Topological Correction of Subcortical Segmentation

Florent Ségonne[1,2], Eric Grimson[1], and Bruce Fischl[1,2]

[1] MIT AI Lab
[2] Athinoula A. Martinos Center - MGH/NMR Center

Abstract. We propose a method for automatically correcting the spherical topology of any segmentation under any digital connectivity. A multiple region growing process, concurrently acting on the foreground and the background, divides the segmentation into connected components and successive minimum cost decisions guarantee convergence to correct spherical topology. In contrast to existing procedures that suppose specific initial segmentation (full connectivity, no cavities...) and are designed for a particular task (cortical representation), no assumption is made on the initial image. Our method applied to subcortical segmentations allows us to correct the topology of fourteen non-cortical structures in less than a minute.

1 Introduction

Excluding pathological cases, most macroscopic brain structures are fully connected and do not possess any topological artifact such as handles or cavities: they have the simple topology of a sphere. Many recent segmentation algorithms are able to identify and precisely locate these structures, although without constraining the topology. Being able to achieve accurate and topologically correct representations of different brain structures is certainly an important goal in medical imaging (shape analysis, visualization . . .)

Only a few automatic techniques have been proposed to produce topologically correct segmentations. Several approaches have tried to directly incorporate topological constraints into the segmentation process [1,2,3,4,5]. An initial region, carrying the correct topology, is usually deformed by addition/deletion of points by minimizing a global energy function while preserving the correct topology. The problem with these methods is that local topological constraints can lead to strong geometrical errors, and that the final segmentation can strongly depend on the order in which points are added.

Recently, new approaches have been developed to retrospectively correct the topology of a segmented image. These methods can be divided in two main classes: volume-based methods that work directly on the volume lattice and correct the topology by addition/deletion of voxels, and surface-based methods that aim at modifying the tessellation by locating and cutting handles/filling holes. Most volume-based approaches have been specifically designed to correct the topology of the cortical surface. Shattuck and Leahy examine the connectivity

R.E. Ellis and T.M. Peters (Eds.): MICCAI 2003, LNCS 2879, pp. 695–702, 2003.

of the segmentation to detect topological defects and minimally correct them [6]. Inspired by their work, Han et al. developed an algorithm to remove all handles from a binary object under any digital connectivity [7]. Successive morphological openings correct the segmentation at the smallest scales. Kriegeskorte and Goeble use a region growing method prioritized by the distance-to-surface of the voxels in order to force the cuts to be located at the thinnest part of each topological defect [8]. The same process is applied to the inverse object, offering an alternative solution to each cut. An empirical cost is then assigned to each solution and the final decision is the one minimizing the global cost function.
Other types of approaches operate directly on the triangulated surface mesh. Fischl et al. proposed an automatic procedure to locate and correct topological defects by homeomorphically mapping the initial triangulation onto a sphere [9]. Topological defects are located as non-homeomorphic regions, constituted of overlapping triangles. A greedy algorithm is then used to retessellate incorrect patches, constraining the topology on the sphere while preserving geometrical accuracy by maximum likelihood optimization. Another approach is proposed in [10]. Handles in the tessellation are localized by simulating wavefront propagation on the tessellation: they are detected where wavefronts meet. One potential drawback of this method is that it depends on the vertex used to identify the defect.
While these methods can be effective, they cannot be used to correct the topology of arbitrary segmentations, as they make assumptions about the topology of the initial binary image. Most frequently, fully-connected volumes are assumed and cavities are supposed to be removed as a preprocessing step. In the subcortical segmentation topology problem, the modification of a small number of voxels per structure is usually sufficient to correct them. However, due to the presence of imaging artifact, anatomical variability, varying contrast properties and poor registration, no assumptions can be made about the initial topology of the segmentation.

In this paper, we develop a completely automated volume-based method to correct the topology of any segmentation under any digital connectivity. The novelty of our approach comes from the fact that any initial segmentation (disconnected regions, handles, cavities, holes ...) will still be corrected. At each step of our iterative topological correction, minimum cost decisions are taken and convergence is guaranteed.

