Part II

Probabilistic Atlases of Human Neuroanatomy: Methods and Applications

Chapter 4

Surface Deformation Review

J. David MacDonald

Montreal Neurological Institute, McGill University
McConnell Brain Imaging Centre
3801, University St., Montreal, Canada H3A 2B4

There continues to be considerable research relating to processing of MR images. The following is a discussion of various techniques which are relevant to the problem at hand. Rather than being competing methods, many of these methods complement each other and typically, several techniques are used in combination to provide an overall solution. The methods discussed are divided into two classes. The following section describes those methods that can be classified as **volumetric pre-processing**, and the subsequent sections deal with techniques for image segmentation. The applicability of each method as well as its relative advantages and disadvantages are explored.

4.1 Volumetric Pre-Processing

Many general image processing techniques are relevant to the three dimensional images produced by MR scanners. Three methods of pre-processing the MR images for subsequent image analysis are described here: image registration, tissue classification, and RF inhomogeneity correction.

4.1.1 MR Image Registration

There are a multitude of methods proposed to handle the task of **image registration** or **spatial normalization**, which is the process of spatially transforming an image to be aligned within some target coordinate system. The goal of MR registration is to provide a standardized frame of reference for subsequent analysis. For processing of human brain MR images, the initial step is typically to apply a simple rigid geometric transformation consisting of three dimensional scaling, rotation, and translation, to align the dataset into a standard coordinate system, often termed a **stereotaxic** coordinate system. This can aid analysis by introducing a certain amount of predictability into the positions of various anatomical structures. This image registration task is addressed by both manual and automatic methods. Manual methods [LHH+91, EMN+92] involve locating corresponding coordinates (either based on anatomy or on artificially introduced objects (**fiducials**)) in both the given image and a standard target image to which all images are mapped. The set of pairs of three-dimensional coordinates is used to define a transformation between positions in the two images. The given image is transformed by this mapping to produce an image that is in registration with the target.

The reliance on human observers for the definition of corresponding points motivates a quest for more automatic methods of image registration. As a result, there are several automated methods which attempt to match image features, such as intensity values or gradients, across a pair of images [Bcc89, CNPE94, KGC+89, RTL+93, TJP+93]. An initial guess for the transformation is given, either by manual intervention, or by an automated procedure such as **principal axes analysis** [RTL+93]. The similarity between the image to be transformed and the target image is measured and the parameters of the transformation are adjusted to attempt to maximize this measure. The measure used is typically a function such as the cross-correlation of intensity values in the original image with the values in the transformed position in the target image. These methods are being used successfully to factor out the gross position, orientation, and size differences of various individual brains, and can be thought of as a very coarse means of segmentation.

The work of Talairach and Tournoux [TT88] in manually defining a standardized coordinate system based on identifiable anatomical landmarks is adapted into a digital framework by various methods. The method of Collins et al [CNPE94] involves the definition of a stereotaxic coordinate system, based on the atlas of Talairach and Tournoux. Using manual methods of landmark identification, the reference model image is placed into stereotaxic space. An automatic three dimensional image registration algorithm can then be applied to register any target image with the model image. Figure 4.1 depicts the registration of an image into stereotaxic space from **native** space, the intrinsic coordinate system of the MR scanner. Subsequent neuroanatomical analysis of the target image is greatly facilitated by its standardized orientation and positioning within the frame of reference defined by the stereotaxic coordinate system.

4.1.2 Tissue Classification

The human brain has several different tissue compositions, the principal ones being gray matter, white matter, and cerebral spinal fluid (CSF). Depending on the imaging parameters of the MR acquisition, these different tissue types give varying signal responses, which are used to discern tissue type distribution in the image. By using several MR images of a single brain, including the T1-, T2-, and PD-weighted images, a three dimensional feature vector is derived for each voxel, consisting of the intensity values for the voxel in the three images. Pattern classifiers such as neural networks, nearest neighbour methods, and Gaussian-modeled linear discriminants [1] are invoked on training data where the tissue types are known, in attempts to create an accurate classifier for arbitrary datasets which are acquired with similar imaging parameters to the training data [ZDM93, CCR+93, KCS+92, LHH+91, RLW93]. This constitutes a low-level segmentation of the brain, which does not necessarily correspond to an anatomical segmentation of the brain, since different anatomical structures may have the same tissue type.

Figure 4.2 illustrates the results of a typical tissue classification algorithm generating an image with four classes: gray matter, white matter, CSF, and background. Although tissue classification methods by themselves are not sufficient for segmentation of anatomical objects, it may be possible to use the tissue class information to assist other methods of segmentation. In particular, tissue classification will be revisited as an edge detection step for MR images in the next chapter.

[1] [Nil90] provides a good introduction to these and other pattern classifiers

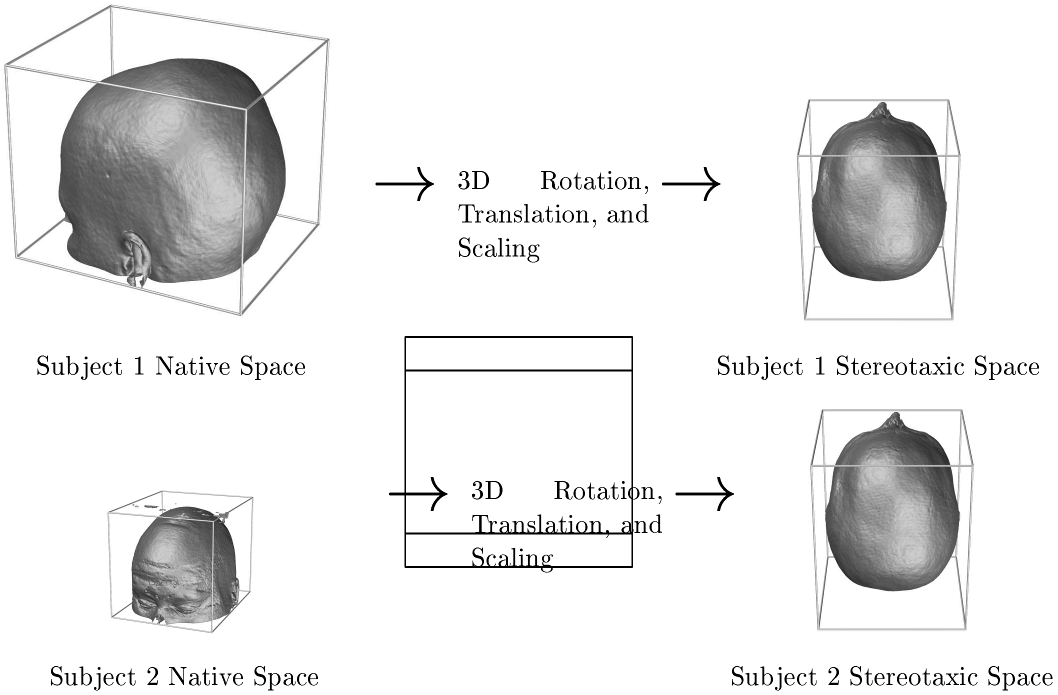

Figure 4.1: Transformation from native space to stereotaxic space.

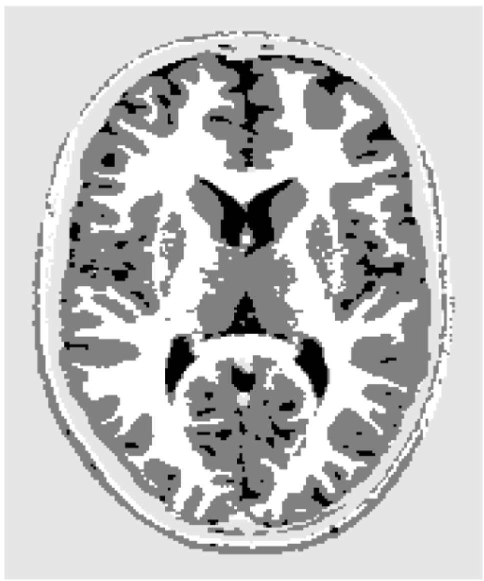

Figure 4.2: One slice through a 4-class volume (white matter is white, gray matter is gray, CSF is black, background is off-white).

