a) shows the sample of a special $P$:

$$P(r,0) = \frac{2 + \cos r + r \int_{0.1}^{r} dz \frac{\sin z}{z}}{r} .$$

We use 0.1 instead of 0 as the lower bound of the integral in the above equation

because it is easier to plot $P$ numerically with this lower bound. From it we derive $\bigtriangledown P \cdot \vec{N} = -\frac{dP(r,0)}{dr} = \frac{\cos r+2}{r^2}$; and $\kappa(r) = \frac{1}{r}$. Plugging them into the following equation

$$\bigtriangledown P \cdot \vec{N} - \alpha\kappa = 0$$

we get

$$\alpha = \frac{\cos r + 2}{r}\,. \tag{5.5}$$

This is the $\alpha$ that makes the circle with radius $r$ an equilibrium position. We plot $\alpha$ against $r$ in figure 5.6.(b). Each point on the curve represents a pair of $\alpha$ and $r$. We mark several points on the curves in figure 5.6, which we will refer to later; we denote values of their $r$'s as $r_A, r_B, \cdots$, and denote values of their $\alpha$'s as $\alpha_A, \alpha_B, \cdots$.

When we can set the curve at the circle of radius $r_C$ with $\alpha = \alpha_C$, the curve will be at equilibrium. After that, if we increase $\alpha$ a little, increased internal force will shrink the curve; furthermore, there is no equilibrium position for an $\alpha > \alpha_C$ in the near neighborhood, so the curve will keep on shrinking until it gets to a circle whose radius is smaller than $r_A$.

Similarly, if we set $\alpha = \alpha_B$, the curve will be at equilibrium at the circle of radius $r_B$. When we decrease $\alpha$ a little, the curve will evolve to a circle whose radius is bigger than $r_D$.

As a consequence, if we treat the new active contour algorithm as a black box which takes one image, one $\alpha$ and one initial position as input and gives back a final position, and we initialize the curve as a circle whose $r \notin (r_B, r_C)$, the radius of the final curve can never be in $(r_B, r_C)$. Later, we will show even if we initialize the curve at a circle with $r \in (r_B, r_C)$, it will get out of that region in practice.

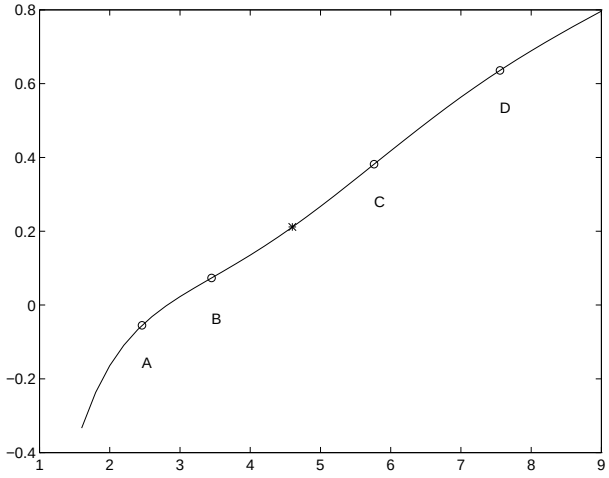



Figure 5.7: The example for arc-length active contour; $P$ against radius

The similar example for the framework of arc-length active contours is:

$$P(r,0) = 0.1r + \frac{r\sin r + r^2 \int_r^\infty \frac{\cos z}{z} dz - \cos r - 2}{2r}\,.$$

With this $P$, if we solve

$$\bigtriangledown P - P\kappa - \alpha\kappa = 0$$

we will get the same $\alpha(r)$ as in figure 5.6.(b). The same situation exists. We show shape of this $P$ in figure 5.7.

- **There are unstable equilibrium positions.**

  We use the example in figure 5.6 to demonstrate this phenomenon. For any $r \in (r_B, r_C)$, we can make the curve at equilibrium there with an $\alpha$ computed according to equation 5.5; however, the equilibrium position is unstable: a small perturbation

will make the curve move far away. For example, suppose the curve stops at $r = 4.6$ with $\alpha = \alpha_*$. We mark the $(r, \alpha)$ pair by "*"s in figure 5.6.

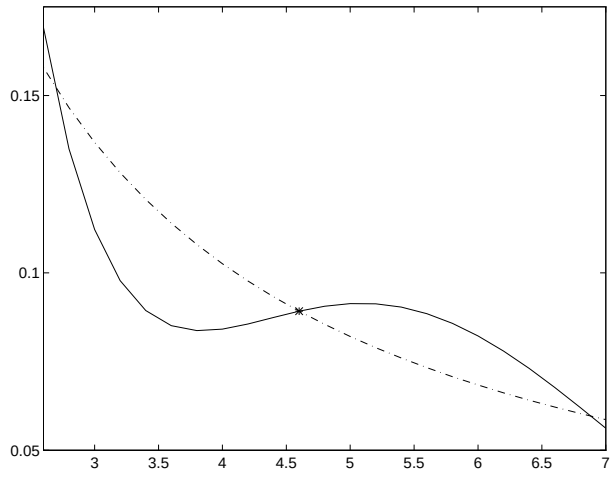



Figure 5.8: Forces

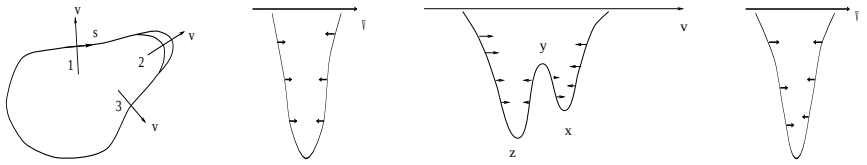
We compute the forces on the curve as if the curve is at the circle with different radius $r$ and the same $\alpha = \alpha_*$. Then in figure 5.8, we plot the absolute value of external force as a solid curve and the absolute value of internal force as a dashed curve. External force always drives the curve outward, while internal force always drives the curve inward. The absolute value of internal force is always bigger than the absolute value of external force in the region between $r_B$ and $r_*$. The absolute value of external force is always bigger than the absolute value of internal force in the region between $r_*$ and $r_C$. As a result, if the curve shrinks a little from the current position ($r = r_*$), internal force dominates, and the curve keeps on shrinking

until it moves out of the region between $r_B$ and $r_C$; if the curve expands a little from the current position ($r = r_*$), external force dominates, and the curve keeps on expanding until it moves out of the region between $r_B$ and $r_C$. In either case, the curve evolves far away from $r = r_*$ with a small perturbation. The same analysis applies to the whole region between $r_B$ and $r_C$, and there is always perturbation due to noise, so the final position of the curve can never be there. This is very surprising, since $P$ in region $r_B r_C$ looks similar to $P$ in region $r_A r_B$ and region $r_C r_D$ in figure 5.6.(a). In arc-length active contour models, the same thing happens.

### 5.2.2 On Generic Curves

So far, we have used circles for computational convenience. Actually, the discontinuity of equilibrium positions is very general, and we will show this on a generic object in figure 5.9.(a). On the upper-right corner of the object, there are actually two close-by edges, which eventually converge. One of the edges is due to noise. This false-edge phenomenon happens a lot at different scales in real images. The samples of $P$ along three normals are shown in figure 5.9(b), 5.9(c) and 5.9.(d) respectively, where the little arrows show the directions of external forces. As we can see in the figure, the sample at point 2 is very different from the other two since it is at the double edge.

Suppose that an active contour gets to an equilibrium position at the outer boundary with a very small $\alpha$. The corresponding position in figure 5.9.(c) is point $x$, and the corresponding positions in figure 5.9.(b) and 5.9.(d) are the bottom of the valley. Let us also assume that the valley at the double boundary is much more shallow than the rest of parts of the curves, so that when we increase $\alpha$, the part of the curve at the double boundary moves much more than the part at the single boundary, in which case we can concentrate on behaviors of the active contour at the double boundary part.




(a) A curve (b) Sample 1 (c) Sample 2 (d) Sample 3

Figure 5.9: A special $P$

When we gradually increase $\alpha$, the equilibrium curve at position 2 will move gradually from $x$ towards $y$. $y$ is in the between of the two boundaries. However, once the curve passes $y$, it can not stop in the region between $y$ and $z$, since in that region, both internal force and external force point inwards the curve, and the net force can not be 0. The equilibrium position jumps from $y$ to a point left to $z$ for a small change of $\alpha$.

Again, the example supports our claim that the equilibrium position is not a continuous function of $\alpha$.

### 5.2.3 Conclusion: Discontinuity of Equilibrium Positions

In the section, we studied the structure of the set of equilibrium positions of active contours. Of all the interesting behaviors we found, one is very important for us to bear when we design the algorithm:

**Discontinuity of Equilibrium Curves:** When the active contour is at an equilibrium position, and the user increases $\alpha$ a little, the next equilibrium position may be far away from the current one.

## 5.3 A Discrete Algorithm

Since the equilibrium position is not continuous with respect to $\alpha$, we can not apply gradient descent method to choose $\alpha$. The reason is: even if we find a functional of equilibrium positions that is continuous with respect to curves, it won't be continuous with respect to $\alpha$, and a gradient descent methods requires the function/al to be continuous with respect to the variable being estimated. Considering this fact and the fact that we do not know the true center of the ridge of $u$, we propose the following discrete algorithm:

1. Given an image and an initial position, we set $\alpha = \alpha_0$ (a very small value) and evolve the active contour from the initial position to a certain equilibrium position $c_0$.

2. For each $k = 1, 2, 3, \cdots$, we set $\alpha = 1.5^k \alpha_0$, and evolve the curve from $c_0$ to a new equilibrium position $c_k$, until $\alpha$ is too large: when we increase $\alpha$, the final position will be away from curve $c_0$ and even collapse into one point. If for an $\alpha$, the curve evolves away from $c_0$ with an average or maximal evolution distance bigger than a certain value, we consider the $\alpha_k$ as too large.

3. Choose the best curve from all the final positions in terms of how well they are centered on the ridge compared to $c_0$. We will explain how we achieve this in the next two sections.

After we define the comparison functional to compare how well curves are centered on a ridge, we will re-state the algorithm in details in section 5.6.

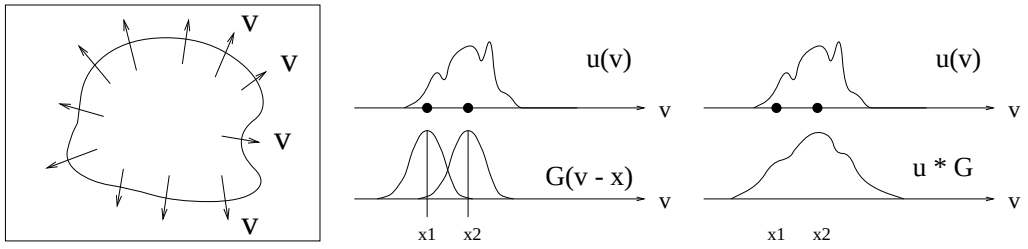
The obvious drawback of this algorithm is that it only tests a number of $\alpha$'s and therefore it may miss some good $\alpha$'s and their corresponding results. It is not a big problem in practice. Since we care more about the positions of the curve than the $\alpha$'s,

we can simply decrease the intervals of $\alpha$'s if we find the distance between curves is too large.

## 5.4 How to Measure the Improvement of One Point

### 5.4.1 Formulation

In this sub-section, we present a function $\tilde{P}$ to measure how well a point fits to the center of a ridge.




(a) A curve (b) $u(v)$ and $G(v-x)$ (c) $u(v)$ and $u*G$

Figure 5.10: To measure one point

Figure 5.10.(a) shows a curve and normals of it, figure 5.10.(b) shows one sample of $u(v)$ along a normal. We know that $u(v)$ is a Gaussian function plus noise. Give two position $x_1$ and $x_2$, we want to compare which one is closer to the center. We can try the following method: for each $x$, we construct a Gaussian function $G(v-x)$ which is centered at $x$. All the Gaussian functions are identical after transformation. We use the integral of the product ([18] Page 310) of $u$ and $G$. of $u$ and $G(v-x)$ as the criterion on

how similar $G(v-x)$ is to $u(v)$:

$$\mathcal{R}_{uG}(x) = \int u(v)G(v-x)dv\,.$$

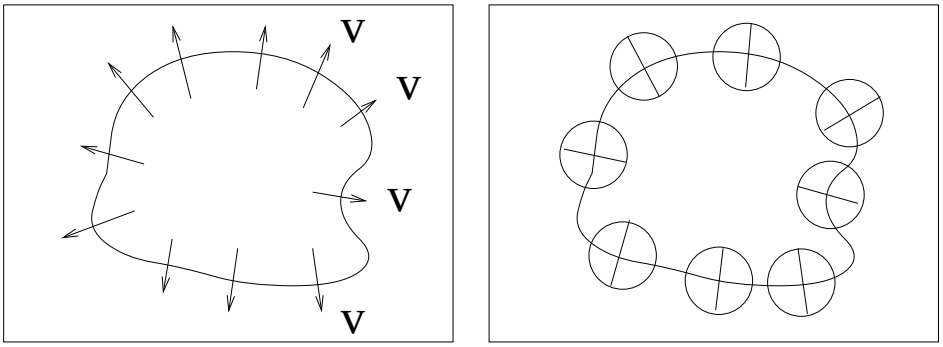
$G(v-x_1)$ is more similar to $u(v)$ than $G(v-x_2)$ is, when $\mathcal{R}_{uG}(x_1) > \mathcal{R}_{uG}(x_2)$. Because $G(v-x)$ is symmetric at $v=x$, we get

$$\mathcal{R}_{uG}(x) = \int u(v)G(x-v)dv = u * G$$

So, $G(v-x_1)$ is more similar than $G(v-x_2)$ when $(u*G)(x_1) > (u*G)(x_2)$.

**NOTE:** when the image is noiseless, this is always true no matter what size of $G$ we chose. With this measure, the principle becomes: choose the curve that maximizes $u*G$.

If we proceed with this measure, we will face a lot of difficulties in computation and implementation, since the computation requires knowing the normal of the boundary. To avoid the problem, we develop a measure function independent of normals. Instead




(a) Measure along normals (b) Circular measure

Figure 5.11: Two ways to measure one point

of measuring $u$ along the normals, as shown in figure 5.11.(a), we sample $u$ in a circle – as shown in figure 5.11.(b) – and define the measure function to be the convolution of the sample with a two-dimensional Gaussian function $G_2$, Mathematically we define a measure function $\tilde{P}$ at position $(x, y)$ as

$$\tilde{P}(x,y) = \int_{-\infty}^{\infty} d\bar{x} \int_{-\infty}^{\infty} d\bar{y}\ u(x-\bar{x}, y-\bar{y}) G_2(\bar{x},\bar{y})\,. \tag{5.6}$$

We denote the STD of $G_2$ by $\sigma_2$. Since Gaussian function almost vanishes beyond the point that is $3\sigma_2$ away from the center, in practice we use:

$$\tilde{P}(x,y) = \int d\bar{x} \int d\bar{y}_{\sqrt{\bar{x}^2+\bar{y}^2} \le 3\sigma_2} u(x-\bar{x}, y-\bar{y}) G_2(\bar{x},\bar{y})$$

Given a point $c_0(q)$ on curve $c_0$ and a point $c_k(q)$ on curve $c_k$, we say $c_k(q)$ is better than $c_0(q)$ when $\tilde{P}(c_k(q)) > \tilde{P}(c_0(q))$.

Now, the question is: how good is the measure function?

### 5.4.2 On Lines and Curves

**Straight Lines in Noiseless Images**

When the boundary is a straight line and $u$ along any line parallel to the boundary is constant, [1], $\tilde{P}$ is identical to $u * G$. We demonstrate this in the figure 5.12, which shows the shape of $u$ near a straight edge.

