Segmentation of a Human Brain Cortical Surface Mesh Using Watersheds

Kathleen Marty, CS766 Final Project, 12/17/02

1.0 Abstract

A 3D cortical surface mesh is a convoluted structure. The purpose of segmenting such a mesh is to impose a higher level structure which represents something about the underlying structure of the mesh itself. This segmentation should reduce the mesh into "meaningful," connected pieces. In this paper, segmentation using the watershed algorithm is implemented on brain cortical surface meshes. The height function used is a curvature measure inherent in the geometry of the mesh. Four different curvature measures are compared: mean, gaussian, absolute, and root mean square.

2.0 Introduction and Motivation

A 3D cortical surface mesh exists simply as a collection of connected polygons. The purpose of segmenting such a mesh is to impose a higher level structure which represents something about the underlying structure of the mesh itself. This segmentation should reduce the mesh into *meaningful*, connected pieces. "Meaningful" implies that the partitioned areas are relevant to the application at hand. For example, segmented sulci from a brain mesh can serve as landmarks, which can be used to register the mesh with other brain meshes to make comparisons. These comparisons could serve to measure brain growth, identify diseases, etc. In addition, the segmented surface can serve as a visualization tool.

The most common segmentation of a cortical mesh is into sulal and gyral regions. The gyri of a brain can be defined as the top surfaces of the brain folds (ridges). The sulci of a brain can be defined as the area within the brain folds (basins). Segmentation of a cortical surface in terms of sulci and gyri can occur in several ways. The most general division is a separation of sulci and gyri into two labelings. Another division could be between individual sulci and gyri, that is, one labeling for all gyri and separate labelings for each sulci region.

3.0 Problem Statement

The purpose of this paper is to segment a brain mesh using the watershed algorithm. The watershed algorithm is implemented as described in [13] and [10] to segment a brain cortical surface mesh, using curvatures as a "height function" for the watershed algorithm. A region-merging step is also implemented. In addition, different curvature estimates as described in [9] will be applied and the results given.

4.0 Related Work

Related work involving the segmentation of sulci using waterheds is done by Rettmann, et. al [10]. This work focuses on segmenting the actual cortical regions surrounding sulci, referred to as *sulcal regions*. This paper uses the geodesic depth of mesh points in the sulci regions as the *height function* of the watershed algorithm.

Other work on segmenting sulci has involved methods such as fitting a surface [14], extracting the volumetric regions within sulcal spaces [4], finding a set of points [7], or extracting curve representations of the sulci [12]. Other relevant references for these methods are listed in [10]. These methods vary in the data they use in that some segment directly from MR images, whereas others segment from a surface mesh.

The classic work using the watershed algorithm for image segmentation is described in Serra [11]. This method was extended to arbitrary 3D meshes by Mangan and Whitaker [6], using discrete curvatures. An efficient watershed method based upon immersions was created by Vincent and Soille [13].

5.0 Theory

Watersheds is a concept taken from the field of topology. The idea is quite straightforward: simulate rain falling upon a surface, and each drop will descend a gradient until it reaches a local minima, or *catchment basin*. The ridges that separate catchment basins are defined as *watersheds*.

Applying watersheds to a surface mesh occurs in two independent steps. A sorting step based upon vertex "heights" and a flooding step. A third step, region merging, can be added to eliminate insignificant basin labelling. These steps are described in more detail below.

5.1 Sorting step (define height function)

The first step is to define a height function for each vertex of the mesh, followed by a sort of the vertices based upon their calculated heights. This height function will determine the order in which the vertices are *flooded* in the second step. It is interesting to note that watersheds may be extended to n-dimensions because the height function is 1-dimensional. That is, for each element in an image, mesh, etc., there is only one value that defines each element's height. Determining a height function for a 3-dimensional mesh is a difficult problem. In general, two different methods are used that use the intuitive structure of the mesh: geodesic distances or curvature measures.

5.1.1 Geodesic distances as heights

Geodesic distances can be determined on a brain mesh by the following concepts (described in detail in [10]). All gyral regions have zero geodesic depth by definition. A gyral "shrink-wrap" can be wrapped around the surface to define the maximum height. The geodesic depth from the gyral shrink-wrap to the vertices in the sulcal regions can be calculated using the fast marching method extended to triangulated domains by Kimmel and Sethian [3].

5.1.2 Curvatures as heights

Curvatures can also be used as a height measure. The idea is that ridges and basins have opposite signed curvatures, and the cortical surface is naturally divided between ridges (gyri) and basins (sulci). The major complication is that curvature cannot be directly evaluated for triangle meshes because it is mathematically defined for smooth surfaces only. However, discrete differential-geometry operators have been developed which can estimate curvatures on triangulated manifolds. These derivations are beyond the scope of this paper but can be explored in [1], [8], and [9]. To obtain the curvatures, the surface can be locally parameterized to satisfy the smoothness constraint. From the parameterization, the first and second *fundamental forms* can be estimated directly. From these fundamental forms, one can calculate various curvature measures, such as the Gaussian, Mean, Root-Mean-Squared, and Absolute curvatures. This is briefly described next, as given in [9] and [8].