4.1.3 RF Correction

The unique nature of the RF image inhomogeneity distortion that plagues MR imaging invites specialized methods to reduce or remove this artifact from images (Fig. 4.3). These methods can be loosely classified into two categories, where the principal difference is whether or not the RF correction is performed in conjunction with tissue classification. When used with tissue classification, the typical method attempts to estimate two unknowns: the tissue classes of the voxels and the spatially varying RF field over the volume. This is accomplished by fixing the estimate of one of the unknowns in order to compute an estimate for the second unknown, which is then used to estimate the first unknown. These two steps are repeated until the sequence converges. A popular method which uses this methodology is called the **Expectation-Maximization** algorithm [WIGKJ94].

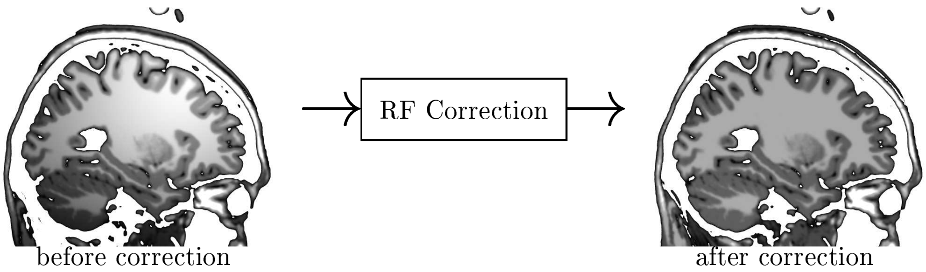

Figure 4.3: RF image inhomogeneity correction reduces the slow-varying intensity gradient across the image.

The other general class of RF correction methods does not explicitly involve tissue classification, but instead uses the intensity histogram to achieve a similar result. By comparing local histograms of image intensity in different spatial locations, the shift in the histogram is identified, and thus an estimate of the RF inhomogeneity field can be derived. One example of this method [SZE97] will be revisited in the next chapter.

Having examined various methods of pre-processing images, a survey of image segmentation algorithms is presented. Any method which divides images into distinct regions or extracts models of specific regions of the image may be considered an image segmentation algorithm. Some important differences among the various methods described in the following sections are the representation of the segmented objects, the level of user intervention required, and the level of model information used to constrain the process.

4.2 Contours

Many methods of image segmentation are based upon choosing a threshold which corresponds to the image intensity at the boundaries between objects in the image, or using gradient-based edge detectors to label particular regions of the image as boundary. Boundary points detected on two-dimensional slices of the image are then connected together into a curve or contour. This partitions the image into a set of connected components (Fig. 4.4), thereby achieving a segmentation of the image. The advantage of contours is that the problem of three-dimensional segmentation is initially converted to a more tractable two-dimensional problem. Often two objects which are connected in the three-dimensional space are not connected on many of the slices through the volume. Even if they are connected, manual segmentation techniques and morphological operators are simpler and more successful when applied independently to each slice, due to the ability of conventional two-dimensional technology to more easily display image slices than entire volumes. Two dimensional image contouring methods applied to three dimensional images are prevalent [CL88, EPO91, FKU77, GD82, MSS92] and generally attempt to construct a three dimensional surface from stacks of contours extracted from parallel slices. The essence of the task is find correspondences between positions on contours of neighbouring slices. The difficulty is that when these methods are applied to complex shaped objects, it may be very difficult or impossible to decide how to connect points on contours to those on neighbouring slices. Because this often occurs when a contour splits into two on the next slice, this is usually called the **branching problem**, and is a significant factor in limiting the use of contour methods in higher dimensional applications. Methods that attempt to solve these problems can involve very complex algorithms [EPO91]. In actuality, the initial advantage of reducing the problem from three to two dimensions is offset by the fact that the complexity of the higher dimensionality problem has not been eliminated, but rather delayed until the final connection phase. However, because of the wealth of published experience in manual and automatic methods of contouring datasets, contouring is a segmentation tool that should not be ignored, and can prove useful for various aspects of the segmentation problem, including preprocessing data and building models for other segmentation methods.

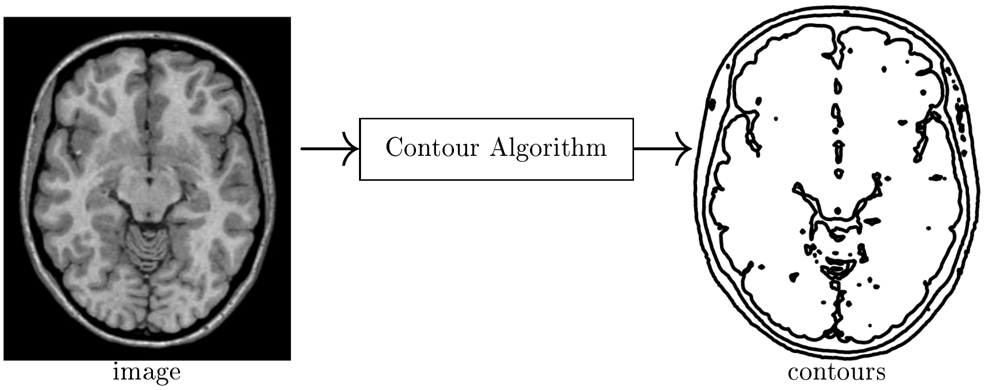

Figure 4.4: 2D image contouring.

4.3 Isosurfaces (3D Contours)

The inefficiency of contours in dealing with three dimensional data has led many researchers to explicitly extend the idea of contouring to three dimensions. The obvious approach is to construct surfaces of the boundaries of the objects, again using an image threshold. Since the surfaces correspond to sets of points which are of equal values, they are called **isosurfaces**. Isosurface algorithms [Blo88, BW90, CLL$^+$88, HW90, KCHN91, LC87, WMW86] are the three dimensional analog of two dimensional contouring and most isosurface methods are implemented in a very similar fashion. Typically, the volume is tessellated into small, simple geometric objects, and the thresholded surface is approximated in each, building up a set of connected polygons. The most cited example is the method of Lorensen and Cline [LC87] which uses rectilinear boxes as the tessellating object, leading them to term the method **marching cubes**, in reference to the algorithm's processing of the cubes one by one. By examining the sign of the difference between the desired threshold and each of the eight corner nodes of a single box, a set of one to six triangles is created to approximate the surface through the box. Figure 4.5 illustrates one situation where four nodes are below the threshold and four are above. In this case, the approximation to the surface is a set of four triangles, whose vertices are found by linear interpolation along the relevant box edges. The algorithm simply examines the eight vertices of each box in the dataset, constructing a small number of triangles for each one, and results in a connected surface for each connected object in the volume, a simple example of which appears in Fig. 4.6. Interestingly, although the marching cubes method is by far the most commonly referenced method, it has the serious problem of generating surfaces that may have holes, where the triangle edges do not join well across adjacent voxels in some cases. A similar method published one year earlier by Wyvill et al [WMW86] avoids this problem and produces a correct surface. More recent methods usually do not suffer from this problem either, such as the method of using tetrahedral tessellations by Hall and Warren [HW90] or a similar one by Payne and Toga [PT90]. However, despite the limitations of the method of Lorensen and Cline, the term *marching cubes* has become synonymous with isosurface construction algorithms.

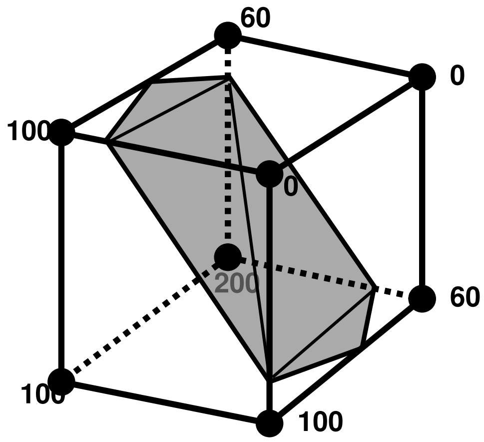

Figure 4.5: Example of marching cubes algorithm approximating the isosurface of value 80.

