NeuroImage **14,** 329–346 (2001)
doi:10.1006/nimg.2001.0831, available online at http://www.idealibrary.com on IDEAL®

An Efficient Algorithm for Topologically Correct Segmentation of the Cortical Sheet in Anatomical MR Volumes

Nikolaus Kriegeskorte and Rainer Goebel

Faculty of Psychology, Department of Cognitive Neuroscience, Universiteit Maastricht, Universiteitssingel 40, 6229 ET Maastricht, The Netherlands

Received October 2, 2000

Polygon-mesh representations of the cortices of individual subjects are of anatomical interest, aid visualization of functional imaging data and provide important constraints for their statistical analysis. Due to noise and partial volume sampling, however, conventional segmentation methods rarely yield a voxel object whose outer boundary represents the folded cortical sheet without topological errors. These errors, called handles, have particularly deleterious effects when the polygon mesh constructed from the segmented voxel representation is inflated or flattened. So far handles had to be removed by cumbersome manual editing, or the computationally more expensive method of reconstruction by morphing had to be used, incorporating the *a priori* constraint of simple topology into the polygon-mesh model. Here we describe a linear time complexity algorithm that automatically detects and removes handles in presegmentations of the cortex obtained by conventional methods. The algorithm's modifications reflect the true structure of the cortical sheet. The core component of our method is a region growing process that starts deep inside the object, is prioritized by the distance-to-surface of the voxels considered for inclusion and is selftouching-sensitive, i.e., voxels whose inclusion would add a handle are never included. The result is a binary voxel object identical to the initial object except for "cuts" located in the thinnest part of each handle. By applying the same method to the inverse object, an alternative set of solutions is determined, correcting the errors by addition instead of deletion of voxels. For each handle separately, the solution more consistent with the intensities of the original anatomical MR scan is chosen. The accuracy of the resulting polygon-mesh reconstructions has been validated by visual inspection, by quantitative comparison to an expert's manual corrections, and by crossvalidation between reconstructions from different scans of the same subject's cortex. © 2001 Academic Press

Key Words: **surface reconstruction; cortex; cortical sheet; MRI; fMRI; topology; handle; genus.**

INTRODUCTION

The human cerebral cortex is a convoluted sheet of varying thickness whose two-dimensional macrostructure early anatomists have recognized long before its layered-microstructure—its third dimension—was discovered. Because of its 2-D macrostructure, the cortical sheet lends itself to representation in a single planar map for each hemisphere. Such maps were first produced manually from postmortem brains by rather coarse techniques. Today a representation of the cortex of a living subject can be unfolded, flattened, and used as a map by computational methods. To this end, the cortical sheet is reconstructed as a polygon mesh from anatomical magnetic resonance (MR) imaging data (Carman *et al.*, 1995). These techniques are useful not only to the neuroanatomist studying the physical structure of the cortex, but also to neuroscientists primarily interested in cortical function (Van Essen and Drury, 1997, 1998). A polygon-mesh representation of the structure of a subject's cortex allows, for instance, the visualization of functional imaging data on a two-dimensional flatmap (Sereno *et al.*, 1995). Taking advantage of the inherent structure of the cortex, surface-based representations are concise in that they assemble in a single image information otherwise spread over a series of slice images, whose detailed spatial relations are cumbersome to understand. Despite a certain degree of inevitable geometric distortion, surface-based representations give an idea of the intracortical—as opposed to the Euclidean—distance between locations. But beyond mere visualization, an accurate polygon-mesh representation of an individual subject's uniquely folded cortex provides an important constraint for statistical analysis of functional data from that subject. On the simplest level, the analysis can be restricted to the region within the functional data set that represents the cortex (Goebel and Singer, 1999; Formisano *et al.*, 2000). More complex applications of surface representations include intraarea mapping, e.g., the computation of retinotopic maps of early visual areas (Sereno *et al.*, 1995; Linden *et al.*, 1999),

1053-8119/01 $35.00

and the use of anatomically informed basis functions (Kiebel *et al.*, 2000) for computation of functional maps in general.

The computational methods used to obtain a polygon mesh that accurately represents the cortex can be grouped according to the type of representation they operate on. On one hand, there are segmentation methods operating on the anatomical MR volume in voxel space, on the other hand, there are iterative morphing methods operating on polygon-mesh representations.

Dale *et al.* (1999) take the approach of segmenting the white matter, thus recovering the inner boundary of the cortical gray matter in voxel space. The initial polygon-mesh representation of the cortex is computed by exact recreation of the boundary of the binary voxel object produced by segmentation. Slight smoothing of this initial polygon mesh yields a representation of the cortical sheet that is geometrically very accurate, but typically contains a number of topological errors, called handles.

