MULTIRESOLUTION MESH SEGMENTATION OF MRI BRAIN USING CLASSIFICATION AND DISCRETE CURVATURE

Sami Bourouis, Kamel Hamrouni
Ecole Nationale d'Ingénieurs de Tunis
Laboratoire de Systèmes et de Traitement du Signal : LSTS
Enit, BP-37, Le Belvédère 1002 Tunis, Tunisia

Mounir Dhibi
Ensieta E312, 2 rue Francois Verny 29806 Brest Cedex -9, France

Keywords: Brain segmentation, MRI, Statistical classification, Progressive meshes, Mesh segmentation, discrete curvatures.

Abstract: This paper presents a method for brain tissue segmentation and characterization of magnetic resonance imaging (MRI) scans. It is based on statistical classification, differential geometry, and multiresolution representation. The Expectation Maximization algorithm and k-means clustering are applied to generate an initial mask of tissue classes of data volume. Then, a hierarchical multiresolution representation is applied to simplify processing. The idea is that the low-resolution description is used to determine constraints for the segmentation at the higher resolutions. Our contribution is the design of a pipeline procedure for brain characterization/labeling by using discrete curvature and multiresolution representation. We have tested our method on several MRI data.

1 INTRODUCTION

Medical image analysis is a complex task because medical images usually involve a large amount of data. The process of generating 3D views to visualize structural information and spatial anatomic relationships is so difficult, which is usually carried out in the clinician's mind. Magnetic Resonance Imaging provides three-dimensional description of the human brain which serves as a reference for clinical investigations as well as for functional studies. Geometric modelling, mesh representation and mesh segmentation of anatomical structures in 3D images are becoming an increasingly important processing step for visualization, analysis and interpretation of the brain activity. Moreover, geometric measurement of the cortex helps in understanding brain anatomy and function. In addition, the patient brain surface labeling provides a map of the brain folds where the neuroradiologist and the neurosurgeon can easily track the features of interest.

The 3D mesh representation is widely used to represent 3D objects in various applications. It is also used to modelize and visualize complex objects and scenes, which cannot be modelled by any geometric functions. One interesting problem in analysis of surface mesh is the mesh segmentation. The problem is to cluster vertices of a mesh that are close to each other and have similar property (e.g., curvature or distance to a fitting plane) into the same group. Mesh segmentation has become an important component in the field of medical applications and computer graphics. For example, the need to precisely locate the site of functional activation within an automatic framework requires the characterization/labeling of the surface mesh.

In the last several years, many algorithms have been proposed in this growing area, offering a diversity of methods and various evaluation criteria. For example, Mangan and Whitaker (Mangan and Whitaker, 1999) proposed an approach using the watershed algorithm for partitioning 3D surface meshes. In their method, total curvature is computed and used in segmentation. Initial segments are set up from local minima of the total curvature. However, in some objects, local minima of the total curvature is not easy to find. Jaume et al. (Jaume et al., 2002) proposed an algorithm that matches progressively an atlas labeled mesh to the patient brain mesh from the largest folds to the smallest ones. Then they transfer the labels from the matched mesh to label the patient mesh.

Due to the surface meshes produced at increas-

Bourouis S., Hamrouni K. and Dhibi M. (2008).
MULTIRESOLUTION MESH SEGMENTATION OF MRI BRAIN USING CLASSIFICATION AND DISCRETE CURVATURE.
In *Proceedings of the Third International Conference on Computer Vision Theory and Applications*, pages 421-426
DOI: 10.5220/0001078704210426

ing complexity, the need for an accurate simplification of surface meshes is becoming more and more urgent. This need is not only due to rendering speed reasons, but also to allow fast transmission of 3D models in network-based applications. Indeed, efficient simplification algorithms may optimize initial meshes and replace them by nearly indistinguishable approximations and can hope to automate these tasks. In recent years, the problem of surface simplification has received increasing attention. Several different algorithms have been formulated for simplifying surfaces. Schroeder et al. (Schroeder et al., 1992) proposed an algorithm ”decimation of triangle meshes” to reduce the number of triangles in a triangle mesh, while preserving the original topology and forming good approximation to the original geometry. Another algorithm has been proposed by Rossignac and Borrel (Rossignac and Borrel, 1993). It is one of the few works which is able to process arbitrary polygonal input. Others algorithms have been published that simplify models by iteratively contracting edges. The essential difference between these algorithms lies in how they choose an edge to contract. Some notable examples of such algorithms are those of Hoppe (Hoppe, 1996) and Ronfard et al. (Ronfard and Rossignac, 1996).