In the figure, the $s$-axis is along the edge, and the $v$-axis is perpendicular to the edge. We use $G$ to denote a one-dimensional Gaussian function whose STD is $\sigma_2$ and $G_2$

[1] The two assumptions are roughly satisfied for many parts of an object in real images.

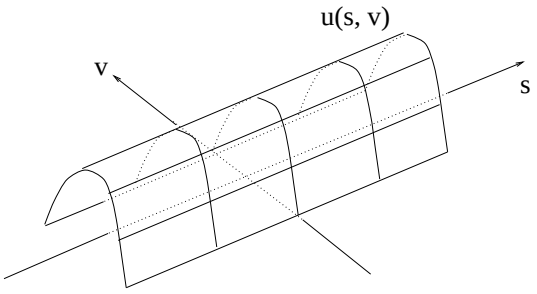



Figure 5.12: Straight edge model

to denote a two-dimensional Gaussian function whose STD is also $\sigma_2$, then

$$\begin{aligned} G_2(s,v) &= \frac{1}{2\pi\sigma_2^2} e^{-\frac{s^2+v^2}{\sigma_2^2}} \\ &= \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{s^2}{\sigma_2^2}} \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{v^2}{\sigma_2^2}} \\ &= G(s)G(v) \end{aligned}$$

We use $u(s,v)$ to denote the gradient magnitude at point $(s,v)$. Under our assumptions, $u(s_1,v) = u(s_2,v)$ for any $s_1$, $s_2$, and $v$. In this coordinate system,

$$\tilde{P}(s,v) = \int d\bar{s} \int d\bar{v}\ u(s-\bar{s}, v-\bar{v}) G_2(\bar{s},\bar{v}) \tag{5.7}$$

We will prove

$$\tilde{P}(s,v) = \mathcal{R}_{u(s,.)G}(v) \tag{5.8}$$

which means $\tilde{P}$ is consistent with the one dimensional measure when we are dealing with

a constant straight edge.

$$
\begin{aligned}
\tilde{P}(s,v) &= \int d\bar{s} \int d\bar{v}\ u(s-\bar{s}, v-\bar{v}) G_2(\bar{s},\bar{v}) \\
&= \int d\bar{s} \int d\bar{v}\ u(s-\bar{s}, v-\bar{v}) G(\bar{s}) G(\bar{v}) \\
&= \int d\bar{v} G(\bar{v}) \int d\bar{s} u(s-\bar{s}, v-\bar{v}) G(\bar{s}) \\
&= \int d\bar{v}\, G(\bar{v}) \int d\bar{s}\, u(s, v-\bar{v}) G(\bar{s}) \\
&= \int d\bar{v}\, G(\bar{v}) u(s, v-\bar{v}) \\
&= \int d\bar{v}\, u(s,\bar{v}) G(v-\bar{v}) \\
&= \int d\bar{v}\, u(s,\bar{v}) G(\bar{v}-v) \\
&= \mathcal{R}_{u(s,.)G}(v)\,.
\end{aligned}
$$

From the above analysis, we know that the measure function is precise on a straight line with constant image intensity along any line parallel to the center: in this case, when $\tilde{P}(c_k(q)) > \tilde{P}(c_0(q))$, $c_k(q)$ is closer to the center than $c_0(q)$ is.

**Curves**

The measure function $\tilde{P}$ is biased when the boundary is not straight, by which we mean the positions of maxima of $u$ are different from the positions of maxima of $\tilde{P}$ in the same noiseless image.

This is obvious: $\tilde{P}$ is the convolution of potential function $P$ and Gaussian function $G_2$, and researchers have proved that the convolution of image with a Gaussian function moves minima/maxima [1], so the maxima of $\tilde{P}$ must be different from the maxima of $u$. We demonstrate the effects in next sub-section.

### 5.4.3 Bias on a Circle

We will illustrate that $\tilde{P}$ is biased, but the bias is small in our case. The wide solid curve in figure 5.13 stands for the maxima of $u$, and the dotted curve stands for the maxima of $\tilde{P}$. Since we can always fit a circle to one point on the curve, and make the circle has the same curvature of the curve at the tangential point, we study the bias on a circular object, which makes the formulation simpler.

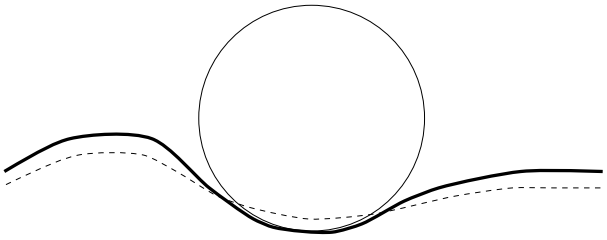


Figure 5.13: Bias introduced by approximation

Figure 5.14 contains a white disc in gray background. To simplify computation even further, we assume the image in the figure is circularly symmetric. In other words, we assume that image intensity is the same at points with the same distance to the center of the disc. We set the origin of the $xy$ coordinate at the center of the disc. Figure 5.14 also contains samples of $I$, $u$, and $\tilde{P}$ along $x$ axis. The maxima of $u$ form a circle with radius equal to $R$, and the circle is the true boundary of the disc. The maxima of $\tilde{P}$ form another circle with radius equal to $bR$. Later, we will prove that $b < 1$, in other words, the maxima of $\tilde{P}$ are inside of the maxima of $u$. As a result, the optimal position defined by $\tilde{P}$ is different from the true boundary even in the noiseless image. The difference is

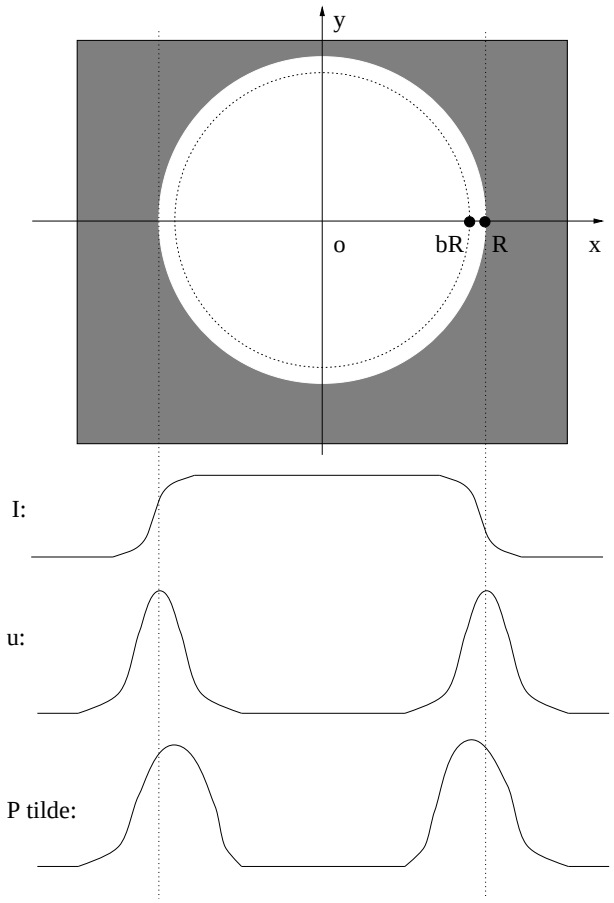



Figure 5.14: A circle

the bias, and we will use $1 - b$ , which is the ratio of the bias over radius $R$, to quantify the scale of the bias.

Since $u$ is circularly symmetric, value of $u$ at a point depends on the distance from this point to the origin only: at $(r\cos\theta, r\sin\theta)$ in the image, the value of $u$ is

$$u(r\cos\theta, r\sin\theta) = u(r, 0)$$

In our model, $u$ along normal direction is an Gaussian function, so

$$u(r,0) = \frac{M}{\sqrt{2\pi}\sigma_1} e^{-\frac{(r-R)^2}{2\sigma_1^2}}$$

We compute $\tilde{P}$ as:

$$\begin{aligned}
\tilde{P}(x,y) &= \int_{-\infty}^{\infty} d\bar{x} \int_{-\infty}^{\infty} d\bar{v}\, u(x-\bar{x}, y-\bar{y}) G_2(\bar{x},\bar{y}) \\
&= \int_{-\infty}^{\infty} d\hat{x} \int_{-\infty}^{\infty} d\hat{y}\, u(\hat{x},\hat{y}) G_2(x-\hat{x}, y-\hat{y}) \\
&= \int_0^{\infty} dr \int_0^{2\pi} d\theta\; ru(r\cos\theta, r\sin\theta) G_2(x-r\cos\theta, y-r\sin\theta) \\
&= \int_0^{\infty} dr \int_0^{2\pi} d\theta\; ru(r,0) G_2(x-r\cos\theta, y-r\sin\theta) \\
&= \int_0^{\infty} dr \int_0^{2\pi} d\theta\; ru(r,0) \frac{1}{2\pi\sigma_2^2} \exp(-\frac{(x-r\cos\theta)^2 + (y-r\sin\theta)^2}{2\sigma_2^2})
\end{aligned}$$

We make several variable substitutions:

$$a = \frac{r}{R}, b = \frac{\sqrt{x^2+y^2}}{R}, \beta = \arctan\frac{y}{x}, \text{and } k_2 = \frac{\sigma_2}{R}$$

. Then,

$$(x,y) = (bR\cos\beta, bR\sin\beta)$$

, Of the new variables, $a$ and $\beta$ are intermediate variables, which we will not use after the computation; $b$ means the ratio of the radius of the circle of $\tilde{P}$ over $R$ (the radius of the circle of $u$), $1-b$ means the ratio of bias over the radius $R$, $k_2$ means the ratio of $\sigma_2$

over $R$, and $\frac{1-b}{k_2}$ means the ratio of the bias over $\sigma_2$. We get:

$$\begin{aligned}
\tilde{P}(x,y) &= \tilde{P}(bR\cos\beta, bR\sin\beta) \\
&= \int_0^\infty da\ R \int_0^{2\pi} d\theta\ a\ R\ u(aR,0)\frac{1}{2\pi k_2^2 R^2} \\
&\quad \exp\left(-\frac{(b\cos\beta - a\cos\theta)^2 + (b\sin\beta - a\cos\theta)^2}{2k_2^2}\right) \\
&= \frac{1}{k_2^2}\int_0^\infty da\ a\ u(aR,0)\exp\left(-\frac{a^2+b^2}{2k_2^2}\right)\int_0^{2\pi} d\theta\ \frac{1}{2\pi}\exp\left(\frac{ab\cos(\theta+\beta)}{k_2^2}\right) \\
&= \frac{1}{k_2^2}\int_0^\infty da\ a\ u(aR,0)\exp\left(-\frac{a^2+b^2}{2k_2^2}\right)I_0\left(\frac{ab}{k_2^2}\right),
\end{aligned}$$

and

$$\frac{d\tilde{P}(x,y)}{db} = \frac{1}{k_2^4}\int_0^\infty da\ a\ u(aR,0)\exp\left(-\frac{a^2+b^2}{2k_2^2}\right)\left(aI_1\left(\frac{ab}{k_2^2}\right) - bI_0\left(\frac{ab}{k_2^2}\right)\right). \qquad (5.9)$$

$I_0$ and $I_1$ are Bessel functions. When $\frac{d\tilde{P}(x,y)}{db} = 0$, $\tilde{P}$ at $(x,y) = (bR\cos\theta, bR\sin\theta)$ is maximal along radius; furthermore, since $\tilde{P}$ is circularly symmetric, it is a maximal in the image.

Given different models of $u$, we can substitute $u(r,0)$ with different functions and set equation 5.9 to 0 to compute $b$. Then, we get $1-b$, which quantifies the bias.

#### 5.4.3.1 Sharp Edge

If the disc has a sharp edge – $u(aR,0) = \delta(aR-R) = \delta(a-1)$, we get

$$\tilde{P}(x,y) = \frac{1}{k_2^2}\exp\left(-\frac{1+b^2}{2k_2^2}\right)I_0\left(\frac{b}{k_2^2}\right),$$

and

$$\frac{d\tilde{P}(x,y)}{db} = \frac{1}{k_2^4}\exp(-\frac{1+b^2}{2k_2^2})(I_1(\frac{b}{k_2^2}) - bI_0(\frac{b}{k_2^2})) = 0\,.$$

Solving $\frac{d\tilde{P}}{db} = 0$ numerically in Mathematica, we get a number of $b$'s for corresponding $k_2$'s; then, we plot $1-b$ against $k_2$ in figure 5.15.

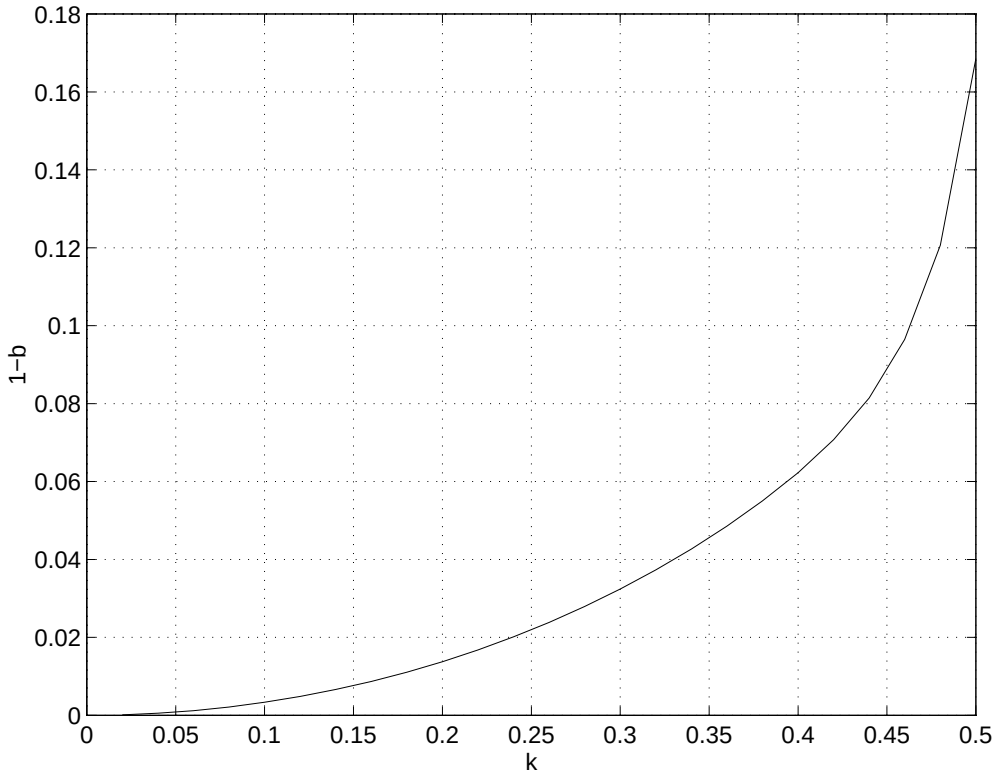



Figure 5.15: $(1-b)$ vs. $k_2$

We will explain its meaning in the next subsection.