Handles are toroidal structures of the polygon mesh. They are incompatible with what is known about cortical anatomy. The cortex is a folded sheet, i.e., the pial surface may touch itself across a sulcus but the gray matter is nowhere grown together through the pia mater. The reconstructed boundary of the voxel object should therefore ideally be a simple polyhedron, i.e., it should be without handles. Handles result from noise and partial volume sampling of the anatomical MR volume. Because the cortex has a simple topology, every handle represents an error in the reconstruction of the cortical sheet. But beyond local inaccuracy of surface representation, handles have particularly deleterious effects—causing large geometric distortions—when the polygon mesh is inflated (Fischl *et al.*, 1999) or flattened. In the approach of Dale *et al.* (1999), handles have to be removed by manual intervention, which typically requires about 30 min per cortical hemisphere.

MacDonald *et al.* (2000) take the contrasting approach of recovering the structure of the cortical sheet by iterative morphing of an initial standard polygon-mesh model. The model includes both the inner and the outer boundary of the gray matter. This top-down, model-based approach lends itself naturally to the incorporation of *a priori* constraints, including simple topology of the surfaces. MacDonald *et al.* (2000) overcome the problem of geometric inaccuracy traditionally associated with iterative morphing methods: their reconstructions accurately represent the inner and the outer boundary of the gray matter. Whereas the resulting representation of the cortex is both topologically and geometrically correct, the extensive morphing of the initial standard model is extremely expensive computationally.

Here we describe an algorithm that detects and removes handles as part of the segmentation in voxel space. In combination with established methods, this algorithm allows topologically and geometrically accurate cortex reconstructions to be obtained at much lower computational costs.

First, we presegment the white matter automatically by methods similar to those described in Dale *et al.* (1999). Then we correct topological errors in the voxel representation using our novel algorithm whose core element is a distance-to-surface prioritized, selftouching-sensitive region growing process. This allows us to obtain a topologically correct initial polygon-mesh representation of an individual subject's uniquely folded cortex by a fast standard one-pass reconstruction method. Finally, the initial polygon-mesh reconstruction is subjected to morphing to obtain precise polygon-mesh representations of the inner and outer boundaries of the gray matter. In contrast to the approach of MacDonald *et al.* (2000), only very few iterations are required and simpler morphing methods suffice since the initial reconstruction already quite accurately represents the cortical sheet.

Our topology correction algorithm can be implemented such that its time complexity is linear in the number of input voxels. With anatomical MR datasets of 1-mm isotropic resolution, the complete process from an anatomical MR volume in Talairach space to polygon meshes representing the inner and the outer boundary of both hemispheres (Fig. 1) takes less than 15 min on a current PC workstation (1000 MHz dual Pentium).

METHOD

Presegmentation

The topology correction algorithm described in this paper can be applied to any binary presegmentation of the white matter that contains topological defects. As most of the white matter presegmentation methods we use have previously been described (see citations below), we only give a brief description of our presegmentation procedure here.

White matter presegmentation is performed by applying a sequence of operations to a 3-D T1-weighted anatomical data set. In our approach, the anatomical MR volume is first projected into Talairach space (Talairach and Tournoux, 1988; for our method see Linden *et al.*, 1999). This is the only step of our procedure currently requiring user interaction. The automatic presegmentation process first applies standard masks in Talairach space to remove the cranium and the cerebellum. This step preserves the intensity information of the original anatomical volume in the segmented region including the cortex. The deletion of voxels representing the cranium and the cerebellum is not yet clean at this stage: bits and pieces representing these structures remain in the volume. The midbrain structures and the ventricles are solidly filled in a pseudocolor. Whereas the midbrain structures are filled using a standard Talairach mask, filling of the ventricles requires a more adaptive approach, if the representation of the cortical sheet in this region is to be preserved. The ventricles are therefore filled by a region growing process seeded at points automatically located by analysis of the intensity values in the ventricular region.

Potential spatial variation of the white matter signal intensity is corrected for by fitting Legendre polynomials to estimate a three-dimensional bias field (inhomogeneity correction, see Vaughan *et al.*, 2001). In the intensity histogram computed over the remaining voxels within the Talairach proportional grid, the two peaks corresponding to white and gray matter are identified. The intensity difference between the two histogram peaks is used to control the inclusion range parameter of a sigma filter (Lee, 1983), which performs nonlinear, edge-preserving smoothing, reducing the noise within white and gray matter without blurring the white-gray matter boundary. The intensity values at the two histogram peaks determine the intensity range, within which a region growing process is performed to segment the white matter. The region growing process, seeded automatically in the white matter of one hemisphere above the corpus callosum, cleanly segments the entire white matter discarding isolated pieces of cranium and cerebellum that have not previously been removed.