This work is initially motivated by the need to describe and characterize 3D data in the analysis of functional imaging data. We propose in this paper a pipeline steps for the problem of 3D mesh segmentation of the brain in magnetic resonance images. Our algorithm uses three stages to segment MRI image. Initially, data is classified into three main classes in order to extract white matter (WM), gray matter (GM), cerebral-spinal fluid (CSF). Then, we generate a discrete representation of each tissue class and we create a mutiresolution representation for each one. In the third stage, we segment the obtained meshes using a criteria based on discrete curvature. The proposed framework is presented in figure 1.

2 DISCRETE REPRESENTATION OF BRAIN TISSUES

The proposed framework uses a combination of statistical classification, isosurface extraction, multiresolution representation and brain labeling. These steps address various problems in computer graphics, scientific visualization, analysis and interpretation of the brain activity and many others. In the next sections we explain in details our method and present some results.

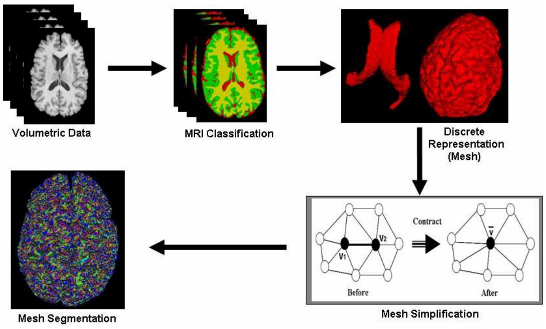

Figure 1: The general principle of the method.

2.1 Expectation Maximization

In statistical classification, each object is represented by $\boldsymbol{d}$ features, and the goal of classification becomes finding compact and disjoint regions for classes in a $\boldsymbol{d}$-dimensional feature space. Such decision regions are defined by decision rules that are known or can be trained. In the case of unsupervised classification, the Expectation Maximization (EM) algorithm (Dempster et al., 1977), is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate in the presence of hidden data $\boldsymbol{f}$. In short, the EM alternates between two steps: an expectation (E) step and a maximization (M) step. In the E-step, the missing data are estimated given the observed data and current estimate of the model parameters. In the M-step, we compute the maximum likelihood estimates of the parameters by maximizing the expected likelihood found in the E-step. The parameters found in the M-step are then used to begin another E-step, and the process is repeated.

Let us consider the mean parameter μ_k and the variance parameter σ_k of the intensity distribution of the k-th tissue class grouped in θ_k such as $\theta_k = \{\mu_k, \sigma_k\}$. We denote also π^k the prior probability of each class k and γ_i^k the posteriori probability calculated in each voxel $\boldsymbol{i}$ for each class k. In the expectation step, we calculate the posteriori probability according to the following formulas:

$$\gamma_i^k = P(x_k/y_k, \theta_k) = \frac{\pi_i^k f_k(y_i/\theta_k)}{\sum_{l=1}^K \pi_i^l f_l(y_i/\theta_l)} \qquad (1)$$

In the maximization step, we estimate data driven parameters by:

$$\begin{cases} \pi_k = \frac{\sum_{l=1}^K \gamma_i^k}{N} \\ \mu_k = \frac{\sum_{l=1}^K \gamma_i^k y_i}{\sum_{l=1}^K \gamma_i^k} \\ \sigma_k^2 = \frac{\sum_{l=1}^K \gamma_i^k (y_i - \mu_k)(y_i - \mu_k)}{\sum_{l=1}^K \gamma_i^k} \end{cases} \qquad (2)$$