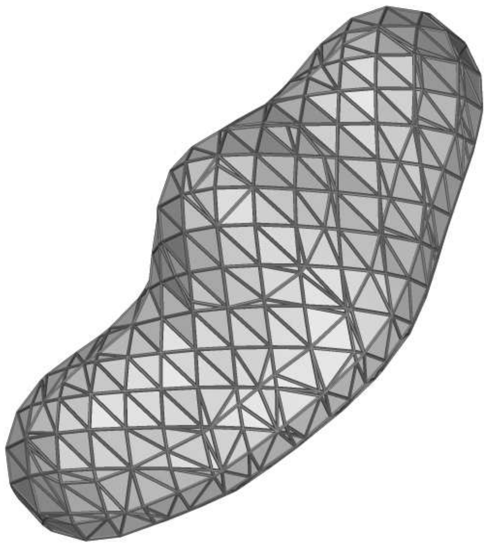

Figure 4.6: Triangulated surface generated by the marching cubes algorithm.

Although the simplicity and speed of contouring and isosurface algorithms is quite attractive, their use for surface segmentation is limited by the implicit assumption that objects that are distinct in reality appear as disconnected regions within the image. However, this assumption is often invalid, such as when objects are close together, similar in intensity, or affected by noise in the image. In these cases, an isosurface algorithm may create a single geometric object which corresponds to several distinct objects, or several distinct geometric objects which correspond to a single object, both of which may be considered incorrect segmentations of the image data. Thus, isosurface algorithms are generally used in conjunction with other manual and automatic segmentation algorithms to produce a surface representation of a previously segmented image. In addition, the lack of topological constraints in these algorithms means that holes in the data due to partial volume and other effects result in surfaces with similar potentially incorrect topological characteristics.

4.4 Morphological Operators

One method to partition the volume into separate objects is to use **morphological operators**. Morphological operators consist of simple transformations on the volume which affect the local connectivity of voxels. A good introduction is presented in [Ser82]. An image is first thresholded or classified in some way to create a binary image, where each voxel indicates whether or not an object may exist at that location. A **fill** operation labels each voxel within a homologous region in the image. An **erode** operation on a binary image changes all object voxels which have a non-object voxel in its neighbourhood (within a certain small number of voxels, defined by a specific kernel) to non-object status. This results in shrinking object regions in the image. A **dilate** operation performs the opposite operation and results in expanding object regions. A **close** operation consists of a dilate operation, followed by an erode operation, and effectively smoothes the boundaries of objects, filling in small holes. An **open** operation consists of an erode operation, followed by a dilate operation, and results in disconnecting regions which started out as very tenuously connected. Results of these operations on two dimensional images are presented in Fig. 4.7. Although the most obvious effects occur in loosely connected regions, it is important to note that all boundaries in the image may potentially be perturbed. Application of these types of operations on a thresholded volume can be used effectively in segmentation to break up an image into meaningful subregions. However, the choice of the sequence of operations and the relevant neighbourhood size will vary depending on the application and the particular characteristics of the dataset and the object being segmented. Morphological operators are thus best applied when the input data is sufficiently close to being disconnected into the desired segmentation. Unfortunately, this is not usually the case in the realm of MRI, and morphological operators have had only limited success for automated neuroanatomical segmentation, often because the sequence of morphological operations has to be empirically determined for each particular dataset. The limitations of morphological operators for automatic MRI segmentation are expressed by Hohne and Hanson: "Application of morphology operators depends strongly on intuition (at least in the case of complicated objects), which suggests that they would be used most effectively in an interactive mode" [HH92].

Morphological operators can be used to fill in holes due to partial volume effects and noise. The method of Dale and Sereno [DS93] uses a tissue classifier to define the volume of white matter voxels in an MRI volume. Then a sequence of three dimensional fill operations is performed to remove holes from the white matter. This guarantees that the surface of the resulting white matter volume is isomorphic to a sphere. However, the addition of non-white matter voxels to the list of white matter voxels calls into question the accuracy of the resulting surface.

4.5 Data-Driven Methods Versus Model-Driven Methods

While the methods described have been used successfully in a semi-automatic context, none of them are powerful enough to provide automatic segmentation of the cortical surface from MR images. The essential weakness of most of these algorithms is that they are almost purely **data-driven**; the only information available to the process is the data itself. Higher levels of information are available only indirectly, usually as input from an experienced human observer. As a result, researchers are investigating

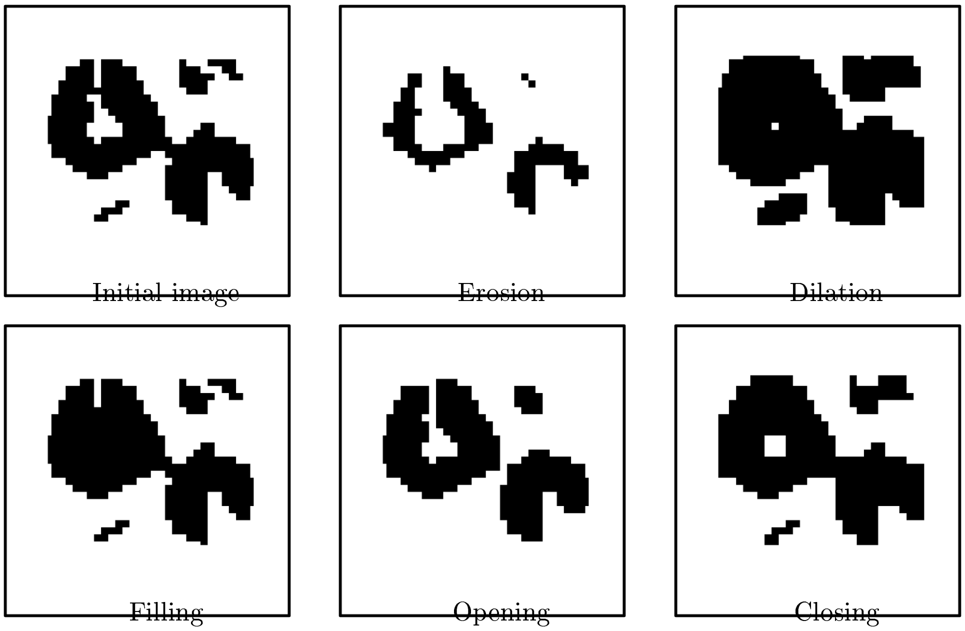

Figure 4.7: Results of five different morphological operators on an image.

methods of incorporating *a priori* knowledge (application-specific information), into the process of image segmentation, in hopes to develop general-purpose fully automatic algorithms. Many of these latter techniques share the concept of a model which is matched to the image, providing a constraint on the segmentation process in a **model-driven** fashion.

4.5.1 Three Dimensional Image Registration as Model-Based Segmentation

Image registration methods that involve local transformations, in addition to global affine ones, provide more of a one-to-one correspondence between positions in images being registered, which makes them applicable to the domain of model-based segmentation. Using higher order transformations, image registration methods can achieve image segmentation and matching [CPDE92, Bcc89]. The essential idea is to register an arbitrary MR image to a standard MR image, the model, which has been labeled in some fashion. The registration provides a mapping from each position and associated classification label in the model to one in a target image, thereby attaching a label to each position in the target image, and effectively segmenting it. This provides a very flexible method of segmentation, in that any individual image may be used as the model image. Furthermore, for each model, several sets of segmented labels may be used, so one registration of an image to the model may provide a suite of several complementary segmentations of the image. Critical to the effectiveness of these methods is the accuracy of the matching between similar structures. In general, these methods work well for large objects such as the four major lobes of the cerebral cortex and some of the regularly shaped anatomical structures such as the thalamus and putamen, but fail to robustly match many of the other structures of the brain, particularly cortical features, due to their inherent variability in size, shape, and topology across subjects. In addition, these methods generally restrict the allowable transformations to continuous deformations of three dimensional space, which limits the ability to represent discontinuities or topological changes between two anatomical images.