Segmenting along the boundary between white and gray matter rather than along that between gray matter and cerebrospinal fluid yields a more accurate initial voxel space representation of the structure of the cortical sheet (Dale *et al.*, 1999). The binary object obtained by region growing is subjected to slight binary-voxel-object smoothing (cf. Dale *et al.*, 1999), which implements the *a priori* constraint of finite curvature of the cortical sheet. Finally, the two hemispheres are disconnected automatically. Below the corpus callosum, the boundary of the presegmented voxel object follows the solidly filled region masking the midbrain structures and the ventricles (henceforth referred to as the "subcallosal mask"), resulting in a flat representation of the medial plane within and below the corpus callosum. Everywhere else the boundary of the presegmented voxel object follows the cortical sheet—including the deep sulcal structure—along the boundary between gray and white matter. All subsequent operations are invoked separately for the two presegmented hemispheres.

At this point, the topology correction algorithm is applied to the presegmented voxel representation of each hemisphere. The boundary of the voxel object representing the cortical sheet is reconstructed by a simple tessellation method, which exactly recreates the cubic voxel structure in the polygon-mesh representation. The resulting polygon mesh is morphed to represent either the inner or the outer boundary of the gray matter by shifting each vertex along its surface normal until its position coincides with the respective intensity contour in the anatomical MR scan. The final result is shown in Fig. 1.

Rationale of the Algorithm

The core of the topology correction algorithm we propose is a region growing process (Fig. 2A) that starts deep inside the binary voxel object obtained by the presegmentation as described above. As the region grows, it eventually comes to represent the object as a whole. During the growth process the region is never allowed to selftouch (i.e., to form rings, definition below), ensuring that its boundary at every point of the process is a simple polyhedron (Fig. 2B). The growing is seeded at a point within the object that is maximally distant from the object's surface. Among the fringe voxels poised to be included into the region, a voxel maximally distant from the object surface is chosen at each step. This ensures that the region grows into the thinner parts of the object, where all voxels are closer to the surface, last (Fig. 2C). Let us assume for the moment that the handle in question resulted from erroneous inclusion of voxels into the object during the presegmentation. The handle is part of a closed ring, which may be cut at any location to render the resulting object ringless (i.e., its outer boundary a simple polyhedron). Where should the cut be placed? The algorithm implies a heuristic for identifying the erroneously included voxels according to which they form the thinnest part of the ring (see Formal definition of the algorithm and Discussion). This is in line with the idea that the modifications of the object should be kept to a minimum. The algorithm, however, does not strictly guarantee that the number of voxels changed is minimal (see Discussion). Since the region grows into the thinnest parts last, the region growing will terminate before the region selftouches in the thinnest part of the ring, excluding a small number of voxels (Fig. 2C).

The prioritization of the process has an intuitive physical analogy if we think of the object as a two-dimensional shape, its contour corresponding to the surface of a 3-D object. Imagine a flat landscape with a pool, the shape of the 2-D object. The two horizontal dimensions represent the object space. The third, vertical dimension is the depth of the pool (axis pointing downward), which represents the prioritization criterion distance-to-surface. While the landscape is flat outside the contour of the object, inward from the contour it declines at 45°, forming the pool, whose depth at every point equals the distance to the contour in the 2-D object (the distance-to-surface).

The prioritized region growing process is analogous to the pouring of a fluid into the pool from above the