2.2 K-means Clustering

K-means clustering is a popular clustering algorithm because it is simple and usually converges to a reasonable solution. Here, we apply the k-means algorithm in order to define classes that represent statistical distributions of intensity values in the voxels and to extract different tissues. K-means works as follows:

- Obtaining the initial k means input,
- Assigning each measurement vector in a sample container to its closest mean among the k number of means,
- Calculating each cluster's mean from the newly assigned measurement vectors (updates the centroid (mean) of k clusters),
- Repeating step 2 and step 3 until it meets the termination criteria.

In the context of Medical Imaging, each class is typically associated to a particular type of tissue and can therefore be used as a form of image segmentation. Fig.2 illustrates the classification of an MRI data into three classes (k=3): GM, WM and CSF. Parameters are estimated using EM.

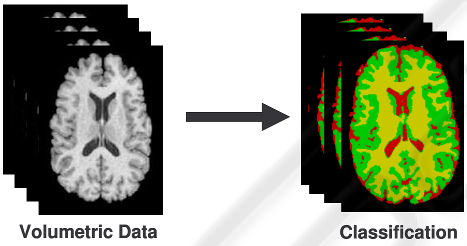

Figure 2: k-means clustering.

2.3 Generating Mesh

Representing the surface as explicit geometry is efficient when used with the conventional computer graphics approaches for shading and viewing. Further, it greatly reduces the necessary data storage and provides a data structure that can be measured. In our case, the initial volumetric MRI data is transformed into a triangulation using a standard isosurface extraction algorithm -Marching Cubes- (Lorensen and Cline, 1987). Marching cubes provides an accurate three-dimensional polygonal representation that can be used for other image processing. A cube is bounded by eight pixels located in two adjacent slices. Each vertex is coded as either inside or outside the object relative to the surface-defining threshold. Based on the configuration of vertices that lie inside and outside the object, the cube is triangulated. The triangles indicate where the surface passes through the cube.

However, the output from the marching cubes usually contains multiple small ”useless” meshes. Since these meshes are physically disconnected from each other. In order to reduce aliasing artifacts in images, we use the algorithm proposed in (Whitaker, 2000). Briefly, the algorithm is as follows: The binary volume is used as a set of constraints in an iterative relaxation process of an initial estimate of the surface. The surface is allowed to deform under curvature flow. A set of constraints is introduced in its movement which fits the surface to the volume in such a way as to minimize surface area, producing a smooth estimation.

3 MESH SEGMENTATION

A surface's behavior can be described by dividing the surface into distinct regions of elliptic behavior and hyperbolic behavior. The regions of elliptic behavior can be classified into convex and concave regions by considering the direction of the surface normal.

3.1 Mesh Simplification

Marching Cubes is very efficient but usually produces more than the necessary number of polygons needed to represent an object accurately. It records all details associated with the original data regardless of whether these details are insignificant. The result contains an enormous number of extremely small triangles that prevent interactive render of models. Therefore, the problem is how to create a new set of surfaces, which are less complex than the original one without modifying the overall geometric information. To solve this problem, we have resorted to mesh simplification techniques. Indeed, if we segment the surface at a high resolution, the segmentation will be too complex. A way to accelerate this process and preserve topology is to create a low-resolution representation of the model by using the PM algorithm. Thus, mesh segmentation at a low resolution defines constraints for more complex features detected at the higher resolution.

The PM algorithm (Hoppe, 1996) is attractive because it simplifies a model by minimizing an energy function. Indeed, the method provides four visual benefits. It uses many polygons to represent areas of high curvature. It aligns long edges along directions of low curvature. It adjusts the aspect ratios of triangles to match the local curvature. Finally, it places edges and vertices near sharp features of the original model. An iteration of this algorithm uses an edge collapse operation. This operation (Fig. 3) selects an edge and replaces it with a single vertex. This removes one vertex, three edges, and two faces. The edge collapse operation is attractive because it allows the new vertex to be placed in a manner that helps preserving the location and the shape of the original surface.