Given a parametric surface of the form $\mathbf{x}=\mathbf{x}(\mathbf{u})$; where:

$$\mathbf{u} = (u, v) \in [\mathbf{a},\mathbf{b}] \subset \Re^2 \qquad \text{(EQ 1)}$$

where u and v take real values. The functions

$$\mathbf{x}(u, v) = (x(u, v), y(u, v), z(u, v)) \qquad \text{(EQ 2)}$$

are single valued and continuous, and are assumed to have continuous partial derivatives. Then, the first fundamental form is given by:

$$I = (\dot{\mathbf{x}} \cdot \dot{\mathbf{x}}) = E du^2 + 2F du dv + G dv^2 \qquad \text{(EQ 3)}$$

where

$$\begin{aligned} E &= \mathbf{x}_u^2 = \mathbf{x}_u \cdot \mathbf{x}_u, \\ F &= \mathbf{x}_u \cdot \mathbf{x}_v, \\ G &= \mathbf{x}_u^2 = \mathbf{x}_v \cdot \mathbf{x}_v \end{aligned} \qquad \text{(EQ 4)}$$

The second fundamental form is given by:

$$II = L du^2 + 2M du dv + N dv^2 \qquad \text{(EQ 5)}$$

where

$$L = \mathbf{N}\mathbf{x}_{uu},\quad M = \mathbf{N}\mathbf{x}_{uv},\quad N = \mathbf{N}\mathbf{x}_{vv} \qquad \text{(EQ 6)}$$

and $\mathbf{N}$ is the surface normal at point **x.**

The *normal curvature* of the surface at point **x** in the direction of tangent **t** is given by

$$\kappa_0 = \kappa_0(\mathbf{x},\mathbf{t}) = \frac{II}{I} \qquad \text{(EQ 7)}$$

The normal curvature is based on direction, and therefore attains maximum and minimum values, called the principal curvatures, κ_1 and κ_2.

The *Gaussian curvature* can then be given by:

$$K = \kappa_1\kappa_2 = \frac{LN - M^2}{EG - F^2} \qquad \text{(EQ 8)}$$

Mean curvature is then given by:

$$H = \frac{\kappa_1 + \kappa_2}{2} = \frac{1}{2} \times \frac{NE - 2MF + LG}{EG - F^2} \qquad \text{(EQ 9)}$$

Root mean square curvature (RMS) is given by:

$$\kappa_{rms} = \sqrt{\frac{\kappa_1^2 + \kappa_2^2}{2}} = \sqrt{4H^2 - 2K} \qquad \text{(EQ 10)}$$

Absolute curvature is given by:

$$\kappa_{abs} = |\kappa_1| + |\kappa_2| = \left|H + \sqrt{H^2 - K}\right| + \left|H - \sqrt{H^2 - K}\right| \qquad \text{(EQ 11)}$$

Using the discrete curvature estimates directly, in general, do not give an accurate estimate of the actual curvatures of the brain. This is because the discrete mesh is inherently noisy, and taking the second derivatives to obtain the curvatures of the mesh only exacerbates the noise. The noise is a result of image acquisition, mesh creation, and surface parameterization. Using a filter can help smooth out the noise, but at the expense of distorting features. Gaussian smoothing is the preferred method in image analysis. The smoothing method used in this paper is *diffusion smoothing* as used in [1]. Diffusion smoothing is a generalization where Gaussian kernel smoothing is reformulated as a solution of a diffusion equation on a Riemannian manifold. The application of smoothing on curved surfaces uses the Laplace-Beltrami operator. Details of using this operator on triangular meshes can be found in [1] and [8].

5.2 Flooding step

After the vertices are sorted by their heights, the second step is progressive flooding of the catchment basins, in the order given by the sorting step. This step is succinctly described by Vincent and Soille [13]:

> By analogy, we can figure that we have pierced holes in each regional minimum of *I*, this picture being regarded as a (topographic) surface. We then slowly immerse our surface into a lake. Starting from the minima of lowest altitude, the water will progressively fill up the different catchment basins of *I*. Now, at each pixel where the water coming from two different minima would merge, we build a “dam”. At the end of this immersion procedure, each minimum is completely surrounded by dams, which delimit its associated catchment basin. The whole set of dams which has been built thus provides a tessellation of *I* in its different catchment basins. These dams correspond to the watersheds of *I*.

The major problem with this step is that each local minima produces its own catchment basin, and thus its own labeling. Because the cortical mesh is inherently noisy with many local minima, the result of the watershed algorithm is bound to be *oversegmented*.

5.3 Region merging step

A strategy used towards reducing the oversegmentation is to merge regions based upon a saliency measure and the structure of the catchment basins. There are a variety of metrics that may define insignificant regions. For example, shallow regions are relative constant in curvature and tend to be caused by noise. Therefore, one could define a minimum depth required for a catchment basin to merge with its surrounding catchment basins. Other measures could be used to merge regions such as following features through a scale-space to identify the most important features (the basins that others would merge into). This is an approach taken by Gauch [2], and not implemented in this paper.

The region merging strategy used in this paper is based on the height of the ridge separating two catchment basins. If the relative depth of both catchment basins is less than a threshold, then they are merged. This strategy was introduced by Rettmann, et. al [10]. The algorithm starts with the deepest basins as they are the ones that should be merged first.

6.0 Method

Below is a detailed description of the method, data, and tools used in this paper.

6.1 Mesh Description

The brain meshes used were obtained courtesy of Moo Chung of the University of WI Statistics Dept., as used in [1]. The format of the meshes are in the Montreal Neurological Institute (MNI) triangular mesh file format. These meshes were created from T1-weighted MRI images using the anatomic segmentation using proximities (ASP) method as described in [5]. Each mesh has 40,962 vertices and 81,920 triangle faces with the average internodal distance of 3 mm.