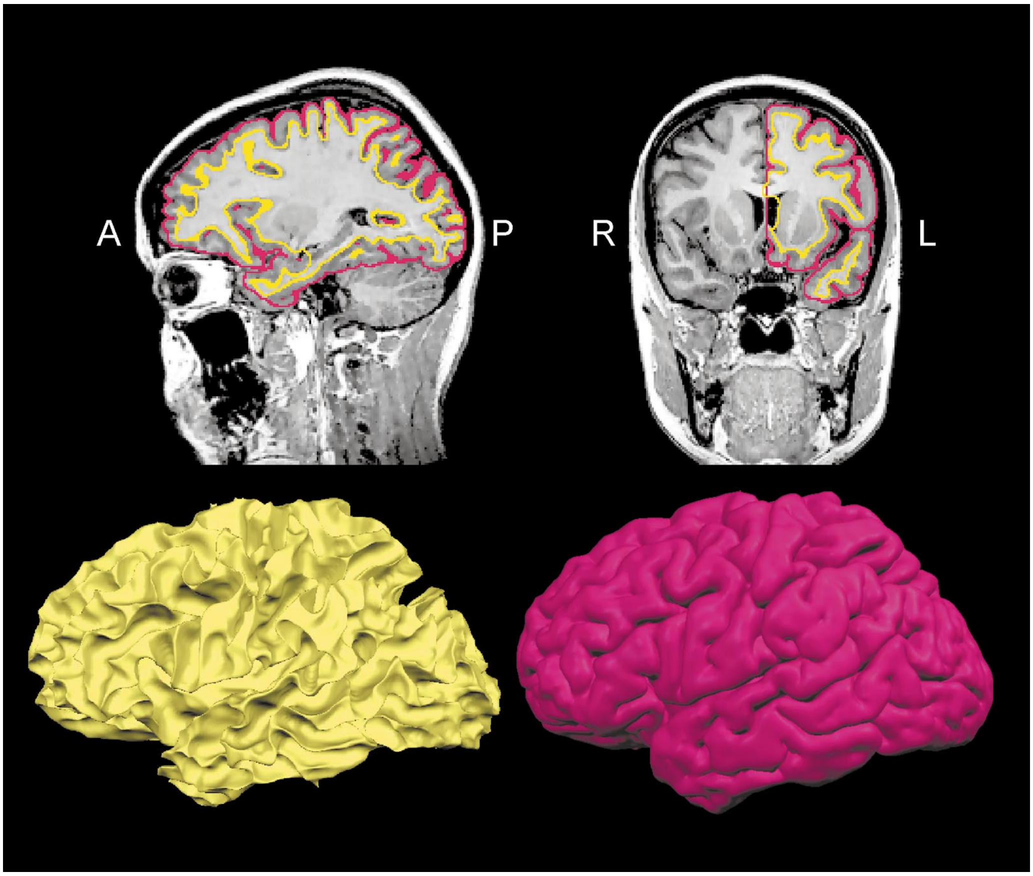

FIG. 1. Precise polygon-mesh representations of the inner and the outer boundary of the gray matter. Our algorithm was used to obtain a topologically correct initial polygon-mesh representation of the cortical sheet. Polygon meshes (bottom) representing the inner (yellow) and outer (magenta) boundaries of the gray matter have been computed by morphing the initial reconstruction. Projections of the surfaces back into the original anatomical MR volume (top) demonstrate the accuracy and precision of the representation of the two gray matter boundaries.

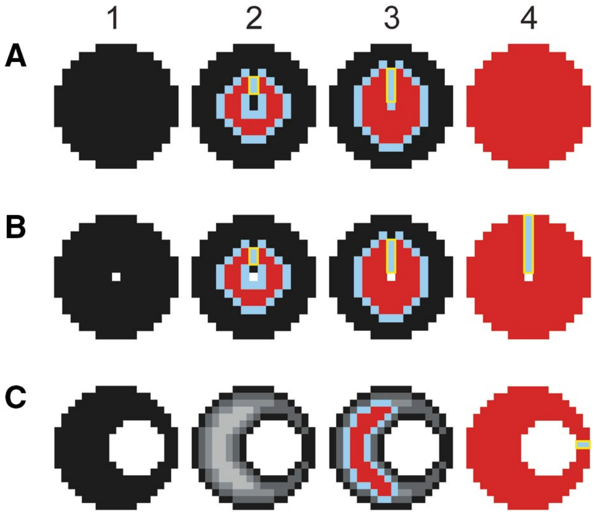

FIG. 2. Rationale: selftouching sensitivity and distance-to-surface prioritization. For three different 2-D objects (A–C), four snapshots of the process are shown in chronological order (1–4). To stress the analogy to the 3-D process, pixels are referred to as "voxels" and the outer contours of the objects as the "surface" in the following. Object voxels are black or gray. Region voxels are superimposed onto the object in red, fringe voxels in blue. Fringe voxels the region selftouches in, which are not eligible for inclusion into the region in the next step, are outlined in yellow. (A) Simple disk (1): During the process (2, 3) the region may, by chance, come to selftouch temporarily in some fringe voxels (yellow rectangles). Note that the region in 3, does not selftouch in the fringe voxel immediately below the yellow rectangle. When this voxel is included into the region, the region no longer selftouches in the fringe voxel above the included voxel. Eventually the region comes to include all voxels (4). (B) Ring: The object (1) differs from the previous one (A) only in that its central voxel is vacant, making it a ring. The region growing terminates (4) while there are still object voxels left that have not been included into the region. The excluded voxels (yellow rectangle) are all fringe voxels the region selftouches in. They represent an arbitrary cut through the ring. (C) Ring of nonconstant thickness: Here, the sequence of snapshots (1–4) demonstrates the distance-to-surface prioritization of the region growing. In this 2-D example, the contours of the object (1) represent the surface. Each voxel's distance-to-surface is indicated by its shade of gray (2): the brighter the voxel, the greater its distance-to-surface. The region growing starts in a voxel of maximal distance-to-surface (deep inside the object) and proceeds to voxels closer to the surface (3) until it reaches the thinnest part of the ring. There the region growing terminates (4), because there are no more fringe voxels the region does not selftouch in.