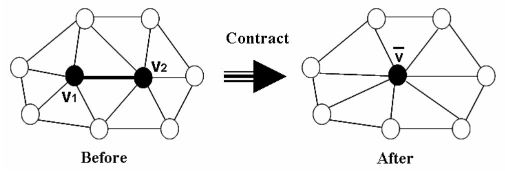

Figure 3: Edge collapse operation .

3.2 Curvature-based Segmentation

In surface analysis, we compute gaussian curvature and mean curvature, and later use them to classify the surface type of vertices. 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.

many approximation schemes for curvature estimates meshes have been developed such as (Max, 1999; Hamman, 1993). We apply here some operators, which are derived recently by Meyer et al. (Meyer et al., 2002), to estimate curvature information at each vertex in the mesh. We denote k_H and k_G, the Mean and the Gaussian curvature operators. In our algorithm, we calculate these operators according to the following formulas:

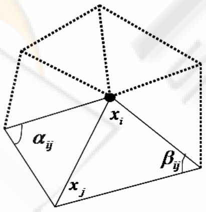

Figure 4: 1-ring neighbors and angles opposite to an edge.

$$\begin{cases} k_H = \frac{1}{2A}\sum_{j\in N(i)}(\cot(\alpha_{ij})+\cot(\beta_{ij}))(x_i - x_j) \\ k_G = \frac{2\pi-\sum_{j=1}^{\sharp f}\Theta_j}{A} \end{cases} \quad (3)$$

Where α_{ij} and β_{ij} are the two angles opposite to the edge in the two triangles sharing the edge (x_i, x_j) as depicted in figure. 4. Θ_j is the angle of the j-th face at the vertex x_i and $\sharp f$ denotes the number of faces around this vertex.

Table 1: Possible combinations of surface types according to discrete curvature criteria.

$Mean(x_i)$	convex	convex	concave	concave
$Gauss(x_i)$	hyperb	elliptic	hyperb	elliptic
Color	green	yellow	blue	red

If the value of k_G is negative, then we have a hyperbolic behaviour, otherwise it is an elliptic one; then, the regions of elliptic or hyperbolic behaviour is convex if the k_H value is negative, otherwise it is concave. This mechanism is illustrated in the figure 5.

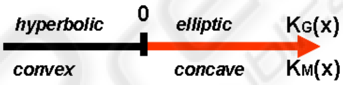

Figure 5: Curvature-based criteria for surface characterization.

4 EXPERIMENTAL RESULTS

We have performed a series of experiments on brain MR images. Resulting segmentations are depicted in Figures 6, 7 and 8. Each label corresponds to a different color. The two curvatures used are mean and gaussian. By combining the operators k_H and k_G, we can encode surface types with different color according to table 1. In our case, this leads to four cases (table 1).

Our segmentation consists of regions of the same type of curvature according to Table 1 by encoding surface types with colors. Figure 6 shows the labeling of the cortical surface into elliptical convex regions (yellow), elliptical concave regions (red), hyperbolic convex regions (green), and hyperbolic concave regions (blue). These segmentations are performed in the case of multiresolution representation as shown in figures 7 and 8). An inherent difficulty in the interpretation of these results is that there is no definition of what is correct. Some papers use expert opinion from neurologists on where sulci and gyri exist. However, visual results can be interpreted given the idea that gyri are the surfaces top of the brain folds (ridges), and that sulci are the surfaces within the brain folds (basins). For example, it will be easy to distinguish between elliptic and hyperbolic forms. The final labels resulting from our segmentation are depicted graphically in figures 7 and 8. Each label corresponds to a different color in all obtained resolutions. According to these results, we notice that segmentation at low-resolution (10% of the original data) leads to an extraction of the major surface features/characteristics, whereas when the algorithm is applied to the surface at high resolution (100%), it does not necessary produces more relevant structure information but includes small features, which do not contribute very much to the overall shape. In addition, low-resolution description can be used to determine constraints for the segmentation at higher resolution. For instance, these first results are encouraging but further investigation is required to extend the algorithm to a large range of data.