2 Background and Definitions

In this section, some basic notions of digital topology are presented. We refer to the work of G. Bertrand for more details [11]. The initial segmentation is a binary digital image $I \subset Z^3$ composed of a foreground object F and an inverse background object $B = \overline{F}$. Following the conventional definition of adjacency, 3 types of connectivity might be considered: 6-, 18- and 26-connectivity. For instance, two voxels are 6-adjacent if they share a face, 18-adjacent if they share at least an edge and 26-adjacent if they share at least a corner. In order to avoid topological paradoxes, different connectivities, n and $\overline{n}$, must be used for F and B. This leaves us with four pairs of compatible connectivities: (6,26), (6,18), (26,6) and (18,6). Considering a digital object, the computation of two numbers are sufficient to check if the modification of one single point will affect the topology. These *topological numbers*, denoted T_n and $T_{\overline{n}}$, have been introduced by G. Bertrand in [11,12] as an elegant way to classify the topology type of a given voxel. We will implicitly use them during the algorithm. In the next sections, we denote by X the object on which we are currently working, and $\overline{X}$ its inverse object. $C_n(X)$ is the set of all n-connected components in X.

we will need the following definitions:

Simple point: a point $x \in X$ that can be added or removed without changing the topology of an object; they are characterized by $T_n(x, X) = T_{\overline{n}}(x, \overline{X}) = 1$.

Isolated point: a point $x \in X$ that is not connected to any other point of X: they are characterized by $T_n(x, X) = 0$.

Residual and body labels: during the algorithm, different components are generated, and voxels are assigned different labels. Body labels characterize voxels belonging to a body component with a known topology, and residual labels characterize voxels belonging to a component with an unknown topology.

Seed point: a residual point of X that is simple or isolated relatively to the body label points of X. Under this definition, changing the residual label of a seed point to body will not introduce any topological defect into the body component segmentation of X.

Multisimple point: a residual point $x \in X$ that can be added to any of its adjacent body components ($\in C_n(X)$) without changing its topology. Equivalently, a multisimple point is a residual point that is simple relatively to each adjacent body component, independently of all the others. Therefore, the merging of a multisimple point into one of the adjacent body components (usually the largest one), associated with the merging of the other adjacent body components into the first one, will not change the topology of the new component.

Multiseed point: either a seed point or a multisimple point of X.

3 Method

Our method aims at correcting the topology of any binary segmented volume under a set of compatible connectivities (n,$\overline{n}$). Any segmentation technique producing binary volumetric images can be used to generate an initial input to our algorithm.

Similarly to the approach described in [8], the distance-to-surface map, representing for each voxel its distance to the surface (Fig. 1b), is used to drive a multiple region growing process that segments each object into a set of connected components (Sect. 3.1). A cost is then assigned to each component and the algorithm iteratively decides to delete the lowest cost component(s); at each iteration, the set of connected components is updated, and convergence is guaranteed by monotone increase of a threshold (Sect. 3.2). During the different steps of the algorithm, we assume that each voxel carries a cost, reflecting its cost of being modified. Discussion of different costs will be done in section 4.

3.1 Segmentation into Connected Components

The first step of our algorithm is the segmentation of each object into a set of body and residual connected components: every point is initially assigned a residual label and body components are slowly expanded outward to incorporate new simple points.

Foreground Object: we remind the reader that a seed point (see Sect. 2) will not introduce any topological defect into any of the *body* components, therefore allowing us to start growing a new body component without introducing any topological artifact.

Every voxel of the foreground object is first assigned a residual label, except the deepest one that is assigned a body label. This seed point, representing the first body component, is then iteratively dilated by adding adjacent simple points, prioritized by the distance map: adjacent voxels are checked in decreasing order (the deepest first) and added if they are simple. When no residual voxel can be added to this body component without changing its topology, the algorithm tries to grow another component by searching for the next deepest seed point. We keep generating and growing new body components until no new seed point is found. Then, the remaining residual voxels are segmented into residual connected components and a cost is assigned to all components: the cost of each component is simply defined as the sum of the costs of all voxels constituting this component.