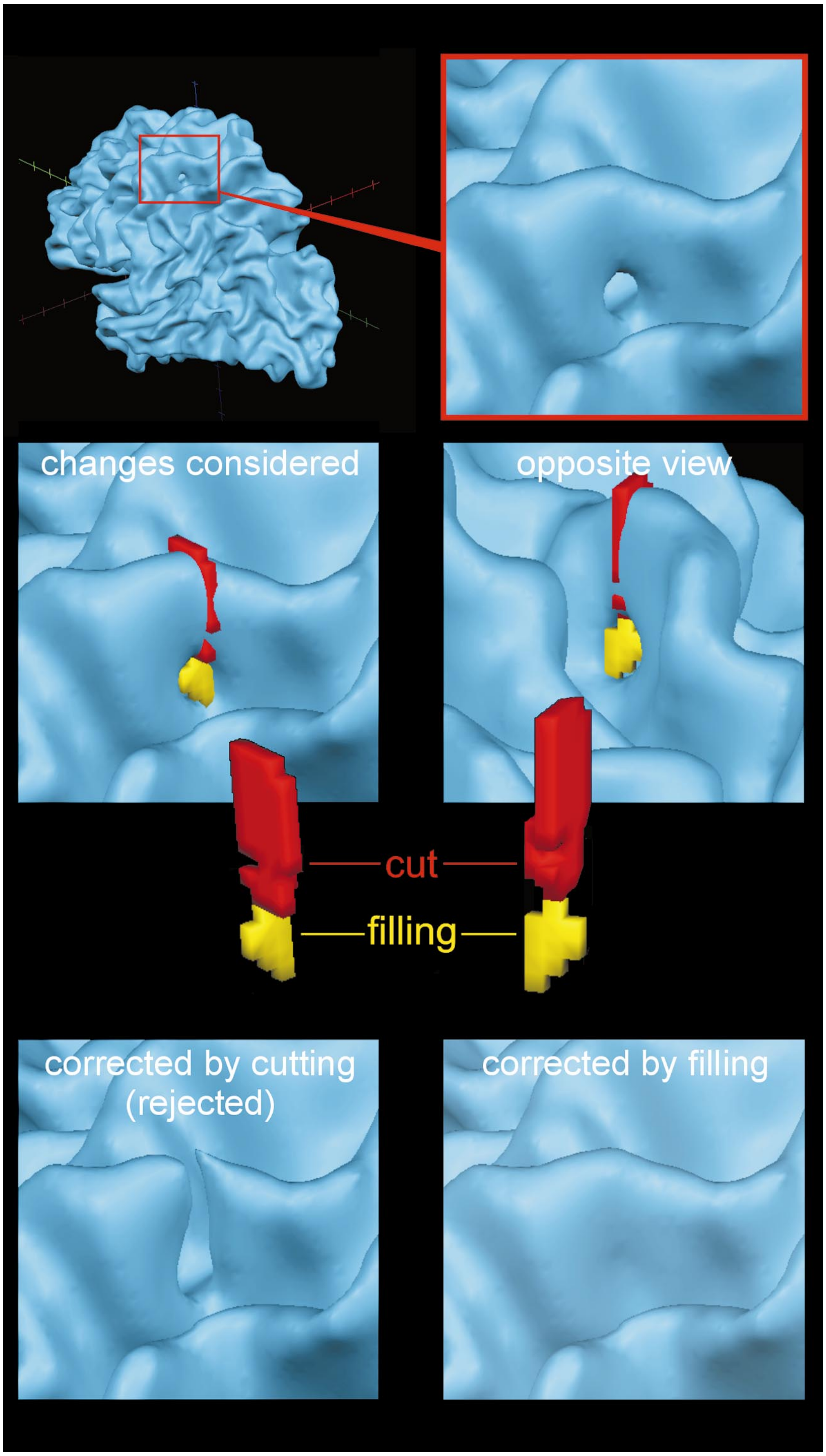

FIG. 3. Cutting the handle versus filling the hole. Topological errors can result from erroneous inclusion or exclusion of voxels in the binary segmentation. The topological correction can therefore consist in removing voxels (cutting the handle, red) or adding voxels (filling the hole, yellow). The figure shows both solutions as computed by distance-to-surface prioritized, selftouching-sensitive region growing. The algorithm chooses the better solution for each topological defect separately by a heuristic (Fig. 5). In this case, it has chosen to fill the hole. Though the algorithm operates in voxel space, surface renderings have been used to visualize the intermediate and final results. Whereas for the cut (red) and the filling (yellow) the surfaces reflect the cubic shape of the voxels, the polygon meshes representing the pre- and postsegmentation (blue) have been smoothed to ease visual orientation.

lowest point. The fluid will form a small pond, its surface rising in the pool until it reaches the level where the fluid can spill over into a second pond. This marks the beginning of a second phase, in which the fluid trickles down toward the lowest point of the second pond. The surface then rises in the second pond while it stays constant in the first. The fluid will wet one pond after another and at any point the surface will only rise in the pond where it is lowest. It will rise again in the first pond only after all other wet ponds have reached the surface level of the first pond. Eventually the fluid will fill the pool completely, reaching the level of the surrounding flat land.