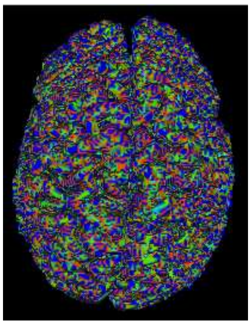

Figure 6: Segmentation of the cortical surface according to table 1.

5 CONCLUSIONS

In this paper, an automated approach for brain MRI segmentation and discrete mesh characterization is proposed. We have currently explored and outlined the importance of multiresolution representation to simplify processing, segment meshes and accelerate medical analysis. Our technique has a great interest in the study of structural and functional characteristics of the brain. It is also relatively computationally efficient. At this stage, we have only applied the approach to a few experimental cases and we have presented some preliminary results to demonstrate its potential: the method gives satisfying results for mesh labelling in the case of multiresolution representation. Even if they have not yet been compared to manual or other automatic segmentation results, we think that they are encouraging and faster than manual procedures. However, there are some future works to do. Clinical validation remains to be done, which will require additional work. Future validations will compare our segmentation with manually labelled data and other segmentation results. Finally, the same framework can be used and extended to segment and quantify abnormal brains.

REFERENCES

Dempster, A., Laird, N. M., and Rubin, D. (1977). Maximum likelihood from in complete data via the em algorithm. *Journal of the Royal Statistical Society*, 1:1–38.

Hamman, B. (1993). Curvature approximation for triangulated surfaces. *In: Gerald Farin et al., editors, Geometric Modelling, Computing Supplementum, Springer-Verlag, Vienna, Austria*, pages 139–153.

Hoppe, H. (1996). Progressive meshes. *SIGGRAPH '96 Proc*, pages 99–108.

Jaume, S., Macq, B., and Warfield, S. (2002). Labeling the brain surface using a deformable multiresolution mesh. *MICCAI*, pages 451–458.

Lorensen, W. and Cline, H. (1987). Marching cubes: A high-resolution 3-d surface construction algorithm. *ACM Comput. Graph*, 21:163–170.

Mangan, A. and Whitaker, R. (1999). Partitioning 3d surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 5.

Max, N. (1999). Weights for computing vertex normals from facet normals. *Journal of Graphics Tools*, 4(2):1–6.

Meyer, M., Desbrun, M., and Schroder, P. (2002). Discrete differential-geometry operators for triangulated 2-manifolds. *International Workshop on Visualization and Mathematics, Berlin, Germany,.*

Ronfard, R. and Rossignac, J. (1996). Full-range approximation of triangulated polyhedra. *Computer Graphics Forum, Proc. Eurographics '96*, 15:1–38.

Rossignac, J. and Borrel, P. (1993). Multi-resolution 3d approximations for rendering complex scenes. In Falcidieno, B. and Kunii, T., editors, *Theoretical Aspects of Computer Software*, Modeling in Computer Graphics: Methods and Applications, pages 455–465.

Schroeder, W., Zarge, J., and Lorensen, W. (July 1992). Decimation of triangle meshes. *Computer Graphics (SIGGRAPH '92 Proc.)*, 26:65–70.

Whitaker, R. (2000). Reducing aliasing artifacts in isosurfaces of binary volumes. *IEEE Volume Visualization and Graphics Symposium*, 21:23–32.