4.5.2 Deformable Models

A very promising method of model-based image analysis involves the use of deformable models. A curve, surface, or volume that approximates the target object is deformed to fit an image volume. One of the earliest and most referenced deformable model techniques applied to image segmentation is the **Snakes** method of Kass et al [KWT88]. A snake is a spline curve which changes its configuration to minimize its energy, composed of two opposing terms: an attraction to image features such as edges, and a set of internal forces which constrain shape and position. An initial configuration is subjected to the forces and deformed until it reaches equilibrium where its energy is minimized. A simple formulation of the energy equation to be minimized is

$$E_{snake} = \int_0^1 \alpha(s)|v_s(s)|^2 \ + \ \beta(s)|v_{ss}(s)|^2 \ + \ \gamma(s)I(v(s));$$

where the spline curve is defined by

$$v(s) = (x(s), y(s)), \ 0 \leq x \leq 1,$$

with first and second derivatives, $v_s(s)$ and $v_{ss}(s)$, respectively, and the image data is represented by the function $I(x, y)$. The functions, $\alpha(s)$, $\beta(s)$, and $\gamma(s)$, represent weights that control the relative effects of each of the three terms. A snake subjected to this equation attempts to move itself to areas of minimum or maximum image intensity, depending on the sign of $\gamma(s)$, subject to the constraints imposed by the first two terms. The first term, involving the gradient of the spline curve, $v_s(s)$, makes the snake act like an infinitely thin membrane preferring not to stretch or compress in length. The second term, involving the second derivative, v_{ss}, makes it act like a thin plate, preferring not to bend from the model configuration. The weighting functions, $\alpha(s)$ and $\beta(s)$, control the relative strengths of these two constraints. The ultimate effect of these two terms is to control the amount of stretching/compression and bending of the curve away from a model curve shape. The essential idea is that in areas where the data is ambiguous or ill-defined, the model shape will be imposed upon the snake. In other areas where the image edges are less ambiguous, the snake will be driven by the data. The choice of the weighting functions must be chosen interactively or empirically by the user, a drawback shared with most of the deformable methods. Figure 4.8 portrays a simple image and resulting deformation of a snake. The stretching and bending energy cost of the snake has prevented it from attaching to the small noise-like dot in the upper right of the image. The sensitivity of the snake to noise and small features is dependent on the choice of weights for the regularization terms of stretching and bending.

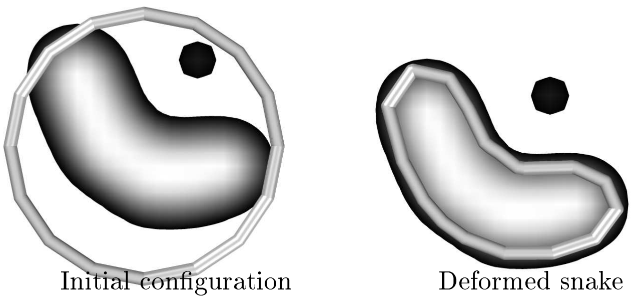

Figure 4.8: Active contour algorithm with snake initialized to be a circle (grey).

Kass et al describe several methods of representing the energy term, including alternate formulations of the image term, $I(v(s))$, involving first and second order edge detection terms, as well as additional constraints such as geometrical proximity forces. A multi-scale approach is suggested, to lessen the sensitivity of the minimization process to local minima, starting with a blurred image and slowly decreasing the blur while the energy of the snake is being minimized. The use of a blurred image is important to increase the range of attraction of the spline to image edges, by increasing the effective width of edges. The actual implementation involves using finite differences to approximate derivatives on a piecewise spline, and the minimization is achieved using implicit and explicit Euler steps in an iterative technique, with order $O(n)$ computations per step, where n is the number of parameters in the spline. Later work by the authors and others have extended these concepts to three dimensions by deforming surfaces to fit two and three dimensional images. The computational complexity of the original Snakes method can be considerable in three dimensions at high resolutions, as a large matrix inverse must be computed at each step. This can be quite expensive, especially when the image forces are quite strong, in which case Kass et al indicate that "the explicit Euler steps of the external forces will require much smaller step sizes" [KWT88]. The many related methods inspired by the ground-breaking Snakes paper use a wide variety of cost functions and minimization techniques, in attempts to optimize the use of active contour models to particular tasks [TWK88, TWK87, CC93, SMG+93b, CHTH93, GA93, HEM92].

The success of the Snakes method as a general purpose model-based image segmentation tool has resulted in its application to a wide variety of domains. The complexity of segmenting neuroanatomical structures from high resolution medical images has motivated creation of many algorithms which can be conceived of as variations on the Snakes method. A survey of some of these most relevant to medical image analysis are presented here, with strengths and weaknesses outlined. A more comprehensive survey may be found in [MT96].

4.5.3 Cohen, Cohen, and Ayache

The authors Cohen, Cohen, and Ayache have several papers describing variations of the Snakes method for segmentation of medical images [CAC91, CCA91, CCA92, CC90, CC93]. The early work of Cohen and Cohen explores the use of a finite element implementation of the Snakes algorithm for curves on slices of MR. Later work adapts the Snakes techniques to three dimensional image segmentation [CCA92]. Again a finite element approach is substituted for the finite difference approach of the original Snakes method, resulting in an improved coverage of the domain of the surface without increasing the number of node points. Their method involves user input of a fairly low resolution initial guess, and produces low resolution surface representations. Use of an automatically generated initial guess would presumably work equally well, and eliminate the need for user intervention. However, its use in an automatic context and applicability to recognizing the highly convoluted and detailed surfaces of the human cortex has not been explored.

4.5.4 Dale and Sereno

Dale and Sereno [DS93] use a simpler, geometric approach to segmenting the human cerebral cortex. MRI data is initially classified into gray and white matter, using a tissue classification algorithm. Using some simple morphological operations, a white matter volume is created which has no topological holes, and whose boundary consists of a two dimensional manifold (isomorphic to a sphere). The boundary of this volume represents the interface between the gray and white matter, and is therefore a good initial guess for locating the gray-CSF boundary, which is the objective of the method. The polyhedral surface of the white matter is then expanded by moving vertices towards the gray-CSF boundary, while also constraining each vertex to be attracted to the centroid of its neighbours. The result is a polyhedral mesh approximating the cerebral cortical surface. While this provides a very high resolution description of the cortical surface with efficient use of computer resources, it has the problem of potentially creating non-simple (self-intersecting surfaces), and does not address the partial volume problem. However, the idea of using the gray-white boundary to guide the search for the cortical surface boundary is an important one that will be revisited later in this dissertation.

4.5.5 Davatzikos and Bryan

Davatzikos and Bryan [DB95] present an active contour method which models the cortical gray matter as a finite thickness sheet. The sheet has a two dimensional parameterization, making it suitable for subsequent morphometric analysis. Rather than parameterizing the complete folded cortical surface, the method parameterizes the outer boundary of the cortex, only entering the upper portion of each sulcus. Location of the depths of the sulcus is performed as a second step, where a curve is initialized at the top of each sulcus and pushed down into the depths by a method similar to the Snakes algorithm. While this method is novel in its attempts to locate the depths of the sulci, the total surface of the cortex is not contained in a single model. Coordinating the two models used, that is, the outer surface model and the set of deep curves, into a single model which faithfully represents the true folded cortex is a nontrivial task.

4.5.6 Sandor and Leahy

Sandor and Leahy [SL97] propose a method to automatically locate the cerebral cortical surface from MR images, and to impose a labeling on the resulting sulci. The authors rely on an edge detection method to define boundaries, followed by a series of morphological operations to create both a smooth voxel representation of the outer limits of the cortex, and a set of voxels corresponding to holes in the volume inside this hull. The assumption is that the holes represent either sulci, noise, or interior brain structures. By choosing the holes that are connected to the outside of the brain, the set of sulci are distinguished from the other two types of holes. A smooth atlas of the cerebral cortex, labeled with points on the extremities of sulci, is warped by a three dimensional Snake method to fit the smooth brain. The labels are then transferred to the smooth brain and to the set of voxels previously labeled as sulci. This method has been demonstrated to label the principal sulci almost completely automatically, but has a few disadvantages in the context of the problem addressed in this thesis. The surface representation does not go very deep into the sulcus, because the authors have chosen to model only the exterior portion of the cortex, stating that “without user interaction deformable models can not be guaranteed to converge to complex and convoluted image features”. There are several morphological operators involved in smoothing the volume, which raises the question of how much error is introduced into local boundary positions. Finally, only the parts of sulci that are connected to the exterior are actually labeled as sulci, thus relying on the success of the morphological operator in opening up the entire depths of each sulcus, without modifying the data so much in other areas as to make the results incorrect.

4.5.7 Staib

The work of Staib et al [SD92b, SL, SD92a] uses deformable Fourier surface models composed of sinusoidal basis functions. Different topologies of surface can be modeled with a variable number of parameters, depending upon the resolution desired. Similar to the previous methods, a cost function is devised that is integrated over the two dimensional parameter space of the surface. The data is preprocessed with Gaussian smoothing to reduce the effects of noise, and filtered with a 3 x 3 x 3 Zucker-Hummel operator to create a smooth boundary. Gradient ascent is used to optimize the solution, starting with a rough initial guess. While this method produces models which facilitate shape analysis by encapsulating gross shape into a few parameters, application to the very complex problem of capturing the total cortical surface and deep sulci has not been demonstrated. Correcting deep sulci for partial volume effects and preventing non-simple surfaces is not addressed.