6.2 Curvatures and Smoothing

Matlab code given by Chung [1] was used to calculate Mean and Gaussian curvatures and smooth the curvatures using diffusion smoothing. The parameters to be set for the smoothing process are the FWHM (Full-width half maximum) size for the Gaussian kernel, in mm, and a boolean parameter set so the diffusion smoothing is calculated by either a local quadratic parameterization or a Finite Element Method. A typical FWHM parameter would equal five mm, as this is about the size of the brain folds. FEM was not used, as it is not as analytically accurate as the local quadratic parameterization method.

The Matlab code was extended to calculate RMS (Root-Mean-Squared) and Absolute curvatures for comparison purposes.

6.3 Watershed Algorithm

The sorting step was achieved using a standard sorting routine.

The flooding step was implemented using the watershed algorithm by Vincent and Soille [13]. To segment only the sulci and to leave the gyri as one region, a flooding threshold is used. For example, one could threshold the mean curvatures at zero, where the curvature changes sign.

6.4 Region-merging Extension

A region-merging step was added using the heuristic as found in [10]. This heuristic was defined in Section 5.3 on page 4. A typical region-merging threshold would be 10 mm.

6.5 Visualization Software

Freely available software, CARET, was used for visualization [15]. To use this software, code was written to convert files form the MNI file format to the more common ".vtk" file format.

Using CARET instead of Matlab offers numerous advantages. The CARET software colors each vertex its assigned color, and then interpolates the colors between differing vertices. Matlab only colors faces, and does no such interpolation, making it more difficult to see how the vertices are labeled. Matlab can interpolate between faces to create a smooth-looking surface, however this feature only works for simple surfaces and does not work on the cortical mesh, whereas CARET can smooth the cortical mesh in a visually appealing way.

7.0 Experimental Results

Results of four different iterations are given. Each iteration used a different curvature measure as the height value for the watershed algorithm. The four curvatures used were mean, gaussian, absolute, and root mean square (RMS).

7.1 Parameters Used

The parameters used in the algorithm are displayed on Tables 1-4 on page 7. These include: the smoothing kernel size (FWHM), the flooding threshold value, and the merging threshold value. The kernel size was set to five mm as this is the typical size of the feature being detected. The threshold values were determined empirically. That is, thresholds were chosen that produced the most visually appealing results. Visually appealing results are those in which: 1) most of the labeled vertices are inside the sulcal basins and are not on the gyri, and 2) there is a good amount of labeled vertices inside the sulcal basins, the idea being to "fill" up the basin.

7.2 Curvature value distributions

Curvature distribution results are shown on Tables 1-4 and in Figure 1 on pages 7-8. The curvature distributions are depicted graphically on Figure 1, and the ranges and mean values are given on Tables 1-4.

7.3 Labeling results

The labels resulting from the watershed segmentation algorithm are depicted graphically in Figures 2-4 on pages 9-11. Each label corresponds to a different color. Tables 1-4 describe how many vertices were labeled, how many different labels resulted after the flooding step, and how many different labels resulted after the merging step. In general, the merging step reduced the number of labels about 40-80%. Results from using absolute and RMS curvature measures show that more vertices overall are labeled than with using mean curvatures. The suspected reason for this is because the absolute curvatures and RMS curvatures are more easily affected by noisy areas, and therefore they produced more labeled vertices in the part of the brain where the brain stem connects (not shown pictorially). Otherwise, the mean, absolute and RMS all appear to give good results. However, the mean curvature as a height function would probably be preferred because more of the labeled vertices are within sulcal basins. The absolute and RMS curvatures follow closely behind in favoribility, their one flaw being that they have a tendency for some area labelings to "spread out" a lot more than other area labelings.

7.4 Results Interpretation

An inherent difficultly in the interpretation of these results is that there is no definition of what is correct. Some papers use expert opinions from neurologists on where sulci and gyri exist. However, visual results can be interpreted given the idea that gyri are the surfaces at the top of the brain folds (ridges), and that sulci are the surfaces within the brain folds (basins). This idea was described in Section 7.1 as *visually appealing results*.

8.0 Concluding Remarks

In this work, results were shown from the segmentation of a cortical surface using the watershed algorithm. Future work could include implementation of different height functions. Another line of inquiry could be the use of graph optimization techniques to segment the cortical mesh.

9.0 References

[1] Chung, M.K. *Statistical Morphometry in Computational Neuronatomy.* Ph.D. thesis, McGill Universtiy, Montreal, Canada, 2001.

[2] Gauch, J.M. *Image Segmentation and Analysis via Multiscale Gradient Watershed Hierarchies*. IEEE Transactions on Image Processing, Vol. 8, No. 1, Jan.1999.

[3] Kimmel, R. and Sethian, J.A. *Computing Geodesic Paths on Manifolds*. Proc. National Academy of Sciences, 1997.

[4] Lohmann, G., and von Cramon, D. Y. *Automatic Labelling of the Human Cortical Surface Using Sulcal Basins*. Medical Image Analysis, No. 4, pg 179-188, 2000.

[5] MacDonald, J.D., Kabani, N., Avis, D., and Evans, A.C. *Automated 3-D Extraction of Inner adn Outer Surfaces of Cerebral Cortex From MRI*. NeuroImage, No. 12, pg. 340-356, 2000.

[6] Mangan, A.P., Whitaker, R.T. *Partitioning 3D Surface Meshes Using Watershed Segmentation*. IEEE Transactions on Visualization and Computer Graphics, Vol. 5, No. 4, Oct.-Dec. 1999.