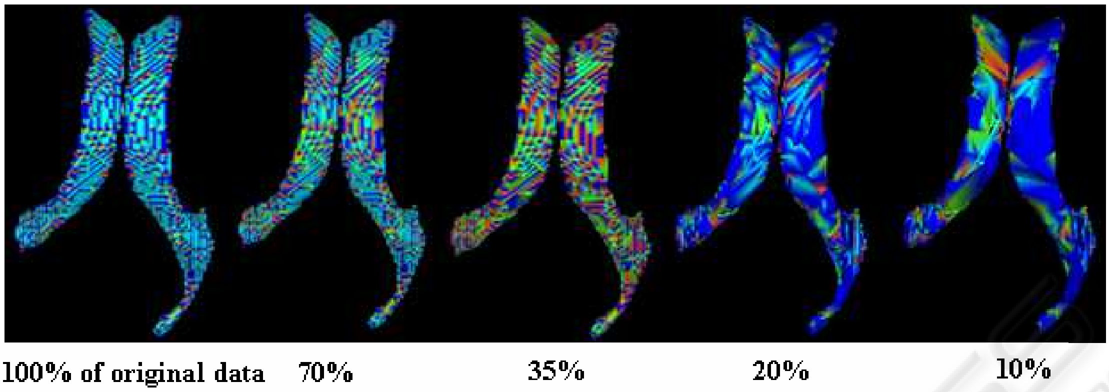

Figure 7: Segmentation of ventricle surface at different resolutions according to table 1.

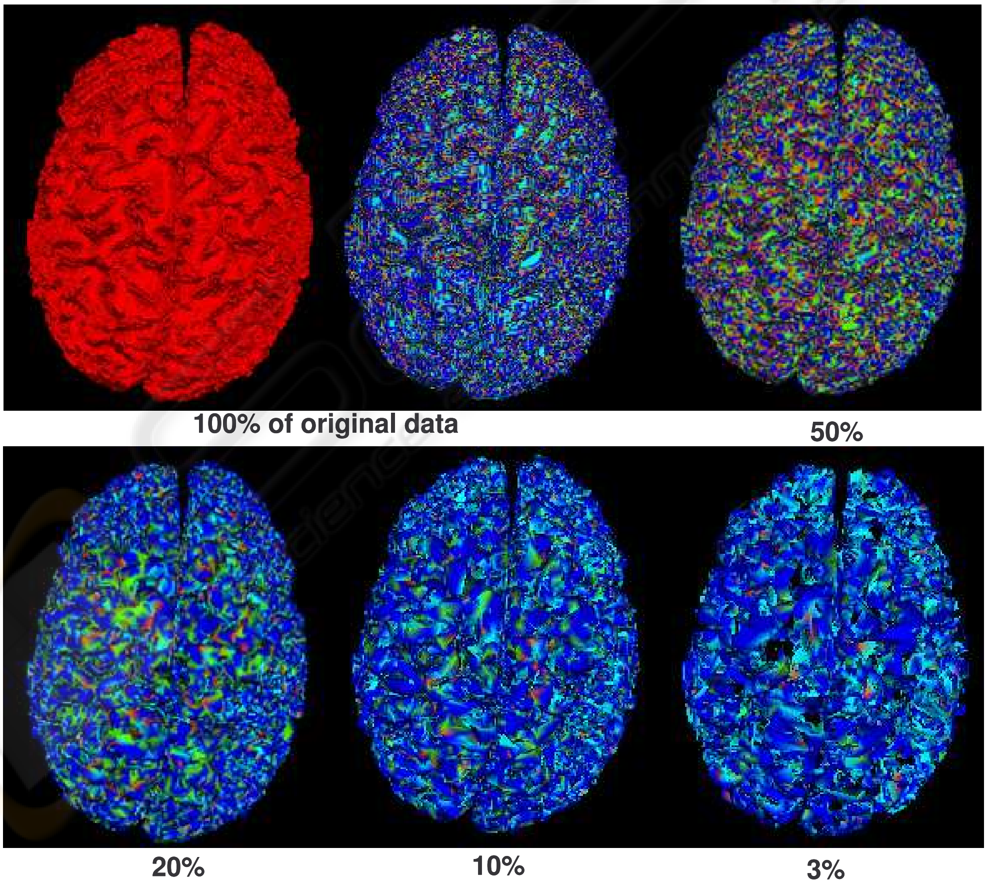

Figure 8: Segmentation of cortical surface at different resolutions according to table 1.