4.5.8 Level Sets

Mallaidi, Sethian, and Vemuri [MSV95] present a novel version of the active contours deformable method. The principal contribution is the re-parameterization of the deforming curve or surface as a level set of a higher dimensionality function. Whereas the original snakes formulation consisted of the points on a parametrically defined curve, $(x(u), y(u))$, the level set method represents the curve by the set of zero points of a two dimensional function, $F(x, y) = 0$. An initial function F is defined, and evolved over time by solving a partial differential equation. The advantage of this formulation is its ability to handle unknown image topologies. The evolving surface automatically splits into multiple components or combines several components into one, depending on the topology of the objects in the image. This provides an efficient solution to the problem of self-intersection avoidance, since level set curves cannot cross, although they can touch. This ability of the algorithm to change the models topology depending on the image data can be an attractive feature in some image segmentation tasks. However, in cases where a specific topology of the object is desired, such as with the cortical surface, this method suffers from the lack of a mechanism to force the algorithm to maintain a particular topology.

4.5.9 Modal Analysis Methods

A considerable amount of work by Pentland has broad application to many vision problems [PHS93, PH91, PS91, PW89, Pen90, Pen89, Pen88, WP92]. These methods generally use a simple object such as a **superquadric** surface to provide concise three dimensional representations derived from typically two dimensional image components. Superquadrics are essentially ellipsoids with a few extra parameters to provide pinched and tapered shapes. Local shape differences are provided by the augmenting of the superquadric models with local offset deformations. Although generally applicable to a wide range of vision problems, the method is suited best for identifying smooth part models from two dimensional images or three dimensional range data, and the segmentation of complex shapes in three dimensional medical images has not been demonstrated. However, related work by Pentland and colleagues [PW89, SP93] in parameterizing shape and measuring shape differences using **modal analysis** has been applied in a medical imaging context. The work of Nastar and Ayache [NA93a, NA93b], uses modal analysis to represent and track two and three dimensional objects that deform over time. Modal analysis uses traditional mechanical engineering techniques to break down shapes into sequences of successively finer modes of vibration. The non-rigid lower order modes can be used to perform matching of similar structures with a very succinct parameter space. Modal analysis can be a powerful tool for subsequent analysis of surfaces segmented from MR images, as well as a technique for matching. However, its applicability to segmenting highly convoluted cortical structures requires more investigation.

4.6 Why Conventional Deformable Surfaces are Insufficient

The current methods of deforming surfaces to fit three dimensional images just described fail to adequately address the problem of creating topologically consistent, simple cortical surfaces with partial-volume corrected deep sulci in three respects,

- possibility of self intersection,
- sensitivity to partial volume effects, and
- inability to accurately represent the total cortical surface.

Firstly, these methods rely on the model shape constraints (stretching and bending) for a “regularization” effect. That is, by keeping the surface more resistant to bending and stretching, it is less likely to wrap over on itself and become self-intersecting. There are two problems with this, one being the fact that this is only an encouragement not to self-intersect, not a guarantee. The second problem is that, as more regularization is imposed, the surface is less likely to deform to fit the particular dataset. A second problem with the existing active methods is that the partial volume effect is not fully addressed. A single surface is used to try to connect together detected edge features essentially using a simple connectivity constraint, based on a continuous surface whose stretching and bending is constrained. However, in order to account for places in the image where gyral boundaries are blurred by under-sampling, it would seem that more sophisticated models of the possible configurations of human cerebral cortex are required, in order to infer folded gyral configurations in these areas. A lesser, but not insignificant, problem is the computational tractability of applying existing methods to a surface as complicated as the human cortical surface. Of the methods surveyed, only that of Dale and Sereno makes an attempt to find the entire cortical surface in a high-resolution representation. However, this method provides no enforcement of self-intersection avoidance or explicit strategy to circumvent partial volume effects.

4.7 Numerical Minimization Methods

Many of the algorithms covered so far rely in one form or another on minimization of a multi-dimensional function. Minimization is simply the process of finding a point in the parameter space for which the function value is less than or equal to the function value for all other points in the parameter space. The ability to efficiently find a minimum is critical to many of the methods, and therefore, a survey of some numerical techniques for minimization of functions is presented. A reasonable starting point for practical implementation of standard numerical algorithms is the book, *Numerical Recipes* [PFTV88]. The practical differences between the various algorithms involve whether or not derivatives of the function must be calculated, the amount of storage involved, and the sensitivity of the algorithm to local minima.

4.7.1 Gradient Descent

The most basic method of function minimization consists of simply walking downhill from the current position. This is usually termed **gradient descent** because it involves evaluating the first derivative of the objective function, and stepping in the negative direction until the directional minimum (minimum along a line) is found. Performing this in an iterative fashion will eventually terminate at a point in the domain where the gradient is zero. However, this point may not necessarily be the global minimum, but might be a local minimum, or some other feature, such as a saddle point. It is difficult for any algorithm to avoid local minima, but the feature of being trapped in saddle points and other singularities is a significant disadvantage of the gradient descent method. In addition, depending on the curvature of the function being minimized, the gradient descent method can have a slower convergence rate than other methods.

4.7.2 Conjugate Gradients

The method of conjugate gradients addresses the main problem of more naive approaches that simply follow gradients, that of slow convergence and getting trapped in singularities such as saddle points. Conjugate gradient methods rely on a simple formula to compute search directions based on linear combinations of gradients, but in such a way that the successive search directions are roughly mutually orthogonal. This means that minimizing in a particular conjugate direction does not "undo" the minimization achieved in the previously searched directions. In the case of a quadratic function, this method is guaranteed to find the single global minimum in a number of iterations equal to the size of the parameter space. For more complex functions, it has been found to behave very well, although it can still be trapped in local minima. One particularly attractive feature of many conjugate gradient methods is that the storage involved is linear in the dimensionality of the function. A practical conjugate gradient algorithm may be found in [PFTV88].

4.7.3 Golden Section Search

The Golden section algorithm [PFTV88] is a straightforward method of finding a minimum of a one-dimensional function, and therefore, a method of finding the minimum of a multi-dimensional function along a line. The Golden section search is an iterative algorithm involving shrinking the interval of search until a minimum is found to within some specified tolerance. It is essentially the optimization analog of the binary search method of root finding. The Golden section search is initialized with three points along the line, where the the function value at the inner point is less than that of the outer two. The outer two points define the interval of search for the minimum. An iteration consists of computing the function value at a fourth point within the interval, and replacing one of the two outer points with the fourth point, thus producing a new set of three points with a smaller interval. A method of computing the fourth point based on the outer two and the Golden ratio (the value 0.618034), results in a method where the size of each successive interval is the Golden ratio times the previous. The advantages of this method are that is does not require computation of function derivatives and has a predictable convergence rate, exponentially decreasing the size of the interval. The golden section search is often used as the line minimization component of gradient descent and conjugate gradient methods.

4.7.4 Simplex

The **simplex** [PFTV88] method of function minimization (not to be confused with the simplex method of linear programming) is a geometric approach to finding the minimum of a function. The most attractive feature of the algorithm is that no function derivatives are required to be computed. A simplex is the convex hull of $d+1$ points in d dimensions, where d is the number of dimensions of the domain of the function to be minimized. An iteration of the simplex involves trying one of several different geometric contortions on the simplex to reduce the cost of its maximal vertex. These contortions involve contracting and expanding edges, and the resulting behaviour of the simplex has inspired the algorithm to be referred to as **amoeba**. However, the requirement of quadratic storage space ($d+1$ points of d values each) is prohibitive when dealing with functions where d is very large.