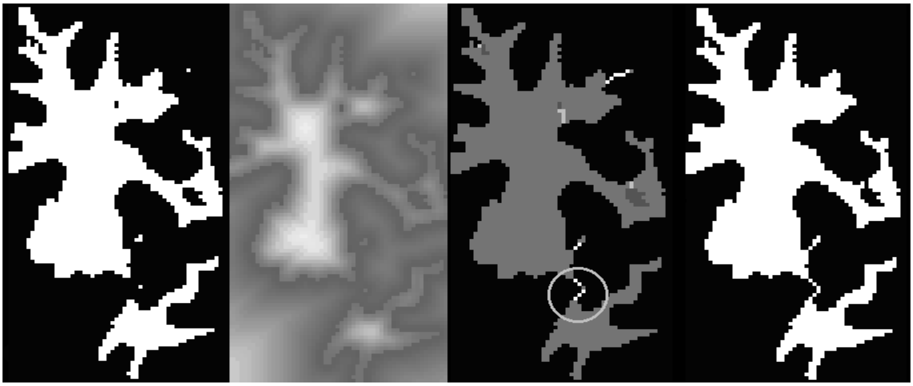

Fig. 1. An initial binary image, its corresponding distance map, the segmentation into connected components and the final corrected image under (26,6). c) Residual and body components are bright and dark respectively. The unique black component is a body background component with the topology of a hollow sphere. d) The algorithm iteratively corrects the topology of the binary image, by adding or deleting components depending on their cost: the final result shows that certain components were preferentially removed from the foreground, and respectively for the background. The topological corrections are located at the thinnest part of the volume.

Background Object: the same multiple region growing process is applied to the inverse object. However, since we are working on the background object that is supposed to surround completely the foreground object, the first body component is composed of the set of voxels located at the border of the image. Therefore,the topology of the first background component will logically be the one of a hollow sphere. Then the algorithm proceeds as previously described.

The use of the distance map to drive the expansion process causes the residual components to be located at the thinnest part of the volume (see Fig. 1c, [8]). Different prioritizations, for instance based on prior or posterior probabilities, might be used in order to better control the location of the components. However, topologically constrained expansions are noise sensitive (an incorrect local topological decision could lead to large geometric errors), and the prioritization must be reliable. In this paper, we use the distance map and postpone the study of different prioritizations to future work.

3.2 Correction of the Topology

We are now ready to start correcting the topology. The goal is to successively decrease the number of residual and body components, until one single component per object remains: a foreground component with a spherical topology and a background component with the topology of a hollow sphere. The algorithm proceeds iteratively, by identifying at each step the lowest cost component and deleting it, adding its constituting voxels to the inverse object. Then the algorithm modifies the set of components by resuming the region growing process.

Identification of Lowest Cost Components: our method simultaneously works on the background and the foreground. Assuming a decomposition into connected components, the algorithm identifies the lowest cost component. Possibly, several components might have the same lowest cost and two cases have to be considered. If all of them belong to the same set $C_n(X)$, then all these components are kept for the next step. In the other case, the user has the option of prioritizing one object (F or B). if the priority is given to B, then only the components belonging to $C_{\overline{n}}(B)$ are kept: by deciding to work on the background first, the algorithm will fill holes and merge disconnected foreground regions before cutting handles or deleting regions of the same cost. Conversely, the priority can be given to the foreground object.

Deletion of a Set of Components: once a set of components has been identified, we turn all its constituting voxels into residual points of the inverse object and resume the region growing process:

```
    Algorithm 1
1. Search among multiseed points in X (see Sect 2).
2. If no multiseed point is found, then Stop. Else go to step 3.
3. If the multiseed point is isolated, generates a new body component.
   If not, merge this point into the largest adjacent body component,
   and merge the other adjacent body components into this component.
4. Update the cost of the modified components.
5. Go to step 1.
```

Multiseed points allow us to locally modify the component segmentation: body connected components might fusion together, but the topology is preserved. We note that local decisions do not imply large geometrical errors. For instance, in Figure 1, the deletion of the circled foreground body component does not lead to the removal of any large component.