#### 5.4.3.2 Smooth Edge

If $u(aR, 0)$ is a Gaussian function whose STD is $\sigma_1 = k_1 R$ ($k_1$ is the ratio of the STD of the Gaussian along the normal over the radius $R$),

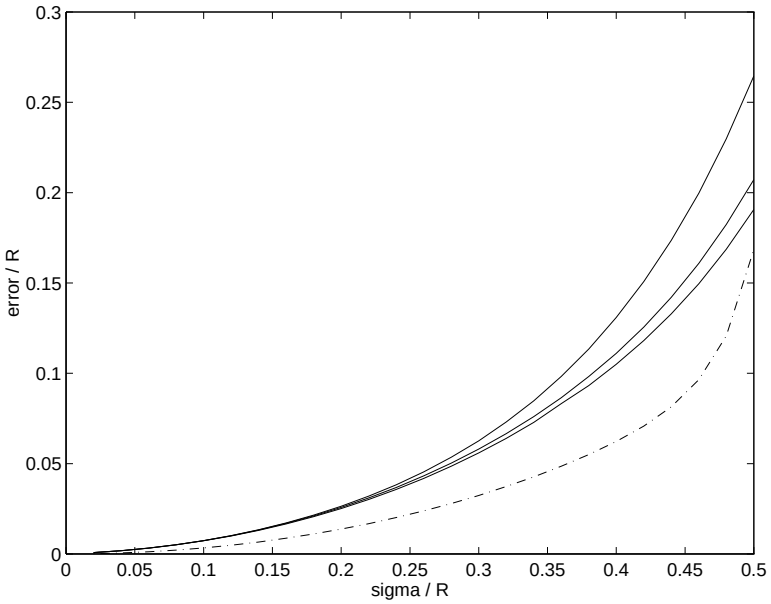
$$u(aR, 0) = \frac{M}{\sqrt{2\pi}\sigma_1} \exp\left(-\frac{(aR - R)^2}{\sigma_1^2}\right).$$

the bias depends on both $k_2$ and $k_1$. For $k_1 = 0.02, 0.2, 0.5$ respectively, we numerically solve equation 5.9 and we we plot bias against $k_2$ in figure 5.16. Figure 5.16.(a) shows the bias over $R$ against $k_2$; figure 5.16.(b) shows the bias over $\sigma_2$ against $k_2$. In figure 5.16, the dotted curves show the bias in the case of sharp edge; while the other three curve show the bias in the cases of smooth edge with different $k_1$'s. In both figures, the bigger $k_1$ is, the higher its corresponding curve is.

As we can see in these figures,

- The three curves with different $k_1$'s are very close; in other words, the effect of the actual shape of $u$ on bias is small. The bias depends more on the shape of the curve.
- In most images, the curvature of an edge is small enough (less than 0.5) that the circle fitting to the curve has $R > 2$. In this case, if we set $\sigma_2 = 1$ all the time, we get $k_2 \leq 0.5$. From figure 5.16.(b), we know that the ratio of bias over $\sigma_2$ will be less than 0.6 when $k_2 \leq 0.5$, which means the bias will be smaller than 0.6 pixel except at high curvature parts. Since the bias is so small, we can ignore it. We will show the effect of scale of $\sigma_2$ in experiments.

(a) $(1-b)$ vs. $k_2$




(b) $\frac{1-b}{k_2}$ vs. $k_2$

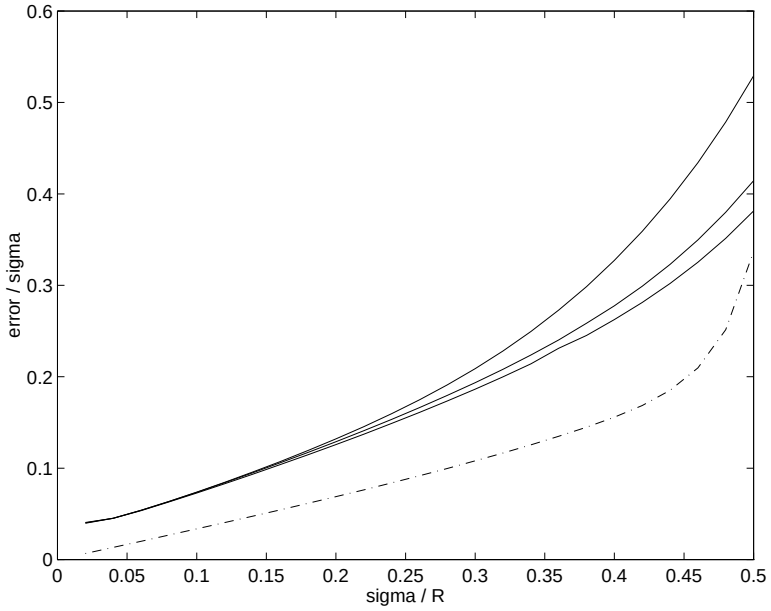



Figure 5.16: Error of $E_2$

### 5.4.4 Conclusion

In the section, we establish a measure function to compare points regarding how close they are to the center. The measure function is precise under two conditions:

- The edge is straight.
- Image intensity is constant along any line parallel to the edge.

We know when the first condition is not satisfied, the measure function incurs a very small bias. The second condition is roughly satisfied in most images since the change along the curve is usually much smaller than the change along the normal. So, the measure function should be a reasonably good one. Instead of proving this in theory, we use experiments to test the performance of the algorithm and the measure function together in chapter 6.

## 5.5 How to Compare Curves as a Whole

In the last step of the algorithm in section 5.3, we need to choose a curve from a group of curves that improves the most over curve $c_0$. We illustrate the problem by figure 5.17, in which $c_0$ stands for the final position of the evolving curve when we set $\alpha = \alpha_0$, and $c_k(k = 1, 2, \cdots)$ stands for the final position of the curve when we initialize it at $c_0$ and set $\alpha = \alpha_k$. The arrows in the figure show the displacement vectors of some points on the curve, and they form a point-to-point correspondence between curve $c_k$ and $c_0$. We want to define an comparison functional $E_2$ to determine which curve *as a whole* fits the center the best.

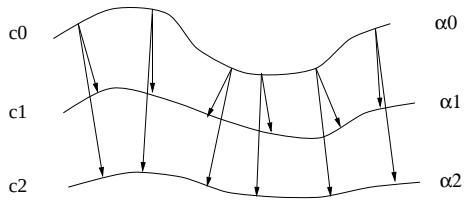



Figure 5.17: Comparison of curves

### 5.5.1 Generalize the Problem

We can generalize the problem as the following:

1. There a *base curve* $c_0$ and a group of curves $c_1, c_2, \cdots, c_k, \cdots$. In addition, we know a point-to-point correspondence between each curve $c_k$ in the group and curve $c_0$. For a point $c_0(q)$ on curve $c_0$, we denote its corresponding point on curve $c_k$ by $c_k(q) = \Phi_k(c_0(q))$, where $\Phi_k$ is a correspondence function.

2. With a measure function $\tilde{P}$, we can determine whether position $c_k(q)$ is better than position $c_0(q)$ by comparing $\tilde{P}(c_k(q))$ and $\tilde{P}(c_0(q))$. Without loss of generality, we assume points with higher $\tilde{P}$ as better.

With the above information, we want to define a comparison functional $E_2(c_k)$ to determine whether curve $c_k$ is better than curve $c_0$ as a whole; furthermore, we want to find the curve in the group that improves the most over curve $c_0$.

### 5.5.2 Two Straightforward Solutions

First of all, we can use the ideas in arc-length active contours [9, 59] and arc-length-parameterized active contours [16, 69, 78, 99, 108] to solve this problem. In both frameworks, a curve has an energy functional, and curves with lower energy are better. The

energy functional is the integral of a potential function along the curve with respect to arc-length. The definition does not need the point-to-point correspondence. To apply the idea to our problem, we define

$$E_2(c) = \int_0^l \frac{1}{1+\tilde{P}(c(s))} ds\,, \tag{5.10}$$

and consider curves with lower $E_2$ as better. We can not define

$$E_2(c) = \int_0^l \tilde{P}(c(s))\, ds \text{ (wrong definition)}$$

and take higher value as better, because this definition has no upper bound and we can always add a loop to a curve to make its $E_2$ higher.

Second, we can use the average of $\tilde{P}$ on the curve as the comparison functional:

$$E_2(c) = \frac{\int_0^l \tilde{P}(\vec{c}) ds}{l}\,. \tag{5.11}$$

and consider curves with higher $E_2$ as better.

#### 5.5.2.1 The Two Solutions are Biased

First we define what is a biased $E_2$.

**Definition of Biased $\mathbf{E_2}$**: if we can find a curve $c_i$ and a curve $c_j$ such that for any $q \in [0,1)$ $\tilde{P}(c_i(q)) \geq \tilde{P}(c_j(q))$ (any point on $c_i$ is better than or equally good to its counterpart on $c_j$), but $E_2(c_i)$ is worse than $E_2(c_j)$, we say the functional $E_2$ is biased.

Similarly to the external term in arc-length active contours, the functional in equation 5.10 is arc-length biased. For two curves in an image where $\tilde{P}$ is constant,

we set up an arbitrary point-to-point correspondence. Any pair of points have the same value of $\tilde{P}$, but shorter curve has lower (better) $E_2$.

The functional in equation 5.11 is also biased. In images where $\tilde{P}$ in one part is higher than $\tilde{P}$ in the other part, if the curve squeezes more and more of itself into the higher $\tilde{P}$ region, the part of curve in that region will have bigger fraction of the whole curve, and thus its $E_2$ value decreases. We illustrate the effect in figure 5.18, where the

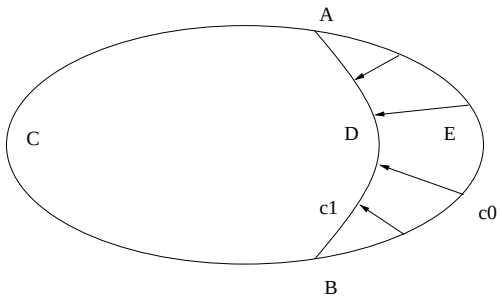



Figure 5.18: $E_2$ as the average $\tilde{P}$ is biased.

eclipse $CAEB$ is denoted by $c_0$. Suppose segment $AEB$ of $c_0$ evolves to $ADB$ as the arrows represent the paths; while the other segment $ACB$ does not move. We denote the new position by $c_1$. Suppose length$(ADB) = 4$, length$(AEB) = 5$, length$(ACB) = 9$; and $\tilde{P}$ is constant on each of the three segments with $\tilde{P}(C) = 1$, $\tilde{P}(D) = 10$, and $\tilde{P}(E) = 9$ According to equation 5.11, $E_2(c_1) = 3.77 < E_2(c_0) = 3.86$. In other words, while each individual point on curve $c_0$ either remains the same or evolves to a better position, the whole functional gets worse. That is, this definition is biased.

### 5.5.3 A New Approach

We propose a new method which uses the arc-length of curve $c_0$ for all the curves to construct $E_2$ –

$$E_2(c_k) = \int_0^1 \tilde{P}(\Phi_k(c_0(q)))\,||c_0'(q)||\,dq\ . \tag{5.12}$$

and considers curves with higher $E_2$ as better.

**Proposition:** This definition is not biased.

Proof: if $\forall q$, $\tilde{P}(\vec{c}_i(q)) \leq \tilde{P}(\vec{c}_j(q))$, then

$$E_2(c_i) = \int_0^1 \tilde{P}(\vec{c}_i(q))||\vec{c}_0'||dq \leq \int_0^1 \tilde{P}(\vec{c}_j(q))||\vec{c}_0'||dq = E_2(c_j)$$

In the approach, the functional of every curve depends on $c_0$ and the correspondence function. If we choose a different base curve or the curves have a different correspondence, the relationship between curves may change. In other words, it does not define an absolute notion of best, but only a concept of goodness relative to a base curve. We compare the performance of this approach with the performance of the other two in experiments.

## 5.6 The Algorithm in Detail

Now, with all the pieces we have built, we write down the algorithm to choose an appropriate value for $\alpha$ and find its corresponding curve in details.

Given an image $I$ and an initial curve $c_{init}$, the algorithm works as the following:

1. Construct the gradient magnitude $u = \bigtriangledown I$, the potential function $P = -u$, and the measure function $\tilde{P} = u * G_2$.

2. Choose a very small value $\alpha_0$ (in our experiments, 0.1 works well).

   Set $\alpha = \alpha_0$, and evolve an active contour from $c_{init}$ to $c_0$ according to equation 4.4.

3. $\alpha_{best} := \alpha_0$

   $c_{best} := c_0$

   $k := 1$

4. $\alpha := 1.5\alpha$. Evolve the active contour from $c_0$ to a new equilibrium position $c_k$ according to equation 4.4.

5. If the average (or maximal) displacement of the points is bigger than a certain threshold, stop the program and return $\alpha_{best}$ and $c_{best}$; otherwise, continue.

6. If $E_2(c_k)$ is better than $E_2(c_{best})$, $\alpha_{best} := \alpha$, and $c_{best} := c_k$.

7. $k := k + 1$.

8. Go to step 4.

# Chapter 6

# Experiments

In this chapter, we report our experiments on the new active-contour algorithm and the parameter-estimation algorithm. We tested them on both synthetic and medical images. On synthetic ones, we compared the results with the ground truth; on medical ones, it is hard to evaluate the performance, so we include the medical images in the thesis to let the reader judge the performance by him- or her-self.

## 6.1 On Synthetic Images

In this section, we report the experiments on synthetic images.

### 6.1.1 Experiment Setup

We tried to keep the simulations as realistic as possible. We created an image that had features which are known to be troublesome for active-contour algorithms – corners, variable contrast, etc. Further, because it is common to regularize the image before calculating its gradient, we included a regularization step. Finally, some heuristics were used to ensure that the active contour was initialized in the same way for different algorithms.

The details of the experiment setup are as follows.

#### 6.1.1.1 Image Synthesis

The synthetic image was created through the following steps:

1. We began with a hand-drawn curve which had varying roughness, significant indentations, and sharp corners. The curve is shown in Figure 6.1.(a).

2. The region inside the curve was filled with a uniform gray level of 150. The upper half of the background region was filled with a gray level of 200 and the lower half was filled with the gray level of 100. This had the effect of changing the sign of the edge contrast between the upper and lower halves. Further, it introduced two T-junctions along the boundary.

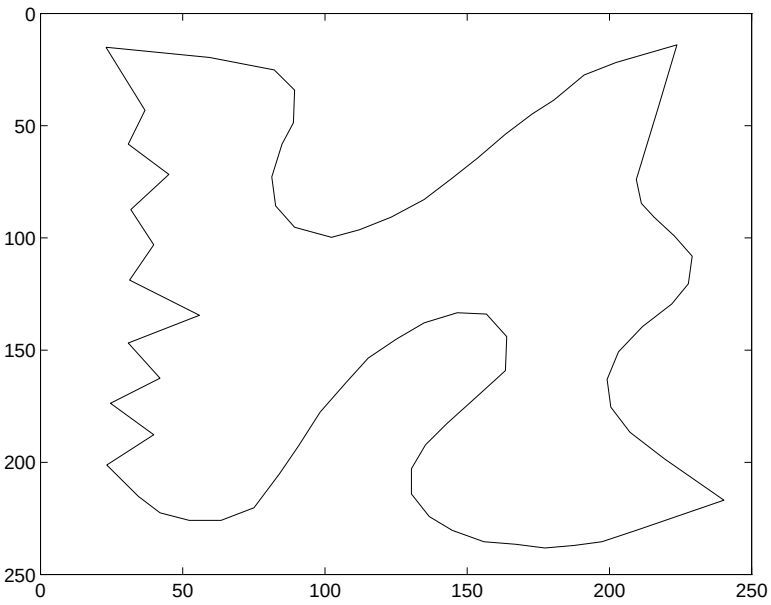
3. The resulting image was multiplied pixel-wise by a "gain" function $g(x, y)$ given by

$$g(x, y) = 0.33 \times (1 - \frac{x}{X}) + 1.0 \times \frac{x}{X}, \qquad 0 \leq x \leq X,$$

   where $X = 250$ is the maximum number of pixels in the horizontal direction. The effect of this is to multiply every pixel with a factor which is 0.33 on the left edge of the image and which linearly increases to 1.0 on the right edge of the image. Pixels that lie on the same vertical line are multiplied by the same number. This simulates variable contrast in the image.