4.7.5 Simulated Annealing

All the numerical minimization techniques described thus far have the trait of searching locally from an initial position, and typically find a local minimum which is very dependent on the initial position. The method of **simulated annealing** introduces randomness into the search, in an attempt to step over local minima and increase the chance of finding the global minimum. Kirkpatrick et al [KGV83] present simulated annealing as an approximation to the process of cooling a physical material from a liquid to a solid. They apply the method to several optimization tasks, including the Traveling Salesman problem. An initial parameter configuration is randomly modified in an iterative fashion. At first the perturbations of the parameters are quite random, but as the iterations progress, the likelihood of a particular change in configuration is increasingly forced to be proportional to the decrease in function value attributable to the change. The function being minimized is analogous to the energy state of a physical substance, and the increasing preference of the parameters to move towards lower function values is analogous to the increase in probability of the substance to move to a lower energy state as the temperature is lowered. This effectively results in a random search at first, smoothly changing into an approximate local gradient-following search near the end. Although the method of simulated annealing has been used successfully on a number of problems, it has the drawback that the temperature reduction "schedule" must be empirically determined for each problem, so that the temperature is reduced slowly enough to ensure a good chance of finding the global minimum, but fast enough to provide a tractable solution. As the authors say, the choice of this schedule requires "insight into the problem being solved and may not be obvious".

4.7.6 Genetic Algorithms

Another more recently proposed class of optimization methods are termed **genetic algorithms**, whose stochastic nature is closely related to simulated annealing, and likewise has an analogy to a real world process. The principal distinction from simulated annealing methods is the method of generating randomized positions in parameter space from previous ones. Genetic algorithms emulate biological evolution, which attempts to optimize the performance of a species using genetic combinations of existing members to create new ones. A "population" of configurations of the parameters is created as a set of initial guesses to the optimum value of the function. The parameters of the objective function are treated as a sequence of genes, and pairs of members of a population breed by combining their genes into a new sequence using simple rules of gene splicing. Borrowing from nature's rule of survival of the fittest, members of the population which have a lower function value are more likely to thrive and enter into reproductive liaisons with other members. The result is that the population evolves into a set of members which optimize the objective function, and careful selection of the reproductive process increases the likelihood of finding the global minimum. Similar to simulated annealing, this method has the drawback of requiring a reproduction strategy for combining estimates into new estimates, the formulation of which may not be obvious.

4.8 Self-intersection and Proximity

One of the main limitations of contemporary surface segmentation research is that creation of non-simple configurations is not precluded: there is no guarantee that the resulting object does not intersect itself. Detection and prevention of non-simple surfaces is directly related to intersection and proximity testing, an area of extensive interest in computational geometry. Applications include collision detection and path planning in robot movement, VLSI circuit design, and animation. An introduction to many of the data structures and overall strategies for solving these types of problems can be found in [PS85].

One ubiquitous class of algorithms for solving geometric queries is termed **plane-sweep**. Originally formulated for problems in the plane, it involves a vertical line that sweeps across the plane from left to right, encountering various "event" points, which cause a sweep line status to be updated. Hinrichs et al [HNS88] use this method to provide a simple and optimal worst case ($\Theta(n \log n)$) algorithm to find the closest pair in a set of n points in the plane. As the vertical line sweeps across the plane, an ordered list of a subset of the points that are to the left of the plane is maintained. When the vertical line encounters a new point in the set, this point is tested with a subset of the points in the ordered list, to see if a new closest pair has been found. The plane-sweep technique can also be generalized to higher dimensions, and has been applied to problems such as detecting intersection of line segments or rectangles.

The method of **bucketing** or **fixed grid** techniques generally have a poor worst case time complexity, but perform very well in terms of expected time, under non-pathological conditions. Bucketing methods impose a uniformly spaced grid over the space of the problem, and partition geometric data into distinct buckets. Solving problems such as the nearest neighbour to a point in space is a simple matter of finding the relevant bucket and testing against the small number of data points in the bucket [Knu73]. Bucketing methods are simple to implement and generalize easily to higher dimensions.

In addition to these methods, there is a plethora of hierarchical data structures applicable to geometric proximity queries. Some of the more common ones include the multidimensional binary tree (**kd-tree**)

[PS85], the **quadree**, and variants thereof [SW, Sam], as well as **binary splitting plane** (**BSP**) trees [FKN80], the **segment** tree, and the **range** tree [Ben79].

A brief survey of published methods of solving proximity problems demonstrates the wealth of methods available for assisting in solving the problem of creating geometrically simple surfaces in the context of segmentation. Dickerson et al [DE96] use Delaunay triangulations to solve various inter-point distances in two or more dimensions. Gupta et al [GJS96] use plane sweep techniques to detect collisions and determine minimum inter-object separation within sets of points, line segments, and axes-parallel hyper-rectangles which are moving along linear trajectories. Canny [Can86] presents algebraic formulae for detecting collision between moving polyhedra, whereas efficient algorithms for determining minimum distance between circles and line segments in 3-space are presented by Neff [Nef90] and Lumelsky [Lum85], respectively. Gilbert et al [GJK88, GF90] provide a mathematical programming method to find the distance between arbitrary convex sets in any dimensional space. These methods and others may be advantageously applied to the task of surface segmentation of geometrically simple surfaces.

Bibliography

[AERT92] A. Aggarwal, H. Edelsbrunner, P. Raghavan, and P. Tiwari. Optimal time bounds for some proximity problems in the plane. *Inform. Process. Lett.*, 42(1):55–60, 1992.

[AGP91] Y. Amit, U. Grenander, and M. Piccioni. Structural image restoration through deformable templates. *Journal of the American Statistical Association*, 86(414):376–387, June 1991.

[AS95] David Adalsteinsson and James A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118:269–277, 1995.

[BB88] Ronen Barzel and Alan H Barr. Modeling system based on dynamic constraints. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 22(4):179–188, Aug 1988.

[Bcc89] R. Bajcsy and Kovačič. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 46:1–21, 1989.

[Ben79] J. L. Bentley. Decomposable searching problems. *Information Procesing Letters*, 8(5):244–251, 1979.

[BGCK93] K. T. Bae, M. L. Giger, C. T. Chen, and C. E. Kahn Jr. Automatic segmentation of liver structure in ct images. *Journal of Medical Physics*, 20(1):71–8, Jan-Feb 1993.

[BGK92] C. Brechbühler, B. Gerig, and O. Kübler. Surface parameterization and shape description. In *Proceedings Visualization in Biomedical Computing 1992*, pages 80–144, 1992.

[Blo88] Jules Bloomenthal. Polygonization of implicit surfaces. *Computer Aided Geometric Design*, 5(4):341–355, Nov 1988.

[Boo89] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, June 1989.

[BW90] J. Bloomenthal and B. Wyvill. Interactive techniques for implicit modeling. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, pages 109–116, 1990.

[CAC91] Isaac Cohen, Nicholas Ayache, and Laurent Cohen. Segmenting, visualizing and characterizing 3d anatomical structures with deformable surfaces. *Proceedings of the Annual Conference on Engineering in Medicine and Biology*, 13:1183–1184, 1991.

[Can86] John Canny. Collision detection for moving polyhedra. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(2):200–209, March 1986.

[Car76] M.P. Do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

[CC90] Laurent D Cohen and Issac Cohen. A finite element method applied to new active contour models and 3d reconstruction from cross sections. *Proc 3 Int Conf Comput Vision*, pages 587–591, 1990.

[CC93] Laurent D Cohen and Isaac Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, Nov 1993.

[CCA91] Isaac Cohen, Laurent D Cohen, and Nicholas Ayache. Introducing new deformable surfaces to segment 3d images. In *Proceedings IEEE Comput Soc Conf Comput Vision Pattern Recognition*, pages 738–739, 1991.

[CCA92] I. Cohen, L.D. Cohen, and N. Ayache. Using deformable surfaces to segment 3-d images and infer differential structures. *Computer Vision, Graphics, and Image Processing*, 1992.

[CCR+93] S. Cagnoni, G. Coppini, M. Rucci, D. Caramella, and G. Valli. Neural network segmentation of magnetic resonance spin echo images of the brain. *Journal of Biomedical Engineering*, 15(5):355–62, Sep 1993.

[CDVE95] G. J. Carman, H. A. Drury, and D. C. Van Essen. Computational methods for reconstructing and unfolding the cerebral cortex. *Cerebral Cortex*, 5(6):506–17, Nov-Dec 1995.

[CHTH93] T.F. Cootes, A. Hill, C.J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. In *Proceedings of Information Processing in Medical Imaging*, pages 33–47, Flagstaff, Arizona, June 1993.

[CL88] James S Chen and Wei-Chung Lin. New surface interpolation technique for reconstructing 3-d objects from serial cross-sections. In *Proceedings - International Conference on Pattern Recognition 9th.*, pages 1100–1102, 1988.