Convergence: stated as previously, the algorithm is not guaranteed to converge: the deletion of a set of components might lead to the creation of even lower cost components in the inverse object. Therefore, we use a threshold that is monotonically increased at each iteration.

```
    Algorithm 2
0. Set threshold $= -\infty$.
1. Find the set of lowest cost components $\{C_i\} \in C_n(X)$ such that:
   $\exists C_j \in \{C_i\} / cost(C_j) > threshold$; set threshold $\leftarrow cost(C_j)$.
2. Delete the components: $\forall x \in \cup C_i$, set $x \rightarrow \overline{X}$
3. If $X$ has one single component then stop. Else go to step 4.
4. Apply Algorithm 1 to the inverse object $\overline{X}$ and go to step 1.
```

We note that, similarly to [7], this algorithm can be modified to force corrections to be made on one single object: it suffices to constrain the search for lowest cost components to the inverse object.

3.3 Post-processing

Once the correct topology has been achieved, the algorithm tries to add back to the foreground object every background voxel that was initially part of the foreground. All these ambiguous voxels are labeled as residual of F and we simply apply a conditional topological expansion of the object F similar to the ones described in Section 3.1. The same process is applied to the inverse object $B = \overline{F}$ with foreground voxels that were initially part of B.

4 Results and Discussion

Our goal, when implementing this algorithm, was to develop a fully automated method that is able to correct the topology of already accurate subcortical segmentations, without any further assumptions on the initial segmentations. In order to validate the proposed algorithm, we have applied our method to 25 brain subcortical segmentations composed of 14 nuclei: left and right ventricle, putamen, pallidum, amygdala, hippocampus, thalamus, caudate nucleus (see Fig. 2). Before presenting some results, we discuss the cost function used in the algorithm.

During the topology correction, each voxel is assumed to carry a cost of being modified. Different cost options are available to the user. Without any more information than the initial segmentation, the user might minimize the number of modified voxels at each step, therefore assigning a constant positive cost to each voxel. However, some external information, such as the posterior probability of being part of the foreground or the background at location x given some variables $V(x)$ (intensity, local curvature...) $p = p(F(x)|V(x))$ and $\overline{p} = p(B(x)|V(x))$, might be available. In this case, a more natural cost would be to assign $ln(p/\overline{p})$

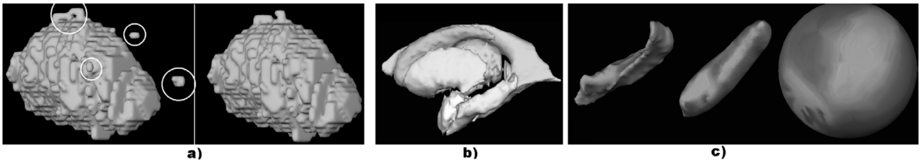

Fig. 2. a) Right thalamus segmentation before and after topology correction. Some topological defects are circled. b) Result of a subcortical segmentation with 14 nuclei: left and right ventricle, putamen, pallidum, amygdala, hippocampus, thalamus, caudate nucleus. The segmentation was obtained with the algorithm described in [13]. Every structure was then topology corrected and a rendering algorithm [14] was used to consistently generate the view. c) Right hippocampus segmentation. Topologically correct volumes can be used to generate spherical atlases by inflating and projecting the surfaces onto a sphere.

to each foreground voxel and $ln(\overline{p}/p)$ to each background voxel, resulting in an algorithm maximizing the MAP estimate at each iteration; also, as previously noted in Sect. 3.1, a prioritization based on each voxel's cost can be used to drive the expansion process, locating the residual components at low cost locations. The use of reliable probability maps can significantly improve the topological correction (MAP), but can lead to large geometric errors if the probabilities are inaccurate. The systematic use of probability maps and their consequences is left for future work. In the following results, the distance map has been used and a constant positive cost has been assigned to each voxel.