[7] Mangin, J.F., Frouin, V., Block, I., Regis, J., and Lopez-Krahe, J. *From 3D Magnetic Resonance Images to Structural Representations of the Cortex Topography Using Topology Preserving Deformations*. Math. Imaging Vision, No. 5, p. 297-318.

[8] Meyer, M., Desbrun, M., Schroder, P. and Barr, A. *Discrete Differential Geometry Operators for Triangulated 2-Manifolds*. Caltech, 2001.

[9] Pulla, S., Razdan, A. and Farin, G. *Improved Curvature Estimation for Watershed Segmentation of 3-Dimensional Meshes*. Arizona State University, 2001.

[10] Rettmann, M.E., Han, X., Xu, C., and Prince, J.L. *Automated Sulcal Segmentation Using Watersheds on the Cortical Surface*. NeuroImage, No.15, p. 329-344, 2002.

[11] Serra, J.P. *Image Analysis and Mathematical Morphology.* Academic Press, 1982.

[12] Tao, X., Han, X., Rettmann, M.E., Prince, J.L., and Davatzikos, C. *Statistical Study on Cortical Sulci of Human Brains.* Proc. of the XVIIth International Conference on Information Processing in Medical Imaging, p. 475-487, 2001.

[13] Vincent, L. and Soille, P. *Watersheds in Digital Spaces: An Effiencet Algorithm Based on Immersion Simulations*. IEEE Transactions on Pattern Analysis and Machine Intellignece, Vol. 13, No. 6, June 1991.

[14] Zhou, Y., Thompson, P.M., and Toga, A.W. *Extracting and Representing the Cortical Sulci*. Computer Graphics Applications, No. 19, p. 49-55.

[15] CARET software, by David Essen of Washington University. *http://stp.wustl.edu/resources/caretnew.html*

10.0 Appendix

Matlab and C++ code attached.

Table 1. Mean Curvature Segmentation

FWHM	5
Curvature distribution range	[-1.400,2.5440]
Mean of distribution	-0.0123
Flooding Threshold	-0.0123
Merging Threshold	10
Total Vertices	40962
Vertices labeled	19438
Labels after Flooding	1618
Labels after Merging	539

Table 2. Gaussian Curvature Segmentation

FWHM	5
Curvature distribution range	[-1.887, 0.2844]
Mean of distribution	-0.0117
Flooding Threshold	0.003
Merging Threshold	10
Total Vertices	40962
Vertices labeled	5844
Labels after Flooding	1034
Labels after Merging	780

Table 3. Absolute Curvature Segmentation

FWHM	5
Curvature distribution range	[0.0002, 2.5440]
Mean of distribution	0.0310
Flooding Threshold	0.018
Merging Threshold	10
Total Vertices	40962
Vertices labeled	23300
Labels after Flooding	1475
Labels after Merging	284

Table 4. RMS Curvature Segmentation

FWHM	5
Curvature distribution range	[0.0271, 4.9607]
Mean of distribution	0.2543
Flooding Threshold	0.21
Merging Threshold	10
Total Vertices	40962
Vertices labeled	23120
Labels after Flooding	1520
Labels after Merging	258

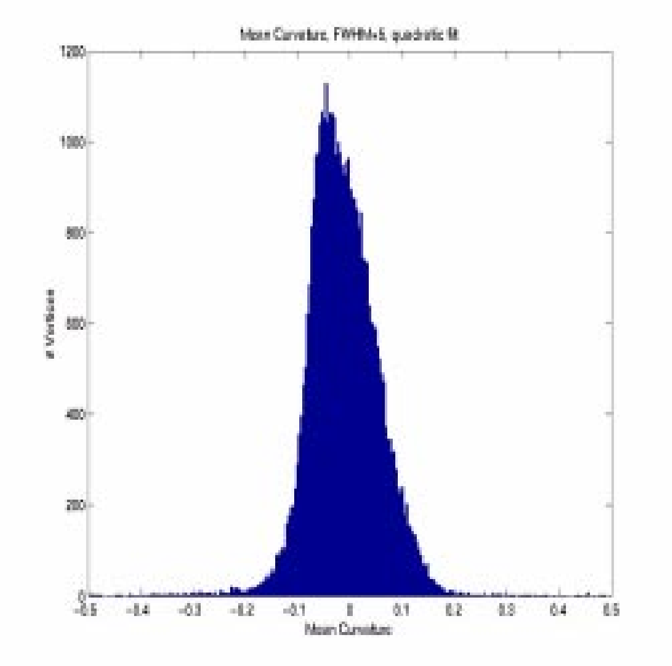

FIGURE 1.1 Distribution of mean curvatures for all vertices.

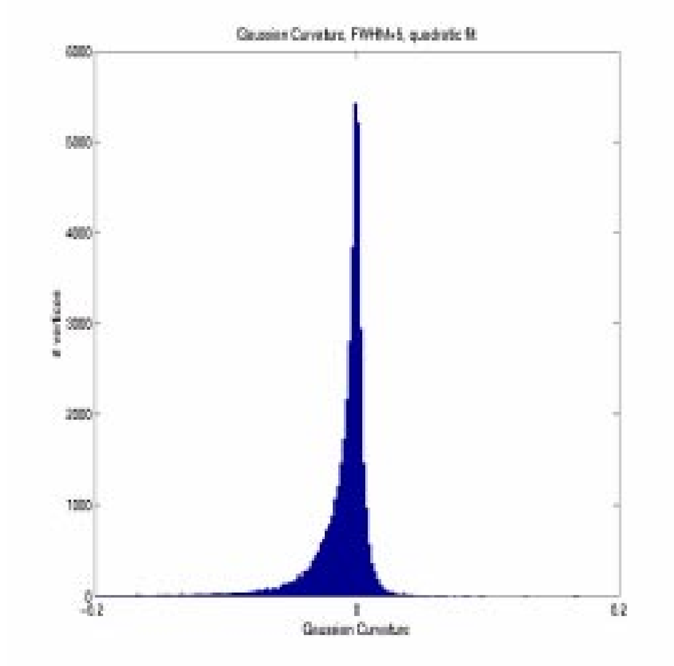

FIGURE 1.2 Distribution of gaussian curvatures for all vertices.