[CLL+88] H. E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford, and B. C. Teeter. Two algorithms for the three-dimensional reconstruction of tomograms. *Journal of Medical Physics*, 15(3):320–7, May-Jun 1988.

[CNPE94] D.L. Collins, P. Neelin, T.M. Peters, and A.C. Evans. Automatic 3d intersubject registration of mr volumetric data in standardized talairach space. *Journal of Computer Assisted Tomography*, 18(2):192–205, March 1994.

[CPDE92] D.L. Collins, T.M. Peters, W. Dai, and A.C. Evans. Model-based segmentation of individual brain structures from mri data. In *Proceedings Visualization in Biomedical Computing 1992*, pages 10–23, 1992.

[CS89] Kenneth L. Clarkson and Peter W. Shor. Applications of random sampling in computational geometry, ii. *Discrete and Computational Geometry*, 4:387–421, 1989.

[CW83] F. Chin and C. A. Wang. Optimal algorithms for the intersection and the minimum distance problems between planar polygons. *IEEE Trans. Comput.*, C-32(12):1203–1207, 1983.

[D̈88] M.J. Düurst. Additional reference to marching cubes (letter). *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 22:72–73, 1988.

[DB95] Chris Davatzikos and R Nick Bryan. Using a deformable surface model to obtain a shape representation of the cortex. *Proceedings of the IEEE International Conference on Computer Vision 1995*, pages 212–217, 1995.

[DE96] M. T. Dickerson and D. Eppstein. Algorithms for proximity problems in higher dimensions. *Comput. Geom. Theory Appl.*, 5:277–291, 1996.

[DS93] A. M. Dale and M. I. Sereno. Improved localization of cortical activity by combining eeg and meg with mri cortical surface reconstruction: a linear approach. *Journal of Cognitive Neuroscience*, 5:162–176, 1993.

[EMN+92] A.C. Evans, S. Marrett, P. Neelin, T. Gum, W. Dai, S. Milot, E. Meyer, and D. Bub. Anatomical mapping of functional activation in stereotactic coordinate space. *Neuroimage*, 1:43–53, 1992.

[EPO91] A B Ekoule, F C Peyrin, and C L Odet. Triangulation algorithm from arbitrary shaped multiple planar contours. *ACM Transactions on Graphics*, 10(2):182–199, Apr 1991.

[FB88] David R Forsey and Richard H Bartels. Hierarchical b-spline refinement. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 22(4):205–212, Aug 1988.

[FKN80] H. Fuchs, Z.M. Kedem, and B.F. Naylor. On vision surface generation by a proiri tree structures. *Computer Graphics*, 14(3):124–133, 1980.

[FKU77] H. Fuchs, Z.M. Kedem, and S.P. Uselton. Optimal surface reconstruction from planar contours. *Communications of the ACM*, 20(10):693–702, October 1977.

[FvDFH90] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, 2 edition, 1990.

[GA93] Andre Gueziec and Nicholas Ayache. Large deformable splines, crest lines, and matching. *Proceedings of SPIE - The International Society for Optical Engineering*, 2031:316–327, 1993.

[GD82] S. Ganapathy and T.G. Dennehy. A new general triangulation method for planar contours. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 16(3):69–75, 1982.

[GF90] Elmer G Gilbert and Chek-Peng Foo. Computing the distance between general convex objects in three-dimensional space. *IEEE Transactions on Robotics & Automation*, 6(1):53–61, Feb 1990.

[GJK88] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi. A fast procedure for computing the distance between complex objects. *??*, 4(2), 1988.

[GJS96] Prosenjit Gupta, Ravi Janardan, and Michiel Smid. Fast algorithms for collision and proximity problems involving moving geometric objects. *Computational Geometry Theory and Applications*, 6:371–391, 1996.

[GS93] R.J-M. Grognard and A.D. Seagar. Modeling surfaces using spherical harmonics. Technical report, CSIRO, Sydney, Australia, 1993.

[HEM92] M.E. Hyche, N.F. Ezquerra, and R. Mullick. Spatiotemporal detection of arterial structure using active contours. In *Proceedings Visualization in Biomedical Computing 1992*, pages 52–62, 1992.

[HH92] K. H. Hohne and W. A. Hanson. Interactive 3d segmentation of mri and ct volumes using morphological operations. *Journal of Computer Assisted Tomography*, 16(2):285–94, Mar-Apr 1992.

[HNS88] Klaus Hinrichs, Jurg Nievergelt, and Peter Schorn. Plane-sweep solves the closest pair problem elegantly. *Information Processing Letters*, 26(5):255–261, Jan 1988.

[HW90] Mark Hall and Joe Warren. Adaptive polygonalization of implicitly defined surfaces. *IEEE Computer Graphics and Applications*, 10(6):33–42, Nov 1990.

[KCHN91] A.D. Kalvin, C.B. Cutting, B. Haddad, and M.E. Noz. Constructing topologically connected surfaces for the comprehensive analysis of 3d medical structures. In *Proceedings of SPIE – Medical Imaging V: Image Processing*, pages 247–258, 1991.

[KCS+92] M. Kamber, D. L. Collins, R. Shinghal, G. S. Franci s, and A. C. Evans. Model-based 3D segmentation of multiple sclerosis lesions in dual-echo MRI data. In *Visualization in Biomedical Computing*, volume 1808, pages 590–600, Chapel Hill, North Carolina, 13-16 October 1992. SPIE.

[KGC+89] Stane Kovacic, Jim C Gee, Wallace S L Ching, Martin Reivich, and Ruzena Bajcsy. Three-dimensional registration of pet and ct images. *Proceedings of the Annual Conference on Engineering in Medicine and Biology*, 11(89):548–549, 1989.

[KGV83] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[Knu73] D.E. Knuth. *The Art of Computer Programming 3: Sorting and Searching*. Addison-Wesley Publishing Company, 1973.

[KWT88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.

[LC87] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 21(4):163–169, Jul 1987.

[LHH+91] E. D. Lehmann, D. J. Hawkes, D. L. Hill, C. F. Bird, G. P. Robinson, A. C. Colchester, and M. N. Maisey. Computer-aided interpretation of spect images of the brain using an mri-derived 3d neuro-anatomical atlas. *Medical Informatics*, 16(2):151–66, Apr-Jun 1991.

[Lum85] V. J. Lumelsky. On fast computation of distance between line segments. *Inform. Process. Lett.*, 21:55–61, 1985.

[MBA93] Gregoire Malandain, Gilles Bertrand, and Nicholas Ayache. Topological segmentation of discrete surfaces. *International Journal of Computer Vision*, 10(2):183–197, Apr 1993.

[MBL+91] J.V. Miller, D.E. Breen, W.E. Lorensen, R.M. O'Bara, and M.J. Wozny. Geometrically deformed models: a method for extracting closed geometric models from volume data. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 25(4):217–226, July 1991.

[MCAG92] M.I. Miller, G.E. Christensen, Y. Amit, and U. Grenander. A mathematical textbook of deformable neuro-anatomies. Technical report, Washington University School of Medicine, St. Louis, MO, August 1992.

[MS96] J. Matoušek and O. Schwarzkopf. A deterministic algorithm for the three-dimensional diameter problem. *Comput. Geom. Theory Appl.*, 6:253–262, 1996.

[MSS92] D. Meyers, Shelley Skinner, and Kenneth Sloan. Surfaces from contours. *ACM Transactions on Graphics*, 11(3):228–258, July 1992.

[MSV95] Ravikanth Malladi, James A Sethian, and Baba C Vemuri. Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175, Feb 1995.

[MT85] M. McKenna and G. T. Toussaint. Finding the minimum vertex distance between two disjoint convex polygons in linear time. *Comput. Math. Appl.*, 11:1227–1242, 1985.

[MT96] Tim McInerney and Demetri Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2), 1996.

[NA93a] C. Nastar and N. Ayache. Non-rigid motion analysis in medical images: a physically based approach. In *Proceedings of Information Processing in Medical Imaging*, pages 17–32, Flagstaff, Arizona, June 1993.

[NA93b] Chahab Nastar and Nicholas Ayache. Physically based analysis of deformations in 3d images. *Proceedings of SPIE - The International Society for Optical Engineering*, 2031:182–192, 1993.