The only aspect of the region growing process not covered by this analogy is the selftouching sensitivity: the region (the fluid) is never allowed to selftouch, that is its outer boundary may expand but not merge with itself to form rings.

The part of the algorithm described so far already constitutes a method of removing all handles: when the region growing terminates, the region is identical to the object except for a few missing voxels, which effectively cut all handles. Deletion of voxels seemed an appropriate correction method for the case of erroneously included voxels. Topological errors, however, can also result from erroneous exclusion of voxels. Such errors can be thought of as “holes,” though from the vantage point of topology each hole implies a handle and vice versa, rendering the two terms equivalent. If we assume the errors of the original segmentation to be random inversions of voxels, a method that can only delete voxels to render the object’s boundary a simple polyhedron seems incomplete.

Each topological error can be removed either by cutting the handle (deleting voxels) or by filling the hole (adding voxels). Assume we chose to remove an error by filling, which voxels should be added to the object? By the same logic applied above, it seems sensible to add a minimum number of voxels. Since each handle of the object corresponds to exactly one handle of the inverse object, we can apply the procedure described above to the inverse object. We will identify a set of voxels whose addition cuts the corresponding handle in the inverse object. If we add this set of voxels to the original object we have corrected the error by filling it. The proposed algorithm computes both solutions and chooses, for each handle separately, the solution that does less damage (defined below) to the segmentation (Fig. 3).

Formal Definition of the Algorithm

The term “object” in the following refers to the binary presegmentation of a cortical hemisphere along the boundary between gray and white matter as described above.

Distance-to-Surface Mapping

The distance-to-surface of a point within the object, ideally, is the length of the shortest line to the surface of the object. We estimate this value for every voxel by counting the number of times the object needs to be eroded until the voxel is removed (Fig. 4A). Let Obj be the set of voxels representing the object. For each object voxel v, the distance-to-surface function $d(v)$ is computed as follows:

$$d(v) = \begin{cases} -1 & \text{if } v \in \text{Obj} \\ 0 & \text{otherwise} \end{cases}$$

$$cd = 0$$

while($\exists$ voxel $u \in \text{Obj} | d(u) = -1$)

$$cd = cd + 1$$

$$d(v) = \begin{cases} cd & \text{if } d(v) = -1 \wedge (\exists \text{ voxel } w | \text{side-adjacent}(v, w) \wedge d(w) = cd - 1) \\ d(v) & \text{otherwise} \end{cases}$$

endwhile,

where cd is the distance-to-surface currently being mapped and side-adjacent(v, w) means that voxels v and w share a side. By analogy point-adjacent(v, w) means that they share at least one point and edge-adjacent(v, w) means that they share at least one edge (Fig. 4B).

Selftouching-Sensitive, Distance-to-Surface Prioritized Region Growing

Let Obj be the set of voxels representing the object, R the set representing the growing region (initially empty) and F the set of fringe voxels poised to be included into the growing region (Fig. 2). The fringe F is initially seeded with a single object voxel maximally distant from the surface. The region growing process representing the core of our algorithm can be expressed as follows:

$$R = \{\ \}$$

$$F = \{\text{voxel } u\},\ u \in \text{Obj},\ d(u) = \max$$

while($\exists$ voxel $v \in F | d(v)$

$= \max$ in $F \wedge \neg$ selftouching(R, v))

$$F = F \backslash \{v\}$$

$$R = R \cup \{v\}$$

$$F = F \cup \{\text{voxels } w | \text{side-adjacent}(v, w) \wedge w \in \text{Obj} \backslash R\}$$

endwhile.

Selftouching(R, v) means the region R growing inside the object Obj selftouches in a voxel $v \in \text{Obj}\backslash R$. The function selftouching(R, v) is true if and only if v connects two region voxels point-adjacent to v that were not already connected within the neighborhood of v:

$$\text{selftouching}(R, v) \Leftrightarrow$$
$$\exists(u, w)|u, w \in \text{neighborhood}(v) \cap R:$$
$$\neg\exists \text{ point-adjacency path set } P \subseteq \text{neighborhood}(v) \cap R|u, w \in P,$$

where neighborhood(v) is the set of all voxels point-adjacent to v excluding v itself and a point-adjacency path set is a set of voxels $P = \{x_1, x_2, \ldots x_n\}$ such that x_i and x_{i+1} are point-adjacent for all $i = 1 \ldots n - 1$. For verbal simplicity, a voxel the region selftouches in will be referred to as a selftouching voxel in the following.