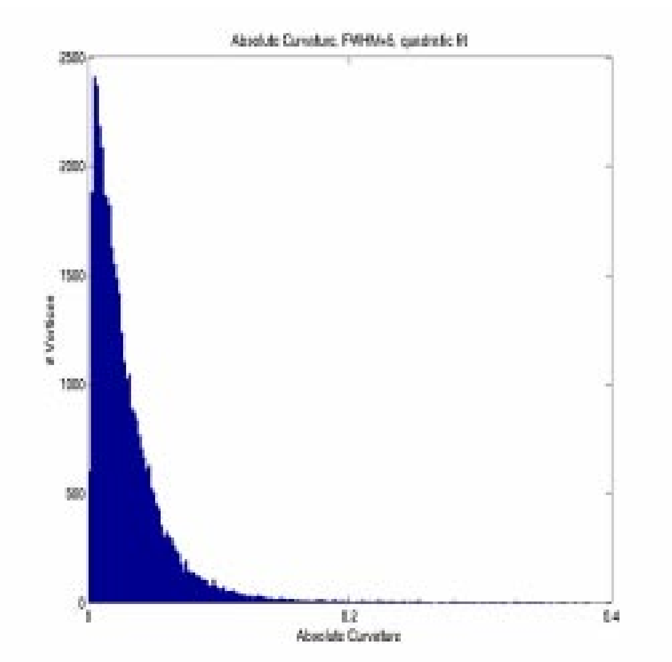

FIGURE 1.3 Distribution of absolute curvatures for all vertices.

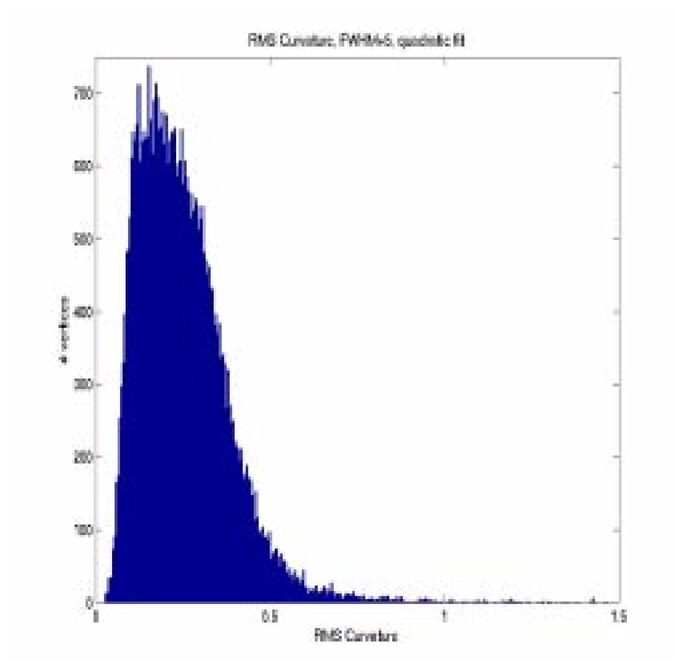

FIGURE 1.4 Distribution of root mean square (RMS) curvatures for all vertices.

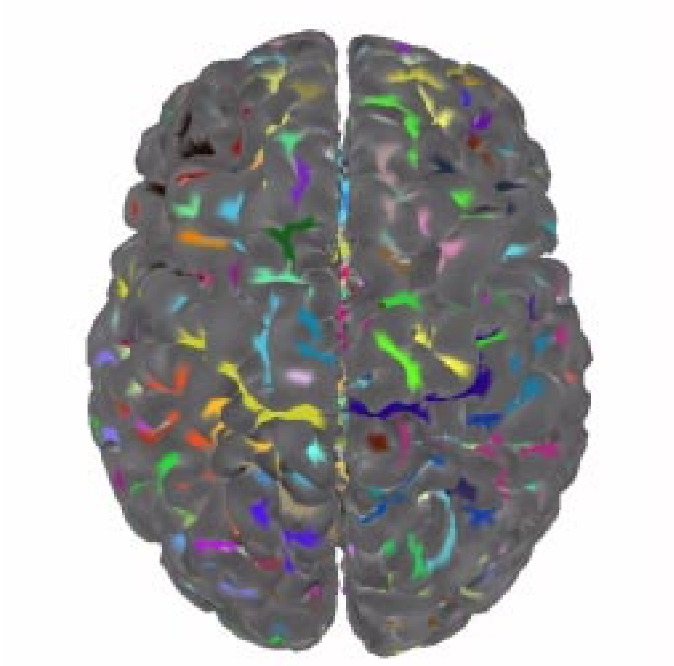

FIGURE 2.1 Top view of segmentation using mean curvatures.