[NAF94] T. Cizadilo S. Arndt D.S. O'Leary V. Swayze N.C. Andreasen, G. Harris and M. Flaum. Techniques for measuring sulcal, gyral patterns in the brain as visualized through magnetic resonance scanning: Brainplot and brainmap. *Proceedings of the National Academy of Science, USA*, 90:93–97, January 1994.

[Nef90] C. A. Neff. Finding the distance between two circles in three-dimensional space. *IBM J. Res. Develop.*, 34:770–775, 1990.

[Nil90] Nils J. Nilsson. *The Mathematical Foundations of Learning Machines.* Morgan Kaufmann Publishers, San Mateo, California, 1990.

[OPY92] Mattias Ohlsson, Carsten Peterson, and Alan L Yuille. Track finding with deformable templates - the elastic arms approach. *Computer Physics Communications*, 71(1):77–98, Aug 1992.

[PA95] Janos Pach and Pankaj K. Agarwal. *Combinatorial Geometry.* John Wiley and Sons, Inc., New York, 1995.

[Pen88] Alex Pentland. Shape information from shading: A theory about human perception. *Second Int Conf on Comput Vision*, pages 404–413, 1988.

[Pen89] A. Pentland. Shape information from shading: a theory about human perception. *Spatial Vision*, 4(2-3):165–82, 1989.

[Pen90] A. Pentland. Automatic extraction of deformable part models. *International Journal of Computer Vision*, pages 107–126, 1990.

[PFTV88] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, Cambridge, England, 1988.

[PH91] Alex Pentland and Bradley Horowitz. Recovery of nonrigid motion and structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):730–742, Jul 1991.

[PHS93] Alex Pentland, Bradley Horowitz, and Stan Sclaroff. Non-rigid motion and structure from contour. *Proc IEEE Workshop Visual Motion*, pages 288–293, 1993.

[PS85] F. Preparata and M. Shamos. *Computational Geometry – An Introduction.* Springer–Verlag, New York, 1985.

[PS91] Alex Pentland and Stan Sclaroff. Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):715–729, Jul 1991.

[PT90] Bradley A Payne and Arthur W Toga. Surface mapping brain function on 3d models. *IEEE Computer Graphics and Applications*, 10(5):33–41, Sep 1990.

[PT92] Bradley A Payne and Arthur W Toga. Distance field manipulation of surface models. *IEEE Computer Graphics and Applications*, 12(1):65–71, Jan 1992.

[PW89] A. Pentland and J. Williams. Good vibrations: modal dynamics for graphics and animation. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 23(4):215–222, 1989.

[RLW93] E. Rinast, R. Linder, and H. D. Weiss. Neural network approach for computer-assisted interpretation of ultrasound images of the gallbladder. *European Journal of Radiology*, 17(3):175–8, Nov 1993.

[RTL+93] H. Rusinek, W.H. Tsui, A.V. Levy, M.E. Noz, and M.J. deLeon. Principal axes and surface fitting methods for three-dimensional image registration. *Journal of Nuclear Medicine*, 34(11), November 1993.

[Sam] Hanan Samet. The quadtree and related hierarchical data structures. *Computing Surveys v 16, June 1984, p 187-260.*

[Sch81] J. T. Schwartz. Finding the minimum distance between two convex polygons. *Inform. Process. Lett.*, 13:168–170, 1981.

[SD92a] Lawrence H Staib and James S Duncan. Boundary finding with parametrically deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(11):1061–1075, Nov 1992.

[SD92b] L.H. Staib and J.S. Duncan. Deformable fourier models for surface finding in 3d images. In *Proceedings Visualization in Biomedical Computing 1992*, pages 90–104, 1992.

[Ser82] Jean Paul Serra. *Image analysis and mathematical morphology.* Academic Press, London, 1982.

[SG93] A.D. Seagar and R.J.-M. Grognard. Fourier descriptors of curves and surfaces. *Computer Vision, Graphics, and Image Processing*, 1993.

[Sie91] Raimund Siedel. *New Trends in Discrete and Computational Geometry.* Springer-Verlag, Berlin, 1991.

[SL] Lawrence H Staib and Xianzhang Lei. Intermodality 3d medical image registration with global search. *Proc IEEE Workshop Biomed Image 1994*, pages 225–234.

[SL97] S. Sandor and R. Leahy. Surface-based labeling of cortical anatomy using a deformable atlas. *IEEE Transactions on Medical Imaging*, 16(1):41–54, Feb 1997.

[SMG+93a] J W Snell, M B Merickel, J C Goble, J B Brookeman, and N F Kassell. Model-based segmentation of the brain from 3-d mri using active surfaces. *Bioengineering, Proceedings of the Northeast Conference*, pages 164–165, 1993.

[SMG+93b] John W Snell, Michael B Merickel, John C Goble, James B Brookeman, and Neal F M D Kassell. Model-based segmentation of the brain from 3d mri using active surfaces. *Proceedings of SPIE - The International Society for Optical Engineering*, 1898:210–219, 1993.

[SP93] Stan Sclaroff and Alexander P Pentland. Modal models: energy-based implicit functions. *Proceedings of SPIE - The International Society for Optical Engineering*, 1828:14–23, 1993.

[ST92] Richard Szeliski and David Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 26(2):185–194, Jul 1992.

[SW] Hanan Samet and Robert E Webber. Hierarchical data structures and algorithms for computer graphics. *IEEE Computer Graphics & Applications v 8, May 1988, p 48-68– Related material:v8 p59-75 Jl '88.*

[SZE97] John G. Sled, Alex P. Zijdenbos, and Alan C. Evans. A comparison of retrospective intensity non-uniformity correction methods for MRI. In *Information Processing in Medical Imaging*, 1997. in press.

[SZL92] W.J. Schroeder, J.A. Zarge, and W.E. Lorensen. Decimation of triangle meshes. *Computer Graphics: A Quarterly Report of SIGGRAPH-ACM*, 26(2):65–70, July 1992.

[TJP+93] T.G. Turkington, R.J. Jaszczak, C.A. Pelizzari, C.C. Harris, J.R. MacFall, J.M. Hoffman, and R.E. Coleman. Accuracy of registration of pet, spect and mr images of a brain phantom. *Journal of Nuclear Medicine*, 34(9):1587–1594, September 1993.

[TM86] P.H. Todd and R.J.Y. McLeod. Numerical estimation of the curvature of surfaces. *Computer-Aided Design*, 18(1):33–37, January 1986.

[TT88] J. Talairach and P. Tournoux. *Co-Planar Stereotactic Atlas of the Human Brain: 3-Dimensional Proportional System: An Approach to Cerebral Imaging*. Verlag, Stuttgart, 1988.

[TWK87] D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models and 3d object reconstruction. *International Journal of Computer Vision*, 1(3):211–221, October 1987.

[TWK88] Demetri Terzopoulos, Andrew Witkin, and Michael Kass. Constraints on deformable models: Recovering 3d shape and nonrigid motion. *Artificial Intelligence*, 36(1):91–123, Aug 1988.

[UH92] J.K. Udupa and G.T. Herman. Boundaries in multidimensional digital scenes: Theory and algorithms. Technical report, Department of Radiology, University of Pennsylvania, January 1992.

[WIGKJ94] W.M. Wells III, W.E.L. Grimson, R. Kikinis, and F.A. Jolesz. Statistical intensity correction and segmentation of mri data. In *Proceedings Visualization in Biomedical Computing 1994*, pages 13–24, 1994.

[Wil64] Douglass J. Wilde. *Optimum Seeking Methods*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1964.

[WMW86] G. Wyvill, C. McPheeters, and B. Wyvill. Data structure for soft objects. *The Visual Computer*, 2:227–234, 1986.

[WP92] John R Williams and Alex P Pentland. Superquadrics and modal dynamics for discrete elements in interactive design. *Engineering Computations (Swansea, Wales)*, 9(2):115–127, Apr 1992.

[WS89] Estarose Wolfson and Eric L Schwartz. Computing minimal distances on polyhedral surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(9):1001–1005, Sep 1989.

[WWW87] P Widmayer, Y. F Wu, and C. K Wong. On some distance problems in fixed orientations. *SIAM Journal on Computing v 16, Aug*, pages 728–46., 1987.

[ZDM93] Alex P Zijdenbos, Benoit M Dawant, and Richard A Margolin. Measurement reliability and reproducibility in manual and semi-automatic mri segmentation. *Proceedings of the Annual Conference on Engineering in Medicine and Biology*, 15:162–163, 1993.