Since selftouching fringe voxels are never included into the region, the region's outer boundary remains a simple polyhedron until the process terminates.[1] If the genus of the object's boundary was greater than 0, i.e., if there were handles, there will be a nonempty set $C = \text{Obj}\backslash R$ of cut voxels. C is the union of disjoint sets C_1, $C_2, \ldots C_k$, each of which represents a single cut, i.e., a contiguous set of cut voxels:

$$\forall i = 1 \ldots k:$$
$$\forall(v, w)|v, w \in C_i:$$
$$\exists \text{ point-adjacency path set } P \subseteq C_i|v, w \in P.$$

Region Growing in the Inverse Object

The selftouching-sensitive, distance-to-surface prioritized region growing process described above is also applied to the inverse object $\text{Obj}' = U\backslash\text{Obj}$, where U is the set of all voxels. To ensure that the region can grow around the object, the object must not touch the outer boundary of the block of all voxels; i.e., the outer boundary of U must not contain points belonging to the object. Invoking the procedure described above substituting Obj′ for Obj, yields a set of voxels not included when the region growing terminates. We will call it the inverse or negative cut set C', which is decomposable into disjoint contiguous sets $C'_1, C'_2, \ldots C'_l$, each corresponding to a negative cut as described above for the positive cuts $C_1, C_2, \ldots C_k$.

[1] The algorithm exploits the fact that whether a local change of the region changes its boundary's genus can be determined from local information. This fact also follows from the Euler–Poincaré formula $2 - 2g = v - e + f$, where g is the genus and v, e, and f are the number of vertices, edges, and faces, respectively. Since g is a linear function of v, e, and f, knowing the local change of v, e, and f suffices to predict the change of g.

Filling or Cutting: Choosing the Better Solution for Each Handle

Simple topology could be achieved by performing either all positive or all negative cuts. Our algorithm, however, is locally adaptive in its choice of a combination of cuts that renders the topology simple. If we were forced to sketch the choice of cut combination in three sentences, they would read as follows: For each handle, there is a positive and a corresponding negative cut; either of them removes the handle. Corresponding cuts can easily be identified because they touch. For each handle independently, the cut doing less damage to the presegmentation is chosen.

Unfortunately, matters are more complicated as, first, more than two cuts can form a set whose elements interact in their effects on the topology and, second, because the cuts are not ideal two-dimensional entities, but sets of voxels of nonzero volume, the number of positive cuts can be different from the number of negative cuts.

Concerning the first point, sets of interacting cuts can be identified by their property of spatial linkage. For ideal two-dimensional cuts of zero volume, linkage between two cuts means that the cuts share a point. The elements of a set of interacting cuts can be thought of as the nodes of a graph whose edges represent linkage. The resulting graphs are trees. All cuts together form a forest. Cuts belonging to different trees of the forest have independent effects on topology, so each tree can be treated independently of the others. There is a simple rule for determining which combinations of cuts of a tree can be chosen, such that if, for each tree of the forest separately, one of these combinations is chosen, the genus of the object will be zero. We do not use this rule, however, as, first, the fact that cuts are not ideal but have a nonzero volume further complicates matters and, second, in the domain of cortex segmentation from anatomical MR volumes, we have rarely encountered these more complex cases and, where we did encounter them, it was in the region of the subcallosal mask, where the boundary of the voxel object does not represent the cortical sheet. Instead, the present version of our algorithm chooses, for each set of interacting cuts separately, to make either all positive or all negative cuts. The loss of local adaptivity this entails is negligible, because almost all sets of interacting cuts contain exactly one positive and one negative cut.

First, we group all positive and negative cuts into sets of interacting cuts, which we call choices. A choice E, thus, is a set of cuts. For cuts as defined here (contiguous voxel sets of nonzero volume), the property of linkage, allowing us to identify the choices, is given if cuts are linked by a side-adjacency. Let S be the set of choices, initially containing, for each positive and each negative cut, a degenerate choice composed of