We have applied our method to 25 brain datasets, manually and automatically labeled. Addition and deletion of very few voxels is necessary to correct each structure topology (of the order of 0.05% for the manual segmentations and 0.1% for the automatic segmentation described in [13]). Accuracy is achieved through minimal corrections of supposedly precise initial segmentations; inaccurate segmentations would still be corrected but the location of topological corrections at the thinnest parts of the volume could not guarantee the final accuracy. We note that our method, working independently on each nucleus, might cause some voxels to have more than one label. Results show that this problem concerns less than 0.01% of the voxels. Applied to the white matter correction, this algorithm leads to results visually similar to the ones presented in [7].

Most of the computational time is taken by the region growing process (Sect. 3.1), which has linear time complexity (see [8]): each structure is corrected in a few seconds and a whole subcortical topology correction takes less than a minute on a current machine.

Finally, we note that this algorithm, associated with an accurate preprocessing segmentation technique, can provide precise topologically correct initial images to hybrid techniques that incorporate topological constraints into the segmentation process [2]: strong geometrical errors, often resulting from local topological constraints, would be avoided by the accuracy of the initial labeling.

5 Conclusion

We have presented a novel algorithm, achieving spherical topology correction under any kind of digital connectivity and accepting any initial segmentation. Topological defects are located at the thinnest part of the volume and minimal corrections iteratively rectify the topology. Similarly to the method of Han et al. [7], our algorithm can enforce background or foreground topological corrections exclusively. Applied to subcortical segmentations, the topology of fourteen deep nuclei is corrected in less than a minute.

References

1. J.-F. Mangin, V. Frouin, I. Bloch, J. Regis, and J. Lopez-Krahe, “From 3D magnetic resonance images to structural representations of the cortex topography using topology preserving deformations,” Journal of Mathematical Imaging and Vision, vol. 5, pp.297–318, 1995.
2. F. Poupon and J.-F. Mangin and D. Hasboun and C. Poupon and I. Magnin and V. Frouin, “Multi-object Deformable Templates Dedicated to the Segmentation of Brain Deep Structures”, Lecture Notes in Computer Science, vol. 1496, pp. 1134–1143, 1998.
3. S. Bischoff, and L. Kobbelt, “Isosurface Reconstruction with Topology Control,” Pacific Graphics 2002 Proceedings,pp. 246–255.
4. D. MacDonald, “A Method for Identifying Geometrically Simple Surfaces from Three Dimensional Images”, in Montreal Neurological Institute. 1998, McGill University: Montreal.
5. Davatzikos, C. and R.N. Bryan, “Using a Deformable Surface Model to Obtain a Shape Representation of the Cortex”. IEEE Trans. on Medical Imaging, 1996. 15: p. 785–795.
6. D. W. Shattuck and R. Leahy, “Automated Graph-Based Analysis and Correction of Cortical Volume Topology,” NeuroImage, vol. 14, pp. 329–346, 2001.
7. X. Han, C. Xu, U. Braga-Neto, and J. L. Prince, ”Topology correction in brain cortex segmentation using a multiscale, graph-based approach,” IEEE Trans. Med. Imag., vol. 21(2): 109–121, 2002.
8. N. Kriegeskorte and R. Goeble, “An efficient algorithm for topologically segmentation of the cortical sheet in anatomical mr volumes. NeuroImage, vol. 14, pp. 329–346, 2001.
9. B. Fischl, A. Liu, and A. M. Dale, “Automated manifold surgery: Constructing geometrically accurate and topologically correct models of the human cerebral cortex,” IEEE Trans. Med. Imag., vol. 20, pp. 70–80, 2001.
10. I. Guskov and Z. Wood, “Topological noise removal,” GI 2001 proceedings, pp. 19–26, 2001.
11. G. Bertrand, “Simple Points, topological numbers and geodesic meighborhood in cubid grids,” Pattern Recognition Letters, vol. 15, pp. 1003–1011, 1994.
12. G. Bertrand, “A boolean characterization of three-dimensional simple points,” Pattern recognition letters, vol. 17, pp.115–124, 1996.
13. B. Fischl et al. “Whole Brain Segmentation: Automated Labeling of Neuroanatomical Structures in the Human Brain”. Neuron, vol. 33,pp. 341–355, 2002.
14. B. Natarajan, “On generating topologically consistent isosurfaces from uniform samples,” The Visual Computer, vol. 11(1), pp. 52–62,1994.