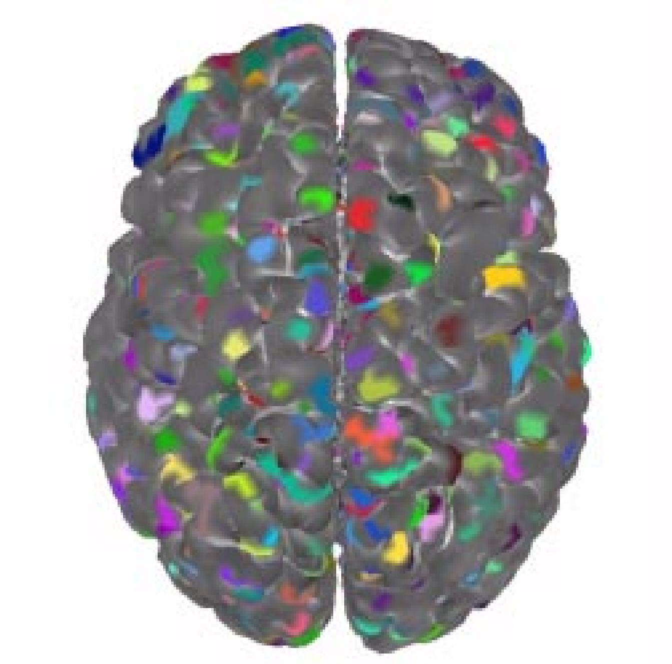

FIGURE 2.2 Top view of segmentation using gaussian curvatures.

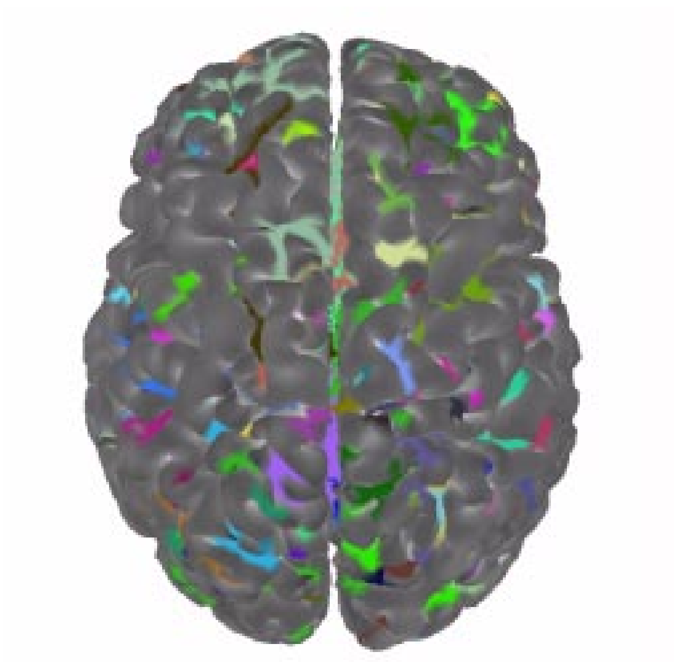

FIGURE 2.3 Top view of segmentation using absolute curvatures.

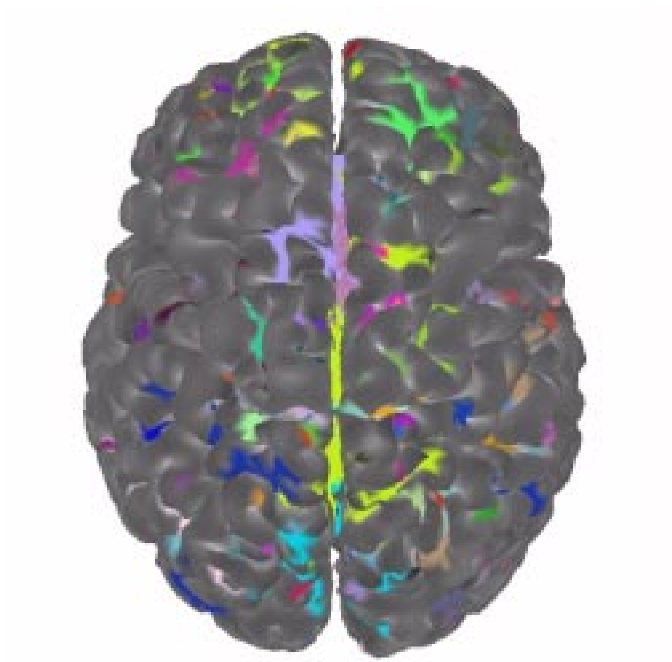

FIGURE 2.4. Top view of segmentation using root mean square curvatures.

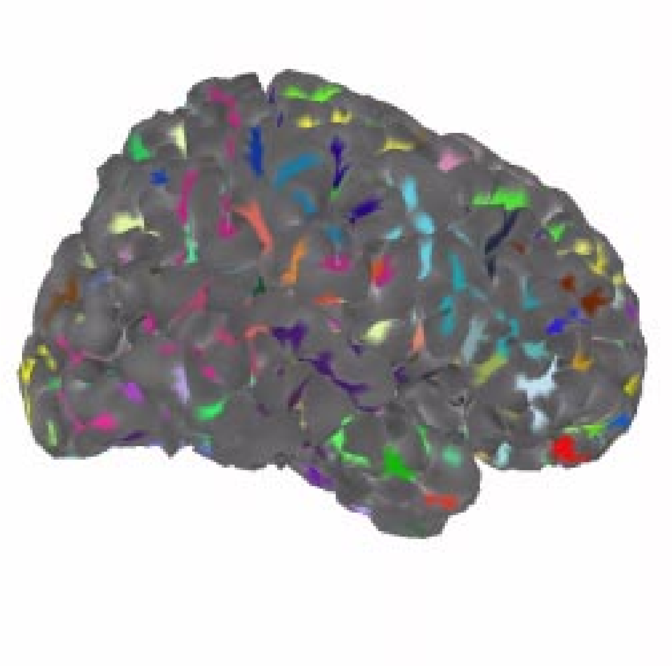

FIGURE 3.1 Lateral view of segmentation using mean curvatures.