4. The image was convolved with a 2-D Gaussian kernel. This makes the edge blurred and simulates finite scale for edges. In the experiments, we set the STD of the Gaussian kernel to 1.0. At the end of this stage, we generated a noiseless image, as shown in figure 6.1.(b).

(a) The synthetic boundary




(b) The image before adding noise

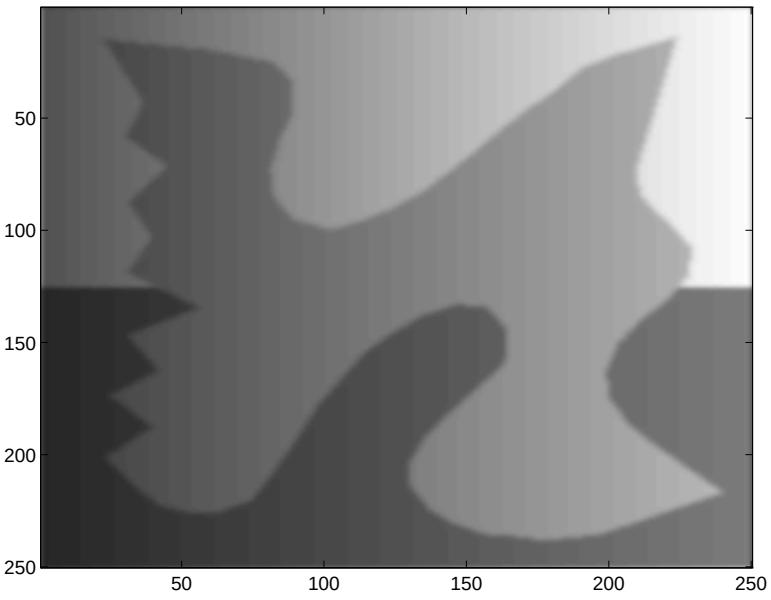



Figure 6.1: The image used in simulations.

5. Finally, independent uniform noise in $[-M, M]$ was added to the noiseless image to generate noisy images.

#### 6.1.1.2 Image Regularization

Each noisy image was regularized by convolving it with a 2-D Gaussian of size 3 pixels $\times$3 pixels. As mentioned before, this step was added because it is common practice to regularize an image before calculating its gradient.

#### 6.1.1.3 Contour Initialization

All active contours were initialized as circles and initially moved under a constant expansion force (a "balloon" force [14, 15]). As a contour entered regions of high gradient, the expansion force was switched off and the normal evolution equation applied. This is just a heuristic initialization strategy that was helpful in our experiments. It has no bearing on the final location of the contour. Almost any other initialization strategy would work as well.

#### 6.1.1.4 Performance Measures

Two different performance measures were used:

1. Length measure: A classic sign of bias is that the active contour becomes stationary in a shape that is consistently smaller or bigger than the true answer. To measure this, we calculated the percentage difference of lengths of the active contour and the true boundary contour:

$$\text{length error} = 100 * \frac{|\text{length(active contour)} - \text{length(boundary)}|}{\text{length(boundary)}}.$$

2. Area measure: We denote the region inside the true boundary as $R_0$ and the region inside the active contour as $R$. The percentage area of difference is computed as:

$$\text{area error} = 100 * \frac{\text{area}((R - R_0) \cup (R_0 - R))}{\text{area}(R_0)}.$$

We take area of difference a better measure than difference of area for the following reason: when two regions have the same area, or say their difference of area is 0, they can still be different; but when the area of the difference of the two is 0, the two regions are identical by definition. Figure 6.2 illustrate this property: in the figure, curve $c$ is drawn by a solid curve, and curve $c_0$ is drawn by a dashed curve. The regions surrounded by the two curves have the same size, but they are very different.

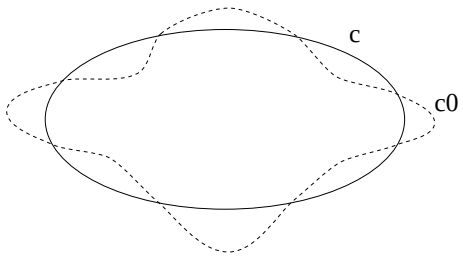



Figure 6.2: Two regions with the same area.

### 6.1.2 Experiments on Different Active Contours

The first set of experiments is to compare different active-contour algorithms and demonstrate the effects of Euclidean and non-Euclidean arc lengths on bias. Experimental

evidence of instability was discussed in section 3.3.

#### 6.1.2.1 The Active-Contour Models

Three active-contour algorithms were compared in the experiments. For reference, they are called AC1, AC2, and AC3.

**AC1:** The active contour AC1 has the non-Euclidean arc-length local energy proposed in section 3.2.2. Its evolution equation is (4.4), with $P(u) = -u$.

**AC2:** The active contour AC2 has the traditional energy with Euclidean arc-length integral (equation 3.14). Its evolution equation is (3.15), with $P(u) = -u$.

**AC3:** The active contour AC3 also has the energy of equation (3.14). Its evolution equation is (3.15), with $P(u) = \frac{1}{1+u}$.

To be fair, for each contour the images were scaled by a constant that kept the maximum force on each active contour the same.

#### 6.1.2.2 Simulated Combinations

Six values of $\alpha$ were chosen for simulations: $\alpha = 0.1, 1.0, 2.0, 3.0, 4.0, 5.0$; and, three noise levels were selected: $M = 0.0, 10.0, 30.0$.

For $M = 10$ and $M = 30$, 25 noisy images were generated for each level, so we have a total of 51 images (1 noiseless one, 25 noisy ones with $M = 10$, and 25 noisy ones with $M = 30$).

The three algorithms with six values of $\alpha$ were tested on each of the 51 images. The active contours were initialized at the same position for each image. When an active contour became stationary, percentage difference of length and percentage area of difference were calculated.

#### 6.1.2.3 Results

Figure 6.3 - 6.5 show the average length measure for each combination of $\alpha, M$, and the active contour algorithm. The following trends can be observed:

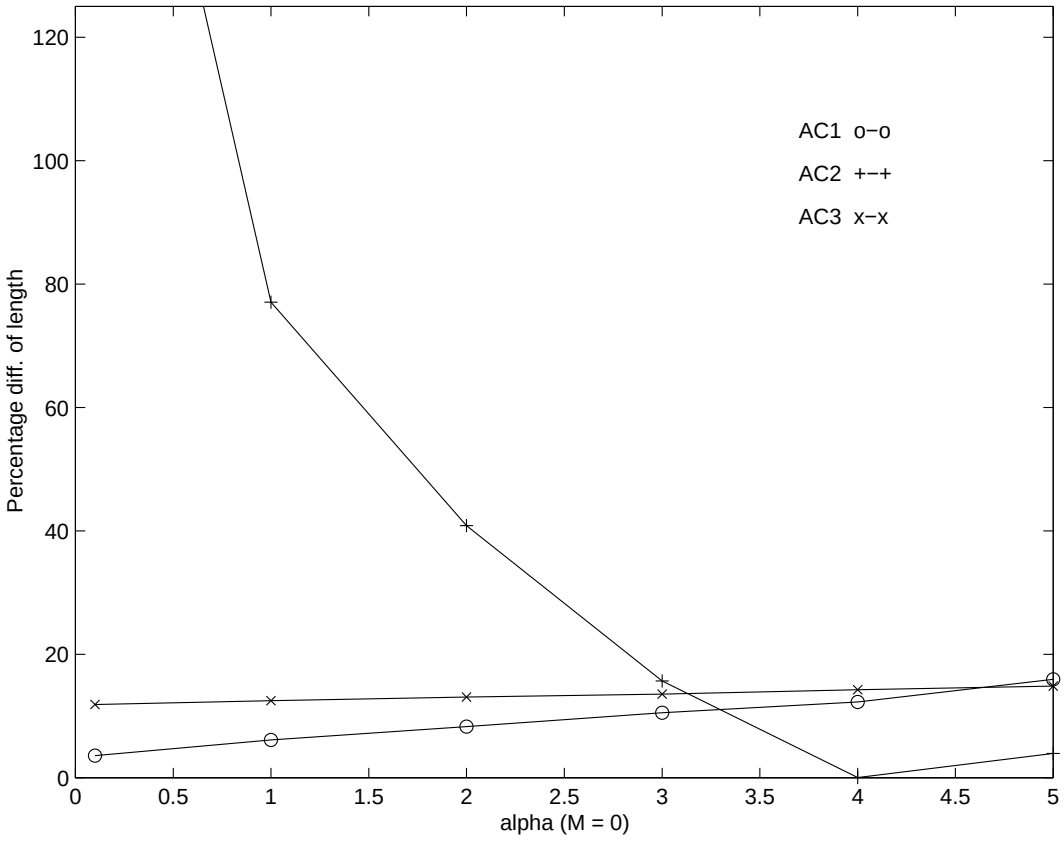



Figure 6.3: Length measures for different active contours ($M = 0$)

1. For low values of $\alpha$, AC2 makes large errors. This is consistent with our theoretical

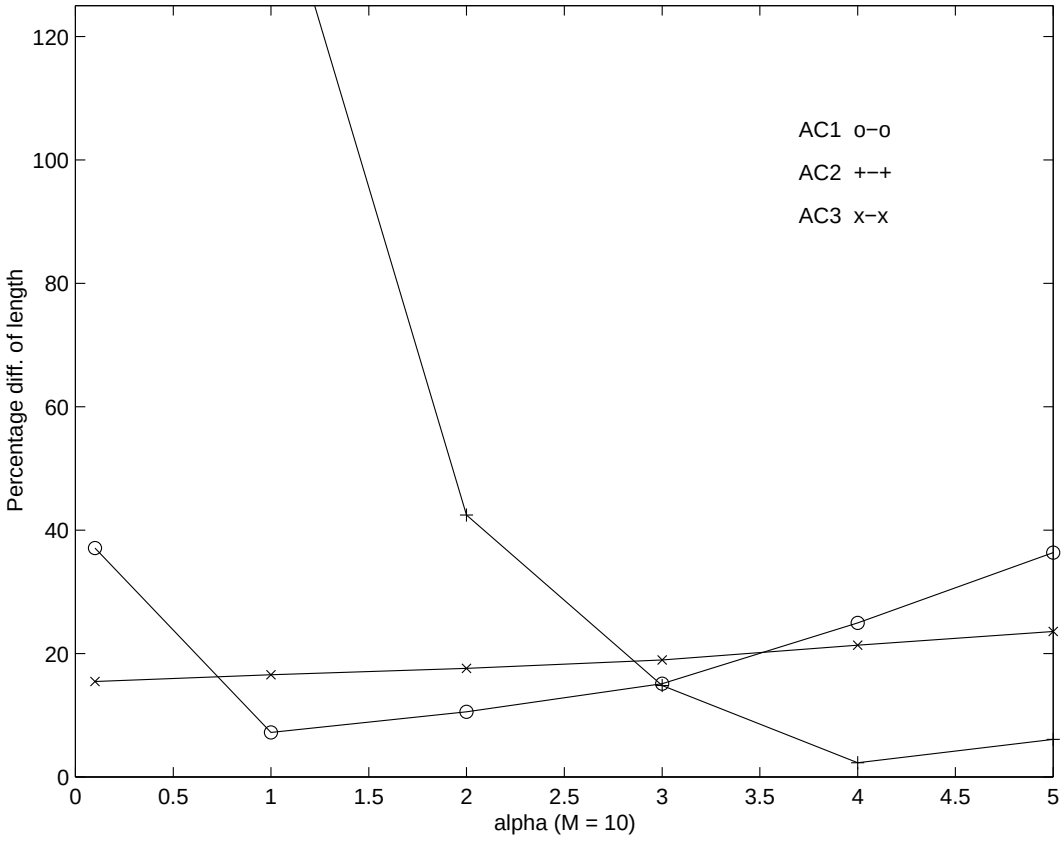



Figure 6.4: Length measures for different active contours ($M = 10$)

result that AC2 is unstable and that the instability is manifested by the contour becoming more jagged.

2. For low values of $\alpha$ and $M = 0$, AC1 performs far better than AC2 and AC3, clearly indicating low bias. This also consistent with the theoretical analysis.

   Figure 6.6 shows the active contours for $\alpha = 0.1$ and $M = 0$. The unstable AC2 results in jagged a curve; the biased AC3 rounds off corners.

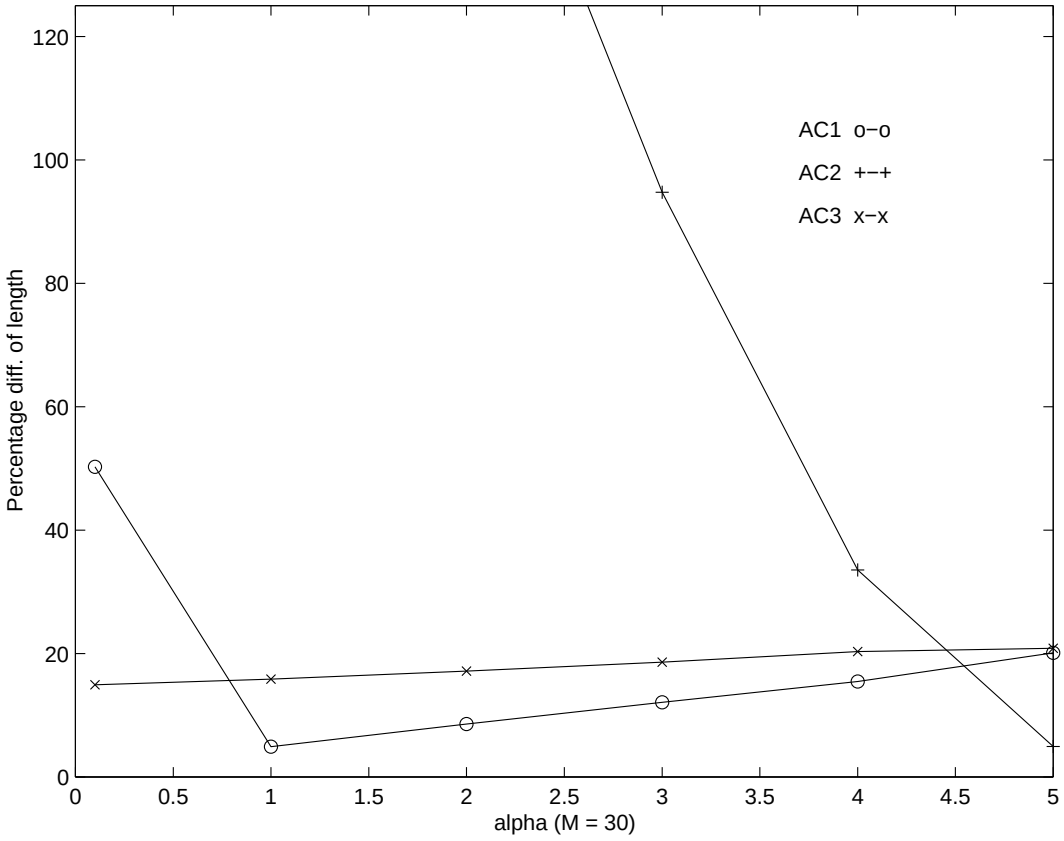


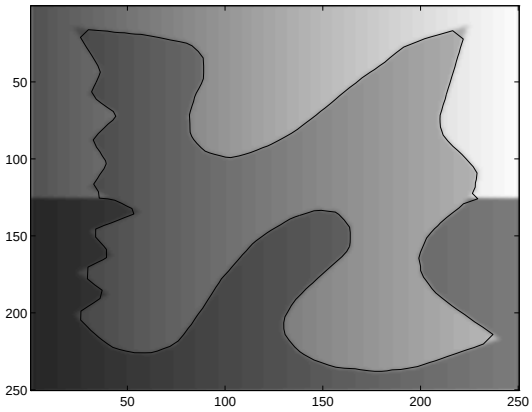

Figure 6.5: Length measures for different active contours ($M = 30$)

3. Further evidence for the theoretical analysis comes from considering the influence of $\alpha$ on AC1 and AC3.

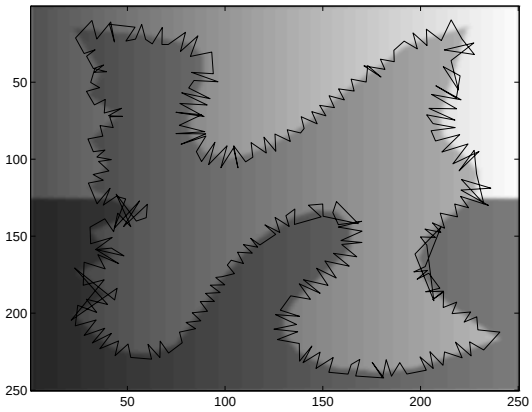
   Increasing $\alpha$ appears to have little effect on the error in AC3. This suggests that the bias due to the external energy of AC3 is large enough that the change in $\alpha$ does not affect it significantly.