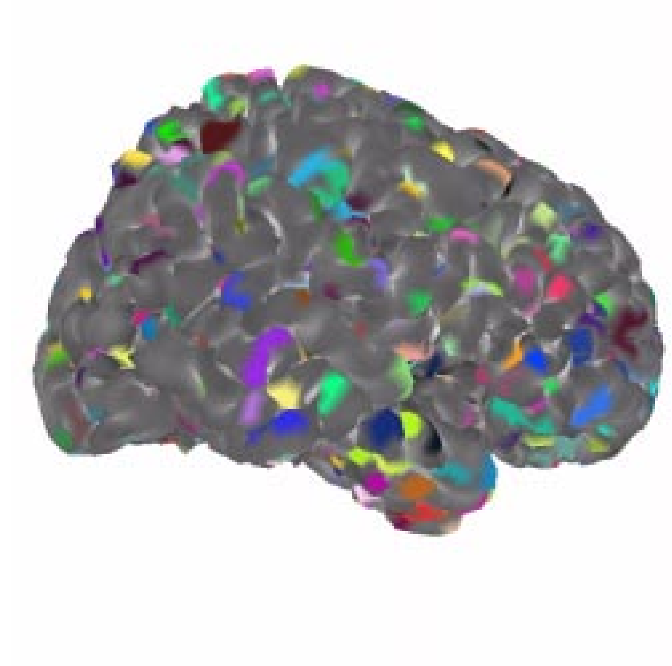

FIGURE 3.2 Lateral view of segmentation using gaussian curvatures.

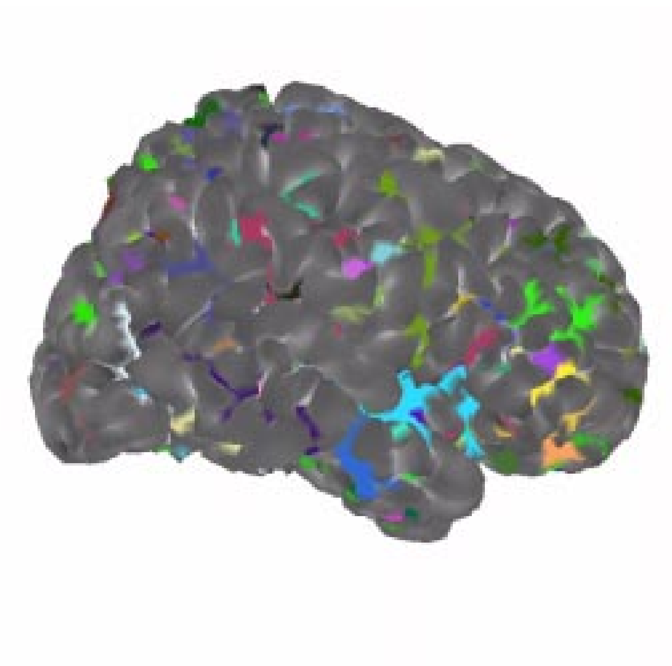

FIGURE 3.3 Lateral view of segmentation using absolute curvatures.

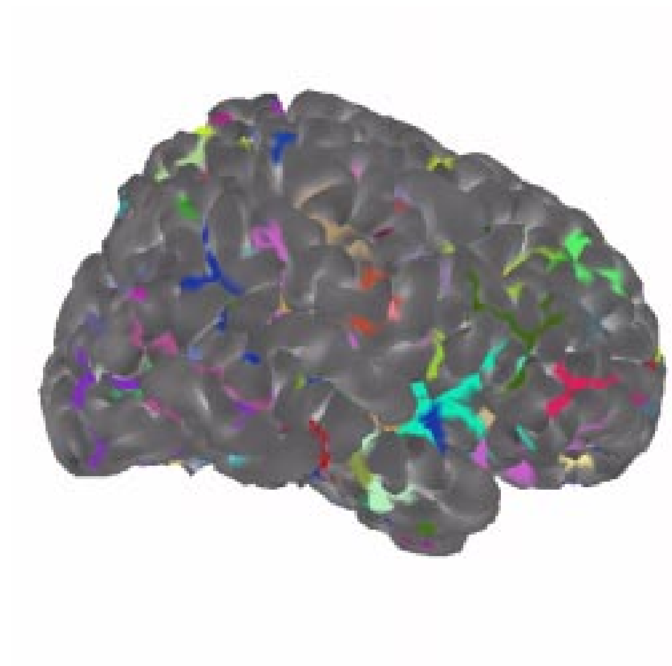

FIGURE 3.4 Lateral view of segmentation using root mean square curvatures.

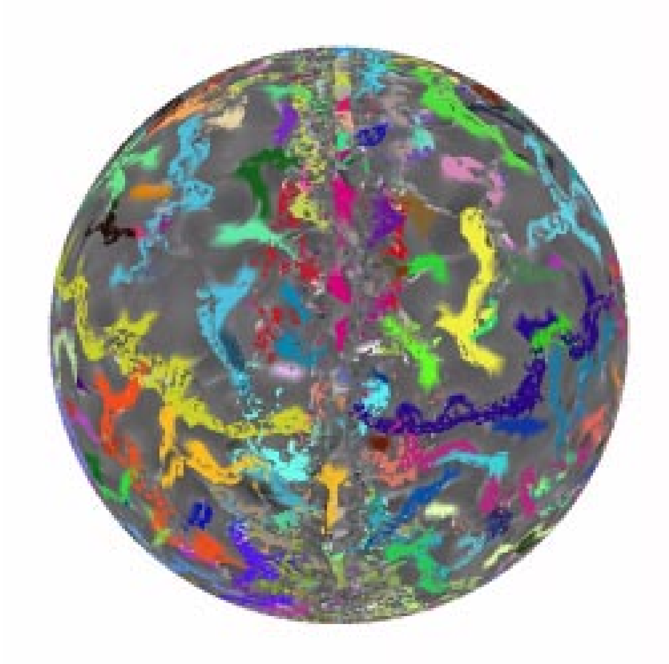

FIGURE 4.1 Top view of segmentation using mean curvatures projected to a sphere.

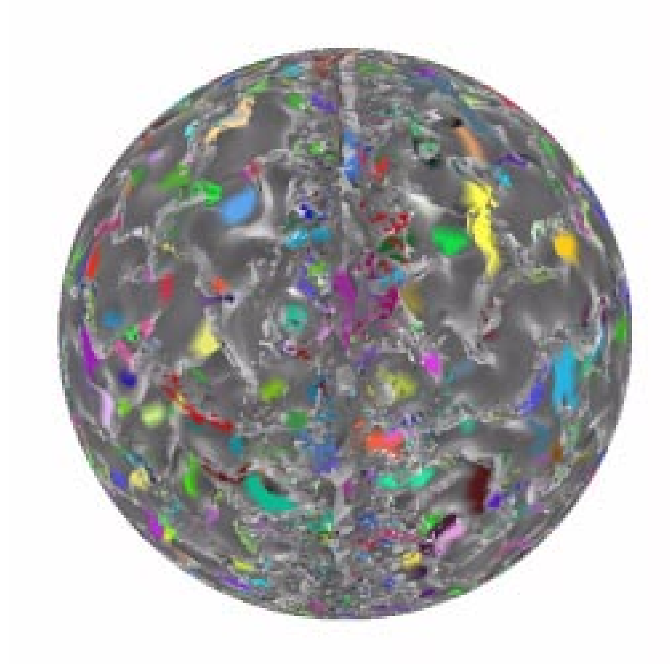

FIGURE 4.2 Top view of segmentation using gaussian curvatures projected to a sphere.

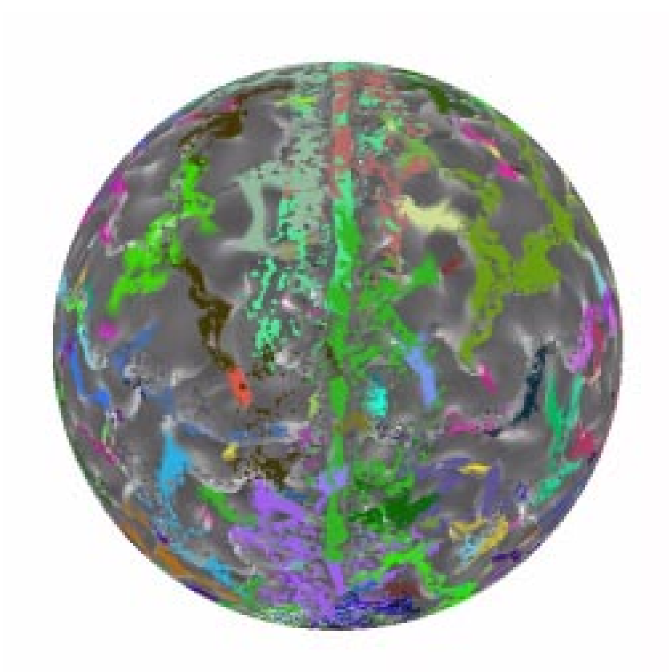

FIGURE 4.3 Top view of segmentation using absolute curvatures projected to a sphere.

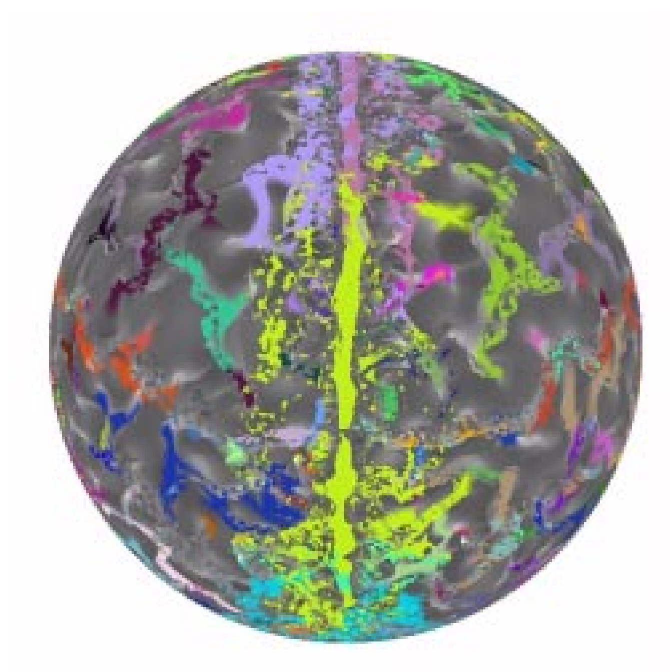

FIGURE 4.4 Top view of segmentation using root mean square curvatures projected to a sphere.