   On the other hand, increasing $\alpha$ affects the performance of AC1 significantly. This

(a) AC1




(b) AC2




(c) AC3

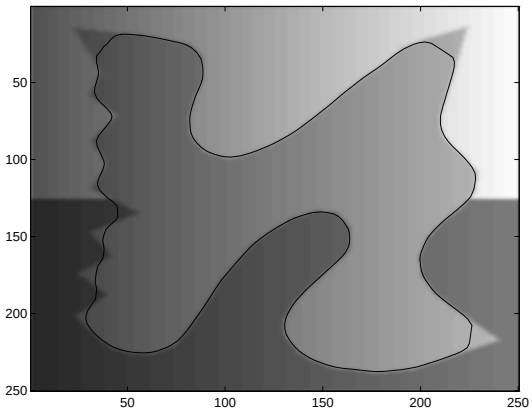



Figure 6.6: The location of active contours for $\alpha = 0.1$ and $M = 0$.

indicates: the external term has no bias, and therefore the shortening effect introduced by the internal term is significant.

Figure 6.7 - 6.9 show the average area measure for AC1,AC2 and AC3 under the conditions of simulation. Again the following trends support the analysis in previous chapters:

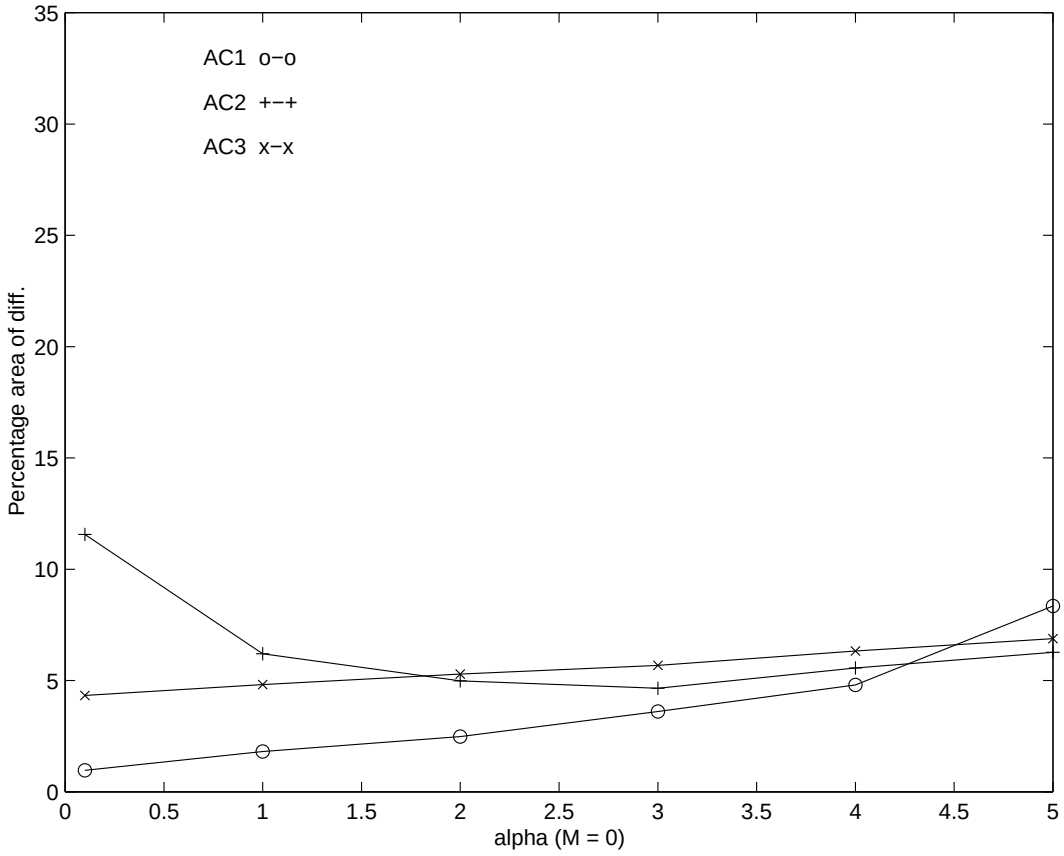



Figure 6.7: Area measures for different active contours ($M = 0$)

1. For the noiseless case ($M = 0$), the active contour AC1 outperforms the other two when $\alpha = 0.1$. This shows that the external energies of AC2 and AC3 have

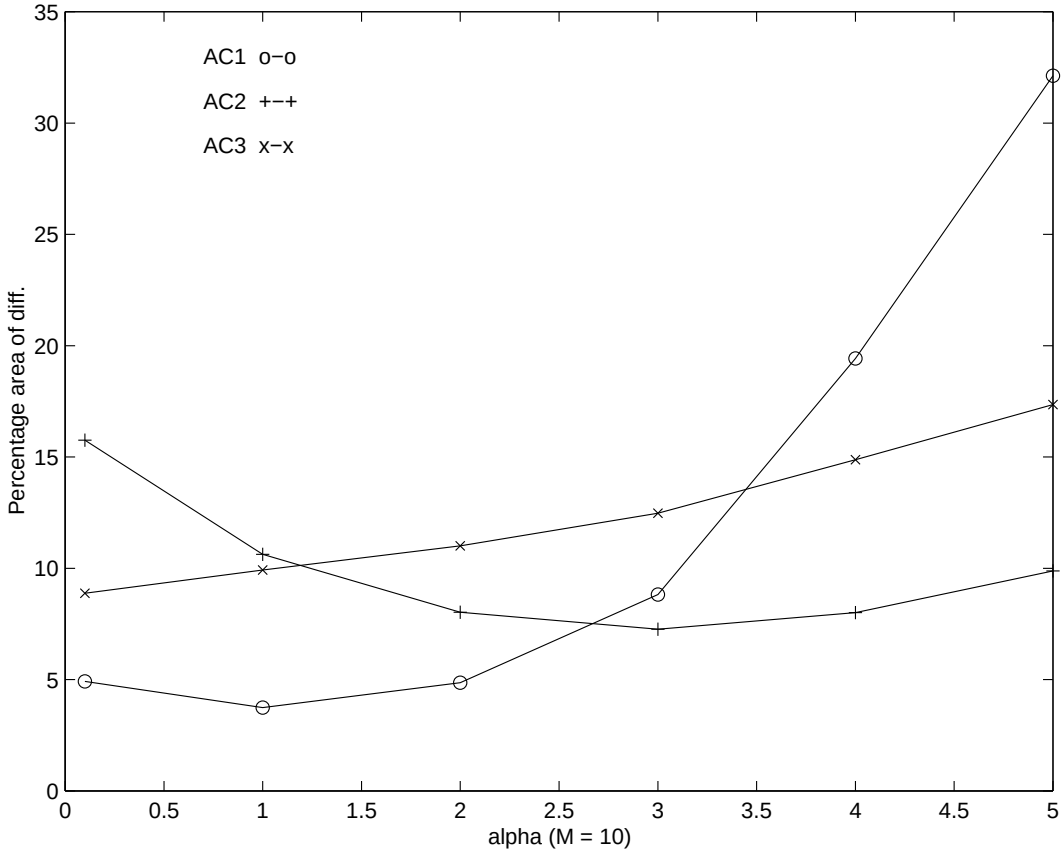



Figure 6.8: Area measures for different active contours ($M = 10$)

significantly more bias than that the external energy of AC1.

2. In contrast to length measure, the area measure for AC2 is of the same order of magnitude as the area measures for AC1 and AC3. Thus the size of AC2 is about the same as the size of AC1 and AC3; but its length is considerably larger. This is quantitative support for the observation that the poor stability of AC2 causes it to be more jagged in order to increase its arc length rather than its area.

3. For a given value of $M$, the lowest error (over different $\alpha$s) is always observed for

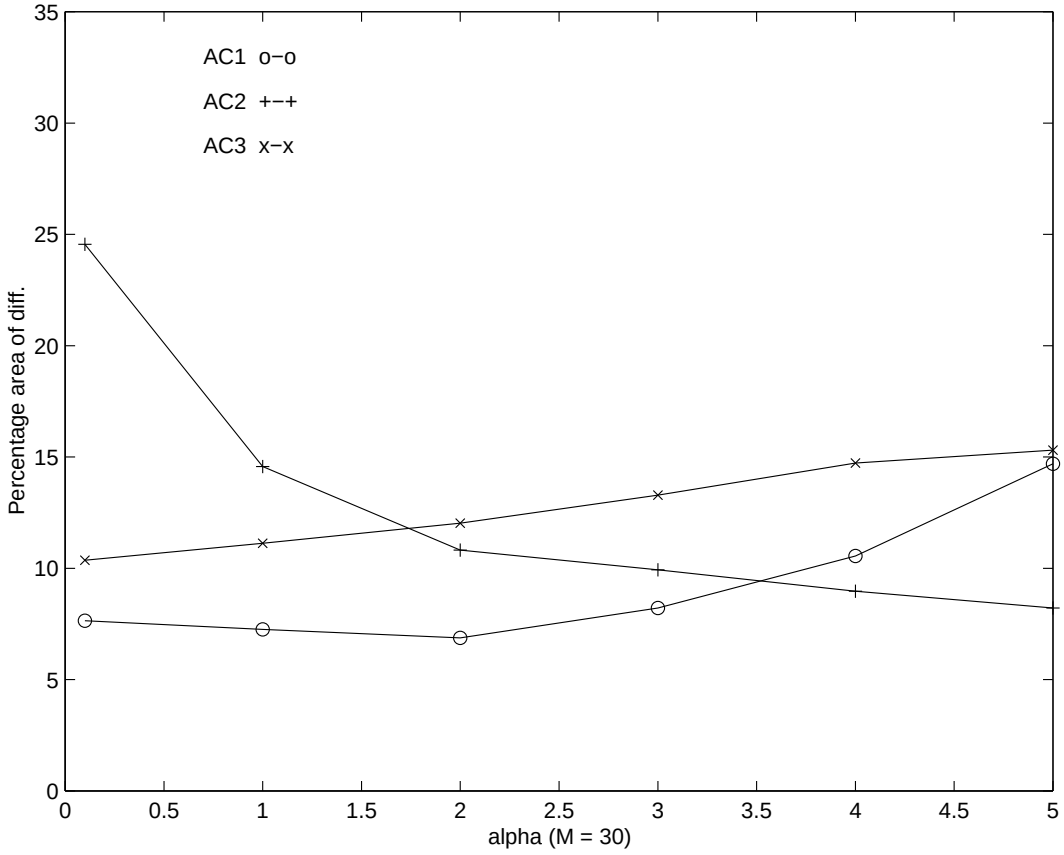


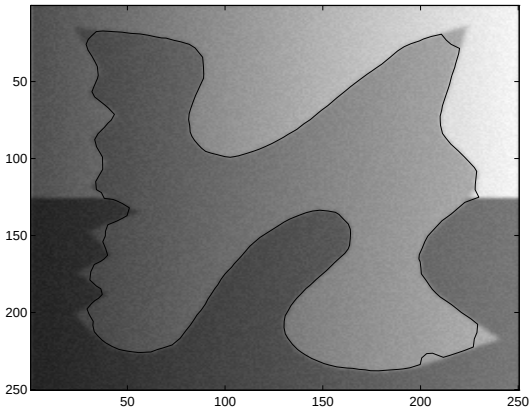

Figure 6.9: Area measures for different active contours ($M = 30$)

AC1. This is exactly the behavior we expect from an active-contour algorithm whose external energy is unbiased.

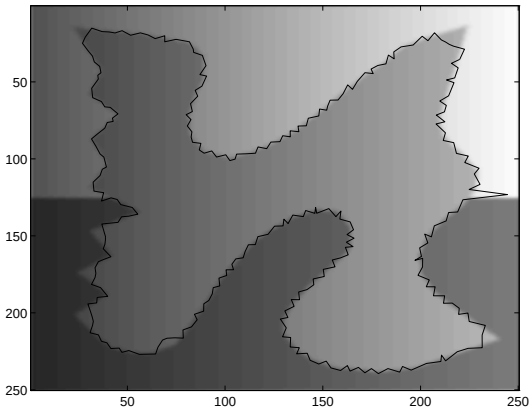
Figures 6.10-6.11 show the location of the active contours for the lowest error at $M = 10$ and $M = 30$.

The results of the set of experiments indicate that the choice of the external energy in active contours is important if we want to use the results of active contours in a quantitative analysis of the data. If we want to measure geometric features which

(a) AC1 ($\alpha = 1.0$)




(b) AC2 ($\alpha = 3.0$)




(c) AC3 ($\alpha = 0.1$)

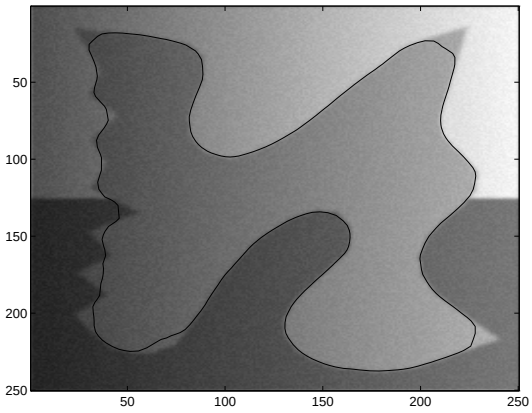
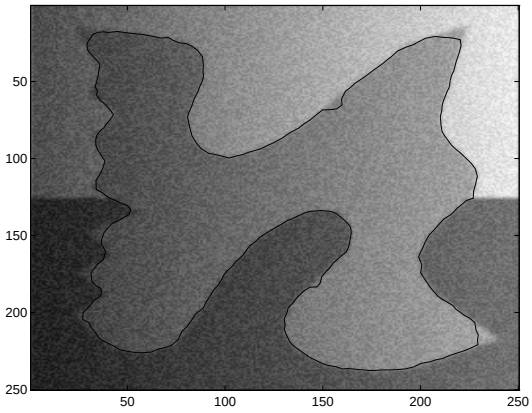
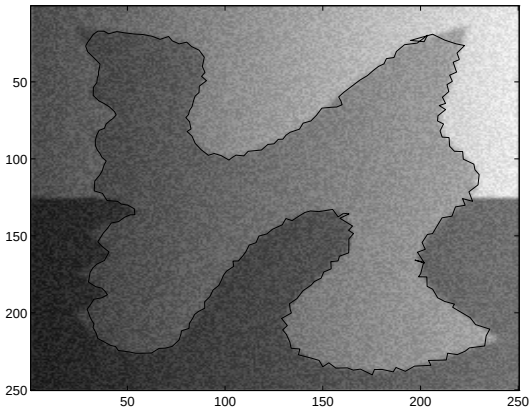



Figure 6.10: The minimum-error active contours for $M = 10$.

(a) AC1 ($\alpha = 2.0$)




(b) AC2 ($\alpha = 5.0$)




(c) AC3 ($\alpha = 0.1$)

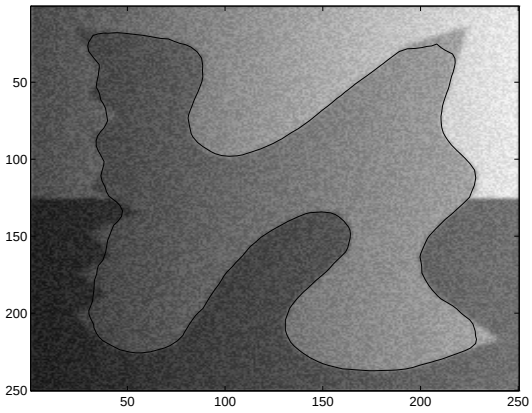



Figure 6.11: The minimum-error active contours for $M = 30$.

depend on arc length or area, then it is appropriate to select the active contour AC1.

### 6.1.3 Experiments on the Parameter-Estimation Algorithm

In the second set of synthetic-image experiments, we tested the parameter-estimation algorithm compared its results to the results of the active contours with fixed $\alpha$s. We also tested the algorithm with different values of $\sigma_2$ so that we can understand its effect on the results.

#### 6.1.3.1 Different Formulations of the Comparison Functional

All of three formulations of the comparison functional, which were discussed in chapter 5, were tested. For reference, they are called $E_2$, $E_2^{arc}$, and $E_2^{ave}$:

$E_2(c_k)$ is defined as the integral of $\tilde{P}$ along $c_k$ with respect to the arc-length of $c_0$ (equation 5.12). Higher means better.

$E_2^{arc}(c_k)$ is defined as the integral of $\frac{1}{1+\tilde{P}}$ along $c_k$ with respect to its own arc length (equation 5.10). Lower means better.

$E_2^{ave}(c_k)$ is defined as the integral of $\tilde{P}$ along $c_k$ with respect to its own arc length divided by the length of $c_k$ (equation 5.11). higher means better.

#### 6.1.3.2 Simulated Combinations

We tested on two levels of noise ($M = 10 \,\text{and}\, M = 30$) and three values of $\sigma_2$ ($\sigma_2 = 0.5, 1.0, \,\text{and}\, 3.0$). The experiment was carried by the following steps:

1. At each noise level, we used the 25 noisy images generated in the previous set of experiments.

| | mean measure (mean $\alpha^*$) | | |
|---|---|---|---|
| $\sigma_2$ | $E_2$ | $E_2^{arc}$ | $E_2^{ave}$ |
| 0.5 | 6.01(0.91) | 6.68(1.24) | 6.62(1.19) |
| 1.0 | 6.02(0.91) | 7.10(1.50) | 6.79(1.27) |
| 3.0 | 6.61(1.23) | 1.08(3.28) | 7.34(1.58) |

Table 6.1: Length measure at $M = 10$.

| | mean measure (mean $\alpha^*$) | | |
|---|---|---|---|
| $\sigma_2$ | $E_2$ | $E_2^{arc}$ | $E_2^{ave}$ |
| 0.5 | 2.27(0.91) | 2.33(1.24) | 2.32(1.19) |
| 1.0 | 2.27(0.91) | 2.42(1.50) | 2.38(1.27) |
| 3.0 | 2.30(1.23) | 4.11(3.28) | 2.52(1.58) |

Table 6.2: Area measure at $M = 10$.

2. We ran the algorithm with all combinations of $\sigma_2$ and comparison functionals. For each combination, the algorithm returns an optimal $\alpha$ ($\alpha^*$) and an optimal curve ($c^*$) on one image.

3. We computed the mean of $\alpha^*$s and the mean of the length and area measures of $c^*$s for each combination of $\sigma_2$ and comparison functional.

| | mean measure (mean $\alpha^*$) | | |
|---|---|---|---|
| $\sigma_2$ | $E_2$ | $E_2^{arc}$ | $E_2^{ave}$ |
| 0.5 | 8.37(2.35) | 11.9(3.28) | 9.99(2.75) |
| 1.0 | 8.57(2.41) | 12.7(3.67) | 9.96(2.73) |
| 3.0 | 8.35(2.36) | 16.3(5.88) | 10.1(2.85) |

Table 6.3: Length measure at $M = 30$.

| | mean measure (mean $\alpha^*$) | | |
|---|---|---|---|
| $\sigma_2$ | $E_2$ | $E_2^{arc}$ | $E_2^{ave}$ |
| 0.5 | 5.10(2.35) | 6.99(3.28) | 5.72(2.75) |
| 1.0 | 5.09(2.41) | 7.61(3.67) | 5.71(2.73) |
| 3.0 | 5.07(2.36) | 9.86(5.88) | 5.75(2.85) |

Table 6.4: Area measure at $M = 30$.

#### 6.1.3.3 Results

We present the results in table 6.1- 6.4. Each entry in the four tables shows two pieces of information: the first number is the average performance measure of $c^*$s, and the number in the parenthesis is the average $\alpha^*$. We see the following trends from the tables:

1. $E_2$ always achieves better results than the other two definitions, while the performance of $E_2^{ave}$ is very close to that of $E_2$.

2. Comparing the performance measures of the algorithm with figure 6.4, 6.5, 6.8,

and 6.9, we found that the algorithm with either $E_2$ or $E_2^{ave}$ gives very satisfying answers in the following sense: the average $\alpha^*$ is close to the best $\alpha$ obtained from figure 6.8 and 6.9; the average performance is close to the best performance of $AC1$.

Figure 6.12 gives the typical optimal curves found by the algorithm at $M = 10$, and figure 6.13 gives the typical optimal curves found by the algorithm at $M = 30$.

3. $E_2$ and $E_2^{ave}$ are insensitive to the change of $\sigma_2$. We explained the reason in chapter 5: the bias of $\tilde{P}$ is small and insensitive to the change $\sigma_2$.

4. $E_2^{arc}$ is sensitive to the change of $\sigma_2$. The reason is: $E_2^{arc}$ is arc-length biased, and the bias will increase when $\tilde{P}$ gets flatter, as we stated in Chapter 3; when we increase $\sigma_2$, though the bias of $\tilde{P}$ will not increase much, $\tilde{P}$ becomes flatter, so the arc-length bias increases a lot.

The results of this set of experiments indicate: the parameter-estimation algorithm finds reasonably good values for $\alpha$ with $E_2$ and $E_2^{ave}$, and the results are insensitive to the choice of $\sigma_2$. Therefore, the algorithm can guide the user to find an appropriate value for $\alpha$. While $E_2^{ave}$ is easier to implement than $E_2$ is, $E_2$ gives better results.

## 6.2 Medical Images

We tested the complete algorithm – the new active-contour algorithm and the parameter-estimation algorithm – on medical images.

Figure 6.14 is the nuclear image of a dog heart. Figure 6.15 is the MRI image of a dog heart. In the images, we first dropped a circle inside the region, as shown in the figures, and ran the balloon algorithm from there to find a rough position of the

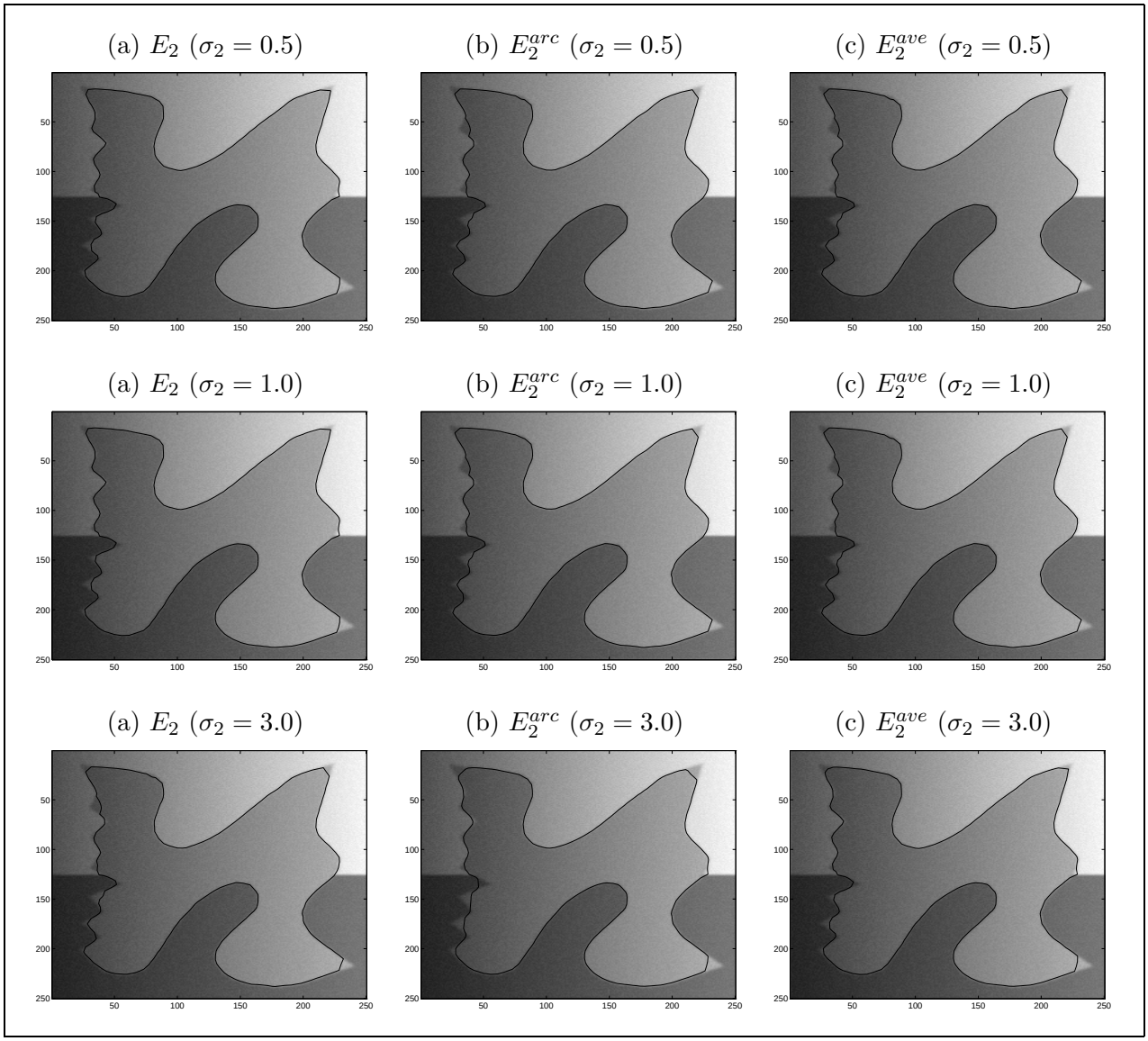



Figure 6.12: Typical optimal curves at $M = 10$

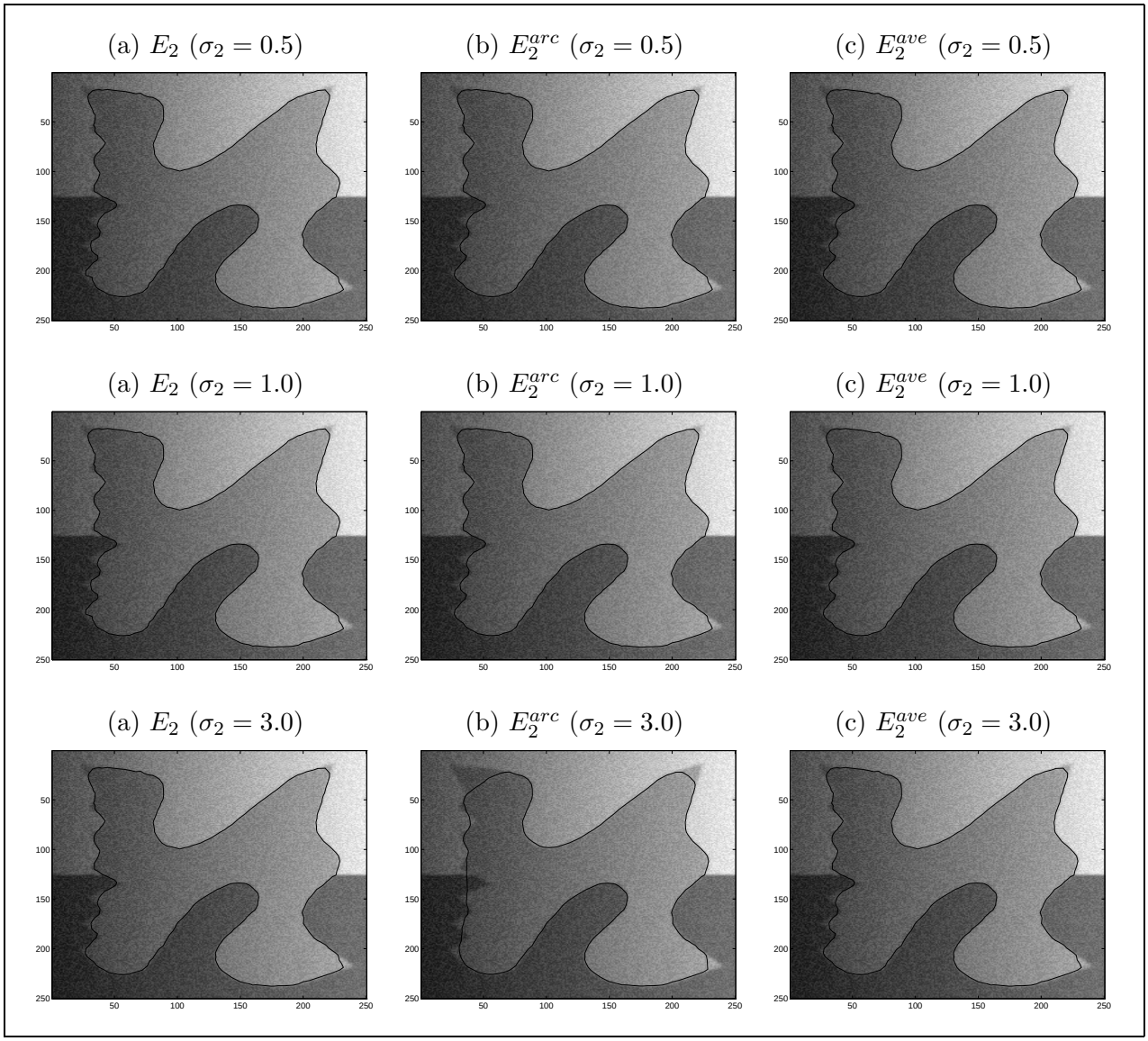



Figure 6.13: Typical optimal curves at $M = 30$

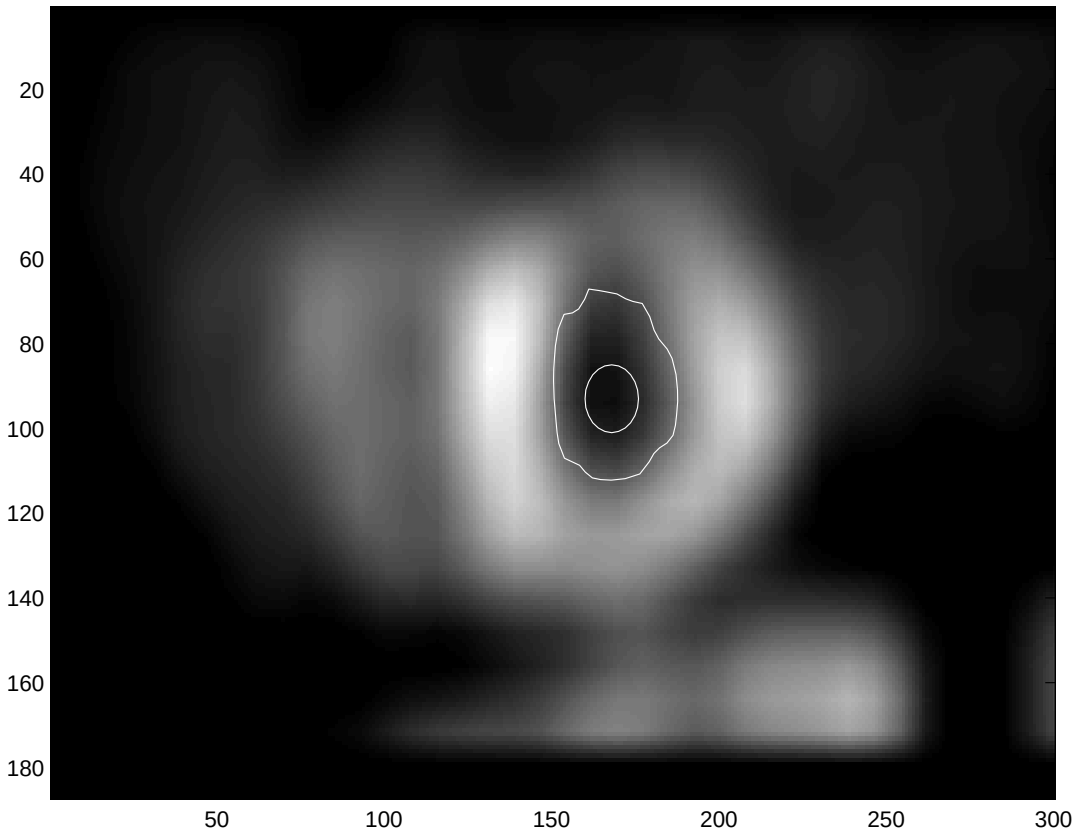
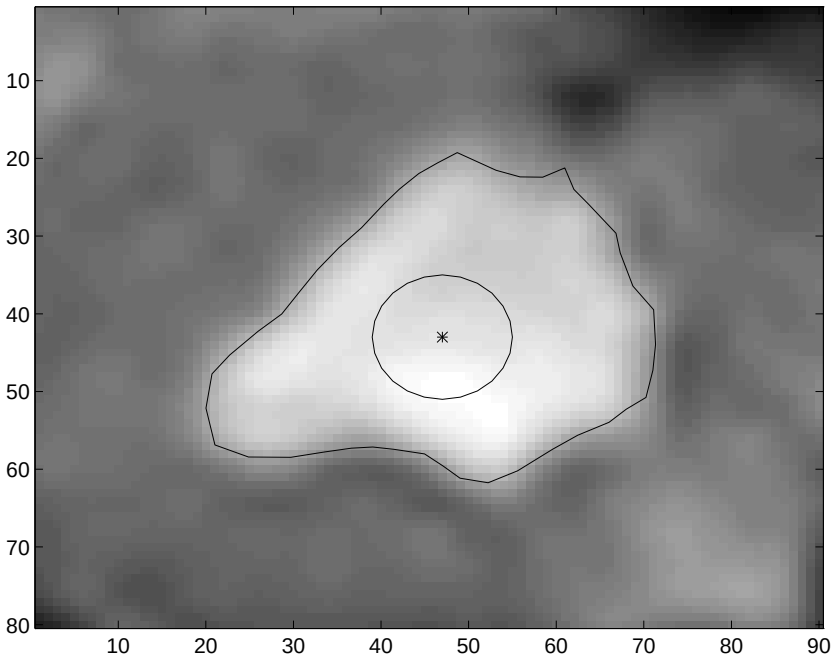



Figure 6.14: Nuclear image of heart (optimal $\alpha = 0.74$)

(a) Optimal curve on $I$




(b) Optimal curve on $P$

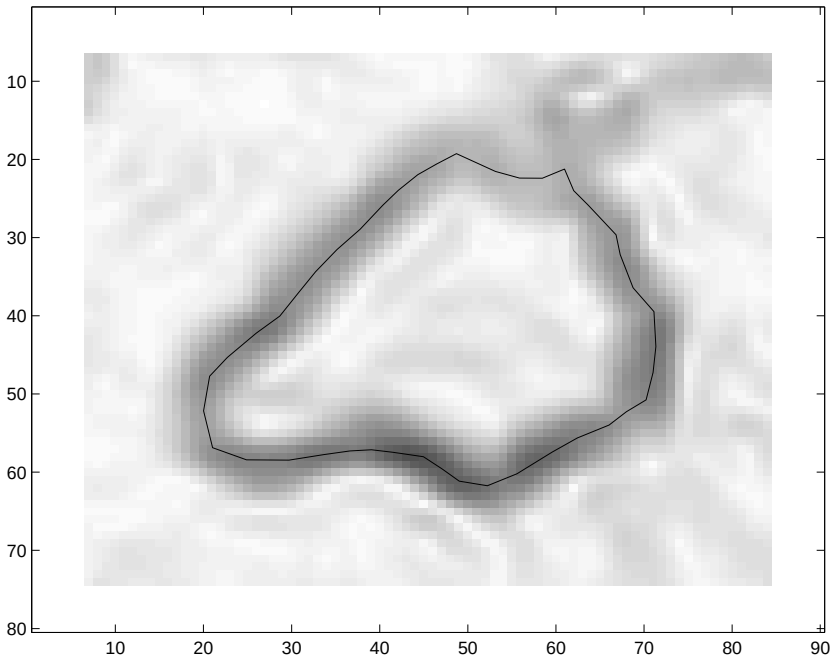



Figure 6.15: MRI image of heart (optimal $\alpha = 0.12$)

boundary. Then we ran our automatic active-contour algorithm from the position, and the final results are shown in the images. The results are quite good.

The boundary in the MRI image is quite clear, so we ran different active contour models with different $\alpha$'s on this one so that the reader can compare them with the optimal curve found by our algorithm. Figure 6.16 compares the results from $AC1$, $AC2$, and $AC3$ with manually set $\alpha$. The definitions of different active contours are in section 6.1.2.1. As we can see,

- The optimal curve in figure 6.15 is good compared with these curves found by manually set $\alpha$s.
- The valid range of $\alpha$ for these algorithms is as large as from 0.1 to 10, so the user has the difficulty to choose one appropriate value; therefore, a guideline will surely help.

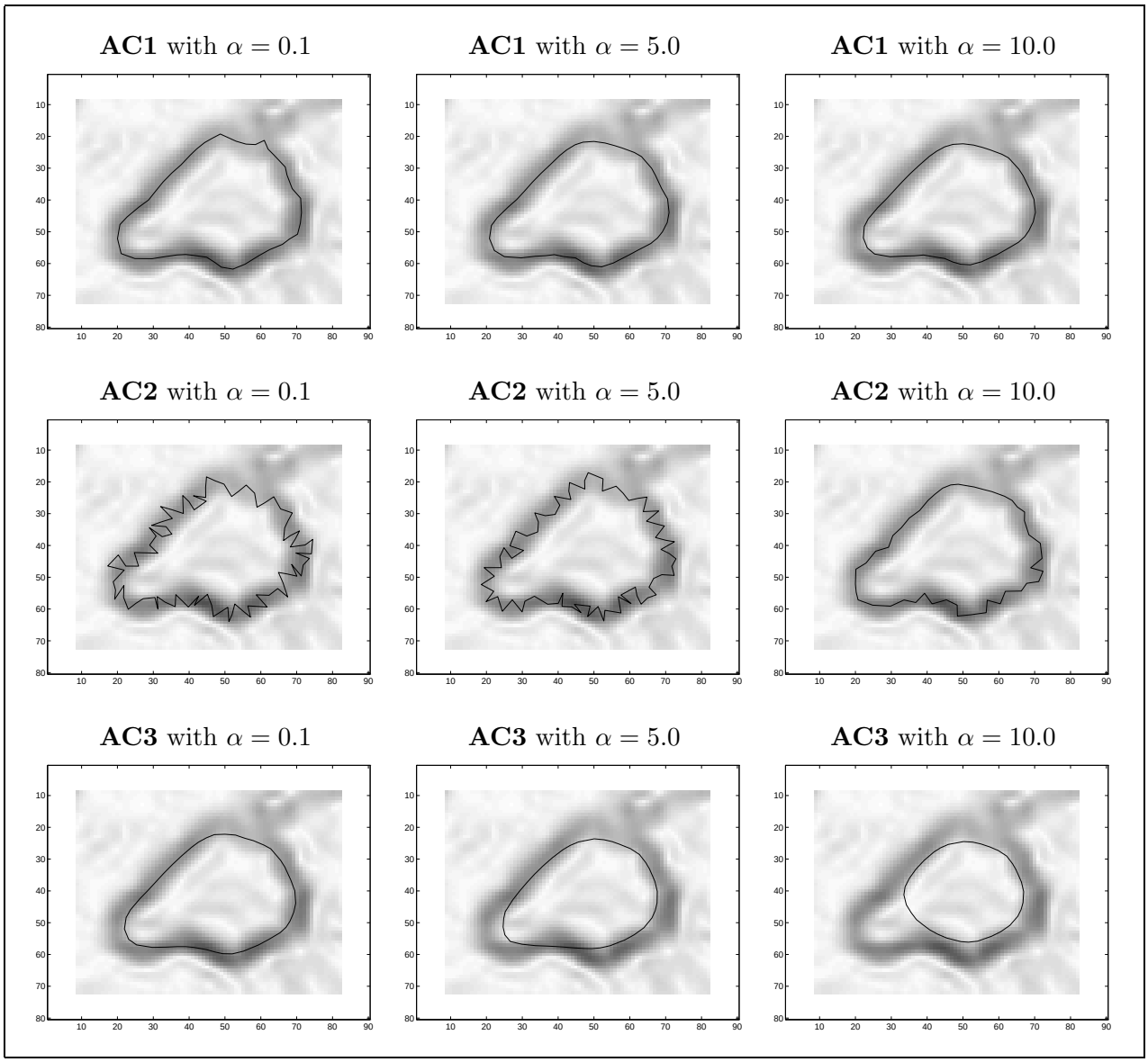



Figure 6.16: Comparisons on MRI image

# Chapter 7

# Summary

One important finding of this thesis is that energy functionals containing Euclidean arc-length integrals can be problematic in active-contour models. We proved theoretically that external energy functionals with such integrals are biased.

Further, certain external energy functionals with Euclidean arc-length integrals make the active contour unstable. We were able to show theoretically that the simple external energy functional $-\int || \bigtriangledown I||ds$ is unstable. An experimental demonstration of this was also provided.

One solution to these problems is to use a non-Euclidean arc length for the contour. The arc length can be defined such that the length of an infinitesimal piece of the contour does not change when it is pushed in the normal direction. This leads to a different formulation of active contours where the contour evolves as an integral curve of a vector field. The vector field is the gradient of a local energy functional. These active contours are not biased. This was proven theoretically as well as supported experimentally.

We believe that these active contours also overcome the stability problems of the

Euclidean arc-length contours, but we do not yet have a proof of it. The stability of variational formulations in active contours appears to be a complicated issue that has not yet been resolved.

Another important finding in the thesis is that there is another property of the gradient magnitude near an edge besides the one exploited by active contours. Namely, the gradient magnitude along the normal of an edge roughly forms a Gaussian function, and the edge is at the center of the Gaussian function.

The property enables us to develop a principle to choose $\alpha$: we choose an $\alpha$ whose corresponding result is best centered on a ridge of the gradient magnitude. After studying the structure of the set of equilibrium positions of active contours, we designed an algorithm to choose $\alpha$ discretely according to the principle.

The algorithm needs a comparison functional, which was established in two steps: first, we built a measure function for individual points, and proved that it is biased but the bias is small; then, we built a comparison functional to compare curves as a whole, defined what is biased in this situation, and proved that the functional is not biased.

In the experiments, we compared the new active-contour model with previous ones, and the new algorithm consistently outperforms the others. We also tested the parameter-estimation algorithm in the experiments, and the algorithm found reasonably good values for the smoothness parameter. The two pieces of work form a non-biased parameter-free algorithm to find object boundaries based on an initial position close to the boundary.

## Possible Extensions and Future Work

As we mentioned in chapter 2, there are 3D extensions of active-contour models. Currently, the external energy functionals in 3D contain Euclidean surface-area integrals. Similarly to their 2D counterparts, they are biased since they favor either large or small surfaces depending on the specific definition. They also need the user to choose some smoothness parameter. Our work can be extended to 3D naturally by replacing the Euclidean surface-area by a non-Euclidean one and studying the 3D-image structure near object boundaries. The extension will result in a non-biased 3D segmentation algorithm and an algorithm to automatically choose the smoothness parameter for it.

Our work is to find object boundaries formed by sharp change of image intensity. Another important type of object boundary is formed by the change of texture. If one can find operators on texture that are similar to $u$ and $\tilde{P}$ as on intensity, our work can be applied to texture segmentation directly.

Another possible extension of our work is to develop theory on varying $\alpha$ along the active contour. In an image, the noise level and the roughness of the boundary vary at different places, so it is natural to vary $\alpha$ along the object boundary instead of setting it to one value everywhere.

Finally and probably most importantly, the active-contour algorithms need a good initial position. There has been work on how to find the initial position for active contours [69]. Theory and techniques regarding the problem will be the key to apply active contours to more areas. By combining edge detection [8, 62], thresholding [54, 68, 73, 85, 102], balloon [14, 15], front propagation [59], and non-linear diffusion [75], one might be able to develop a robust algorithm to find the rough position of an object boundary which fits certain models.

# Bibliography

[1] L. Alvarez, P.L. Lions, and J.M. Morel, "Image Selective Smoothing and Edge Detection by Nonlinear Diffusion," *SIAM Journal of Numerical Analysis*, 29, pages 845-866, 1992.

[2] Amir A. Amini, Terry E. Weymouth, and Ramesh C. Jain, "Using Dynamic Programming for Solving Variational Problems in Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(9), pages 855-867, 1990.

[3] J. Babaud, A. Witkin, M. Baudin, and R. Duda, "Uniqueness of the Gaussian Kernal for Scale Space Filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8, 1986.

[4] Dana H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, 13(2), pages 111-122, 1981.

[5] D.H. Ballard and C.M. Brown, *Computer Vision*, Prentice Hall, Englewood Cliffs, 1982.

[6] A. Blake and A. Zisserman, *Visual Reconstruction*, MIT Press, Cambridge, MA, 1987.

[7] C.R. Brice and C. L. Fennema, "Scene analysis using regions," *Artificial Intelligence*, 1, pages 205-226, 1970.

[8] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8, pages 679-698, 1986.

[9] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro, "Geodesic Active Contours," *Proc. 5th ICCV*, pages 694-699, 1995.

[10] Amit Chakraborty, Lawerence H. Staib, and James S. Ducan, "Deformable Boundary Finding Influenced by Region Homogeneity," *Proc. CVPR*, pages 624-627, 1994.

[11] A. Charkraborty and J.S. Duncan, "Integration of Boundary Finding and Region-based Segmentation Using Game Theory," *XIVth International Conference on Information Processing in Medical Imaging*, pages 189-200, 1995.

[12] A. Chakraborty, *Feature and Module Integration for Image Segmentation*, PhD thesis, Yale Univeristy, 1996.

[13] C.C. Chu and J.K. Agarwal, "The Integration of Image Segmentation Maps Using Region and Edge Information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15, pages 1241-1252, 1993.

[14] L.D. Cohen, "On Active Contour Models and Balloons," *CVGIP: Image Understanding*, 53(2), pages 211-218, 1991.

[15] Laurent D. Cohen and Issac Cohen, "Finite-Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(11), pages 1131-1147, 1993.

[16] L.D. Cohen and R. Kimmel, "Global Minimum for Active Contour Models: A Minimal Path Approach," *Proc. CVPR*, pages 666-673, 1996.

[17] I. Cohen, L.D. Cohen, and N. Ayache, "Using deformable surfaces to segment 3D images and infer differential structure," *CVGIP: Image understanding* 56(2), pages 242-263, 1992.

[18] George R. Cooper and Clare D. McGillem, *Methods of Signal and System Analysis*, Hold, Rinehart and Winston, Inc.

[19] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham, "Active Shape Models – Their Training and Application," *Computer Vision, Graphics and Image Processing: Image Understanding*, 61, pages 38-59, 1995.

[20] R. Courant and D. Hilbert, *Methods of mathematical physics*, Wiley, New York, 1989.

[21] C. A. Davatzikos and J. L. Prince, "Adaptive Active Contour for Extracting and Mapping Think Curves," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 524-529, 1993.

[22] C. A. Davatzikos and J. L. Prince, "An Active Model for Mapping the Cortex," *IEEE Trans. Medical Imaging*, 14(1), pages 65-80, 1993.

[23] H. Derin and W.S. Cole, "Segmentation of Textured Images Using Gibb's Random Field," *Computer Vision, Graphics and Image Processing*, 35, pages 72-98, 1986.

[24] H. Derin and H. Elliot, "Modeling and Segmentation of Noisy and Textured Images Using Gibb's Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9, pages 39-55, 1987.

[25] H. Derin, H. Elliot, R. Cristi, and D. Geman, "Bayes Smoothing Algorithms for Segmentation of Binary Images Modeled by Markov Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, pages 707-720, 1984.

[26] Richard O. Duda and Peter E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, 15(1), pages 11-15, Jan. 1972.

[27] James S. Duncan and Lawrence H. Staib, "Shape Determination from Incomplete and Noisy Multisensor Imagery," in A. Kak and S. Chen, editors, *Spatial Reasoning and Multi-Sensor Fusion: Proceedings of the 1987 Workshop*, pages 334-344, Morgan Kaufmann, Los Altos, CA 1987.

[28] K.S. Fu and J.K. Mui, "A Survey of Image Segmentation," *Pattern Recognition*, 13, pages 3-16, 1981.

[29] P. Fua and Y.G. Leclerc, "Model Driven Edge Detection," *Machine Vision and Applications*, 3, pages 45-56, 1990.

[30] D. Geiger and A. Yuille, "A Common Framework for Image Segmentation," *International Journal of Computer Vision*, 6, pages 227-243, 1991.

[31] Davi Geiger, Alok Gupta, Luiz A. Costa, and John Vlontzos, "Dynamic Programming for Detecting, Tracking, and Matching Deformable Contours," *IEEE Trans. Pattern Annalysis and Machine Intelligence*, 17(3), pages 294-302, 1995.

[32] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6, pages 721-741, 1984.

[33] D. Geman, S. Geman, and P. Dong, "Boundary Detection by Constrained Optimization and the Bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12, pages 609-628, 1990.

[34] R.C. Gonzalez and P.A. Wintz, *Digital Image Processing*, Addison-Wesley, MA, 1983.

[35] Steve R. Gunn and Mark S. Nixon, "A Robust Snake Implementation; A Dual Active Contour," *IEEE Trans. Pattern Annalysis and Machine Intelligence*, 19(1), pages 63-68, 1997.

[36] J.F. Haddon and J.F. Boyce, "Image Segmentation by Unifying Region and Boundary Information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, pages 929-948, 1990.

[37] R.M. Haralick and L.G. Shapiro, "Survey, Image Segmentation Techniques," *Computer Vision, Graphics and Image Processing*, 29, pages 100-132, 1985.

[38] Berthold Horn, *Robot Vision*, MIT Press, Cambridge MA, 1986.

[39] S. L. Horowitz and T. Pavlidis, "Picture Segmentation by a Directed Split-and-merge Procedure," *Proceedings of the International Joint Conference on Pattern Recognition*, pages 424-433, August 1974.

[40] A. Hummel, B. Kimia, and S. Zucker, "Deblurring Gaussian Blur," *Computer Vision, Graphics and Image Processing*, 38, pages 66-80, 1987.

[41] J. Illingworth and J. Kittler, "A Survey of the Hough Transform," *Computer Vision, Graphics and Image Processing*, 44, pages 87-116, 1988.

[42] Ramesh Jain, Rangachar Kasturi, and Brian G. Schunck, *Machine Vision*, McGraw-Hill, 1995.

[43] P. Karaolani, G.D. Sullivan, and K.D. Baker, "Active Contours using Finite Elements to Control Local Shape," *British Machine Vision Conference*, 1992.

[44] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, "Snakes: Active Contour Models," *Int. J. Computer Vision*, 1, pages 321-331, 1988.

[45] S Kichenassamy, A Kumar, P Olver, A Tannenbaum, and A Yezzi, "Gradient Flows and Geometric Active Contour Models," *Proc. 5th ICCV*, pages 810-815, 1995.

[46] B. B. Kimia, A. Tannenbaum, and S. W. Zucker, "Snakeps, shocks, and deformations," *Int. J. Computer Vision.*

[47] J. Kitter and J. Illingworth, "On Threshold Selection Using Clustering Criteria," *IEEE Transactions on Sysmtem, Man and Cybernectics*, 15, pages 652-655, 1985.

[48] R. Kohler, "A Segmentation Based on Thresholding," *Computer Vision, Graphics and Image Processing*, 15, pages 319-338, 1981.

[49] J. Koenderink, "The Structure of Images," *Biological Cybernetics*, 50, pages 363-370, 1984.

[50] Kok F. Lai and Roland T. Chin, "Deformable Contours: Modeling and Extraction," *Proc. CVPR*, pages 601-608, 1994.

[51] S. Lakshmanan and H. Derin, "Simlultaneous Parameter Estimation and Segmentation fo Gibbs Random Feilds Using Simulated Annealing," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 11, pages 799-813, 1989.

[52] O. V. Larsen, P. Radeva, and E. Marti, "Guidelines for Choosing Optimal Parameters of Elasticity for Snakes," *Proc. of Computer analysis of Image and patterns*, pages 106-113, 1995.

[53] O. V. Larsen, P. Radeva, and E. Marti, "Bounds of the Optimal Elasticity Parameters for a Snake," *Proc. Image analysis and processing*, 1995.

[54] M.D. Levine and A.M. Nazif, "Dynamic Measurement of Computer Generated Image Segmentatoin," *IEEE Transactions on Systems, Man and Cybernectics*, 7, pages 155-164, 1985.

[55] Frederic Leymarie and Martin D. Levine, "Simulating the Grassfire Tranform Using an Active Contour Model," *IEEE Trans. Pattern Annalysis and Machine Intelligence*, 14(1), pages 56-75, 1992.

[56] Frederic Leymarie and Martin D. Levine, "Tracking Deformable Objects in the Plane Using an Active Contour Model," *IEEE Trans. Pattern Annalysis and Machine Intelligence*, 15(6), pages 617-634, 1993.

[57] L. Liu, G. Chunck, and C. R. Meyer, "Optimal contour approximation by deformable piecewise cubic splines," *Proc. CVPR*, pages 638-643, 1991

[58] David Lowe, "Three Dimensional Object Recognition from Single Two-dimensional Images," *Artificial Intelligence*, 31(3), pages 355-396, 1987.

[59] Ravikanth Malladi, James A. Sethian, and Baba C. Vemuri, "Shape Modeling with Front Propagation: A Level Set Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(2), pages 158-175, 1995.

[60] B.S. Manjunath and R. Chellappa, "Unsupervised Texture Segmentation Using Markov Random Field Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13, pages 478-482, 1991.

[61] David Marr, *Vision*, W.H. Freeman and Company, New York, 1982.

[62] D. Marr and E. Hildreth, "Theory of Edge Detection," *Proceedings of the Royal Society of London*, B207, pages 187-217, 1980.

[63] Tim McInerney and Demetri Terzopoulos, "Topologically Adaptable Snakes," *Proc. 5th ICCV*, pages 840-845, 1995.

[64] D. Metaxas and E. Koh, "Efficient Shape Representation using Deformable Models with Locally Adaptive Finite Elements," *Geometric Methods in Computer Vision*, 2031, pages 160-171, 1993.

[65] Frank Morgan, *Riemannian Geometry: A Beginner's Guide*, Jones and Bartlet Publishers, 1993.

[66] W. Mulder, S. Osher, and J.A. Sethian, "Computing interface motion in compressible gas dynamics," *J. of Computational Physics*, 79, pages 12-49, 1988.

[67] D. Mumford and J. Shah, "Boundary Detection by Minimizing Functionals," *IEEE Conference on Computer Vision and Pattern Recognition*, pages 22, 1985.

[68] Y. Nakagawa and A. Rosenfeld, "Some Experiments on Variable Thresholding," *Pattern Recognition*, 11, pages 191-204, 1979.

[69] W. Neuenschwander, P. Fua, G. Szekely, and O. Kubler, "Initializing Snakes," *Proc. CVPR*, pages 658-663, 1994.

[70] Ramakant Nevatia and K. Ramesh Babu, "Linear Feature Extraction and Description," *Computer Graphics and Image Processing*, 13, pages 257-269, 1980.

[71] R. B. Ohlander, K. Price, and D. R. Reddy, "Picture Segmentation Using a Recursive Region Splitting Method," *Computer Graphics and Image Processing*, 8, pages 313-333, 1978.

[72] Stanley Osher, "Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *Journal of Computational Physics*, 79, pages 12-49, 1988.

[73] N. Otsu, "A Threshold Selection Method from Grey Level Histograms," *IEE Transactions on Systems, Man and Cybernetics*, 9, pages 62-66, 1979.

[74] N.R. Pal and S.K. Pal, "A Review on Image Segmentation Techniques," *Pattern Recognition*, 26, pages 1277-1294, 1993.

[75] P. Perona and J. Malik, "Scale-space and Edge Detection Using Anisotropic Diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, pages 629-639, 1990.

[76] T. Pavlidis and Y. Liow, "Integrating Region Growing and Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, pages 225-233, 1990.

[77] A.P. Pentland, "Automatic Extraction of Deform-able Part Models," *International Journal of Computer Vision*, 4, pages 107-126. 1990

[78] Petia Radeva, Joan Serrat, and Enric Marti, "A Snake for Model-Based Segmentation," *Proc. 5th ICCV*, pages 816-821, 1995.

[79] Anothai Rattarangsi and Roland T. Chin, "Scaled-Based Detection of Corners of Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(4), April 1992.

[80] B. Romeny, *Geometry Driven Diffusion in Computer Vision*, Kluwer, 1994.

[81] Remi Ronfard, "Region-Based Strategies for Active Contour Models," *Inter. J. of Computer Vision*, 13(2), pages 229-251, 1994.

[82] A. Rosenfeld, Robert A. Hummel, and Steven W. Zucker, "Scene Labelling by Relaxation Operations," *IEEE Transactions on Systems, Man and Cybernetics*, 6(6), pages 420-433, June 1976.

[83] A. Rosenfeld and A. Kak, *Digital Picture Processing, 2nd ed*, Academic Press, New York, 1982.

[84] H. Sagan, *Introduction to the Calculus of Variations*, McGraw-Hill Book Company, 1969.

[85] P.K. Sahoo, S. Soltani, A.K.C. Wong, and Y.C. Chen, "A Survey of Thresholding Techniques," *Computer Vision, Graphics and Image Processing*, 41, pages 233-260, 1988.

[86] Ramin Samadani, "Adaptive snakes: control of damping and material parameters," *Geometric Methods in Computer Vision*, 1570, pages 202-213, 1991.

[87] G. Sapiro and A. Tannenbaum, "Area and length preserving geometric invariant scale-spaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(1), 1993.

[88] G. Sapiro and A. Tannenbaum, "Affine invariant scale-space," *Int. J. Computer Vision* 11(1), 1993.

[89] G. Sapiro, "Vector-Valued Active Contours," *Proc. CVPR*, pages 680-685, 1996.

[90] Pengcheng Shi, Glynn Robinson, and James Duncan, "Myocardial Motion and Function Assessment Using 4D Images," *SPIE*, 2359.

[91] Jack Sklansky, "On the Hough Technique for Curve Detection, *IEEE Transactions on Computers*, 27(10), pages 923-926, October 1978.

[92] L.H. Staib, *Parametrically Deformable Contour Models for Image Analysis*, PhD thesis, Yale Univeristy, 1990.

[93] Lawrence H. Staib and James S. Duncan, "Boundary Finding with Parametrically Deformable Models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(11), pages 1061-1075, 1992.

[94] R. Szeliski, D. Tonnesen, and D. Terzopoulos, "Modeling surfaces of arbitrary topology with dynamic particles," *Proc. CVPR*, pages 82-87, 1993.

[95] Saeid Tehrani, Terry E. Weymouth, and G. B. John Mancini, "Knowledge-guided Left Ventricular Boundary Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 342-347, 1989.

[96] D. Terzopoulos, A. Witkin, and M. Kass, "Constraints on deform-able models: Recovering shape and non-rigid motion," *Artificial Intelligence*, 36(1), pages 91-123, 1988.

[97] Y.F. Wang and J.F. Wang, "Surface Reconstruction using Deformable Models with Interior and Boundary Constraints," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(5), pages 572-579, 1992.

[98] B. Widrow, "The Rubber Mask Technique – i and ii," *Pattern Recognition*, 5, pages 175-211, 1973.

[99] Donna J. Williams and Mubarak Shah, "A Fast Algorithm for Active Contours and Curvature Estimation," *CVGIP: Image Understanding*, 55(1), pages 14-26, 1992.

[100] Andrew P. Witkin and Jay M. Tenenbaum, "What is Perceptual Organization for?" *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1023-1026, 1983.

[101] A.P. Witkin, "Scale-space Filtering," *Proceedings of International Joint Conference on Artificial Intelligence*, pages 1019-1022, 1983.

[102] A.C.K. Wong and P.K. Sahoo, "A Grey Level Threshold Selection Method on Maximum Entropy Principle," *IEEE Transactions on Systems, Man and Cybernectics*, 19, pages 866-871, 1989.

[103] S.D. Yanowitz and A.M. Bruckstein, "A New Method for Image Segmentation," *Computer Vision, Graphics and Image Processing*, 46, pages 82-95, 1989.

[104] A. Yuille and T. Poggio, "Scaling Theorems for Zero Crossings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8, 1986.

[105] A. Yuille, P.W. Hallinan, and D.S. Cohen, "Feature Extraction in Face Using Deformable Templates," *International Journal of Computer Vision*, 8, pages 99-111, 1992.

[106] A. Yuille and T. Poggio, "Scaling Theorems for Zero Crossings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8, 1996.

[107] A. Yuille, D.S. Cohen, and P.W. Hallinan, "Feacture Extractions in Faces Using Deformable Templates," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 104-109, 1989.

[108] S.C. Zhu, T.S. Lee, and A.L. Yuille, "Region Competition: Unifying Snakes, Region Growing, Energy/Bayes/MDL for Multi-band Image Segmentation," *Proc. 5th ICCV*, pages 416-423, 1995.