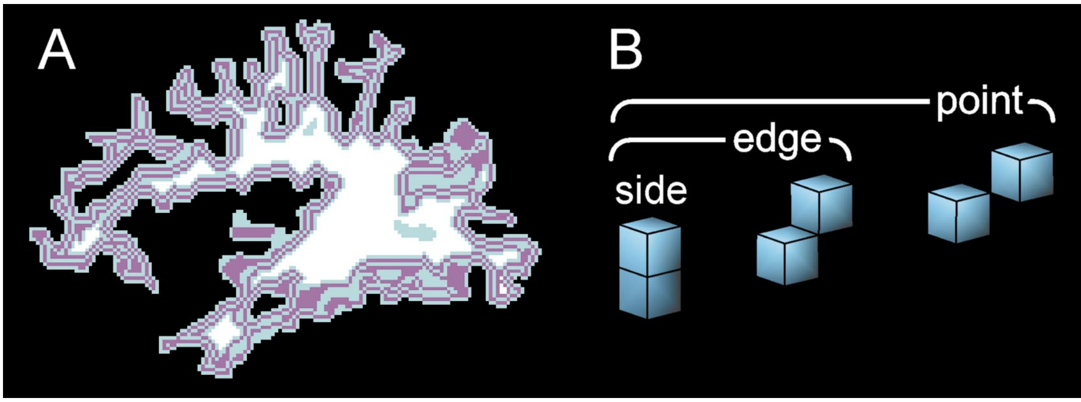

FIG. 4. Distance-to-surface prioritization and adjacency relations between two voxels. (A) Distance-to-surface prioritization: Odd priority levels are shown in blue and even ones in magenta. Larger regions of one priority level occur where the 3-D surface obliquely intersects the sagittal slice shown. The core of the object (white) does not need to be prioritized since its boundary already has a simple topology (see footnote 2, Implementation and time complexity). (B) Adjacency relations between two voxels: Two voxel are side-adjacent if they share a face, point-adjacent if they share at least one vertex and edge-adjacent if they share at least one edge.

only that cut. We can compute the choices by repeatedly merging sets of cuts linked by side-adjacency, as follows:

set of choices S

$$= \{\text{choice } E \mid E = \{D\},\ D \in \{C_1, C_2, \ldots C_k\} \cup \{C'_1, C'_2, \ldots C'_l\}\}$$

while ($\exists$ voxels $v, w \mid v \in$ cut $A \in$ choice $X \in S$, $w \in$ cut $B \in$ choice $Y \in S$, $X \neq Y$, side-adjacent(v, w))

$X = X \cup Y$

$S = S \backslash \{Y\}$

endwhile.

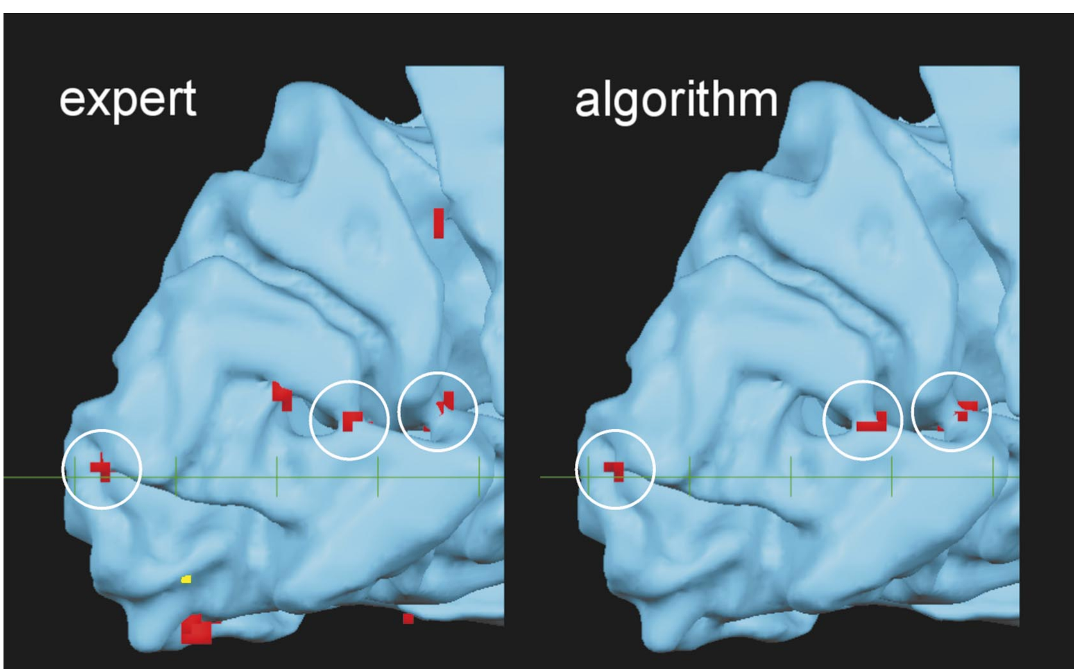

FIG. 6. Deletions by the expert and by the algorithm. The figure shows how the expert and the algorithm corrected three topological errors (highlighted by white circles) located on the medial side of the occipital lobe of a left hemisphere. The presegmentation is shown as a smoothed surface (blue). Voxels deleted by the expert (left) and the algorithm (right) have been inserted in red. Note that the topology-correcting changes chosen by the expert and by the algorithm are extremely similar: both correctly identified the three errors and chose deletion as the method of correction in all three cases. Furthermore, the deletions chosen by the expert and by the algorithm are located at the same position within each toroidal structure. The expert made a number of additional changes (outside the white circles), a number of deletions (red) and one addition (yellow). Though they improve geometrical accuracy, these changes do not affect topology (see text).

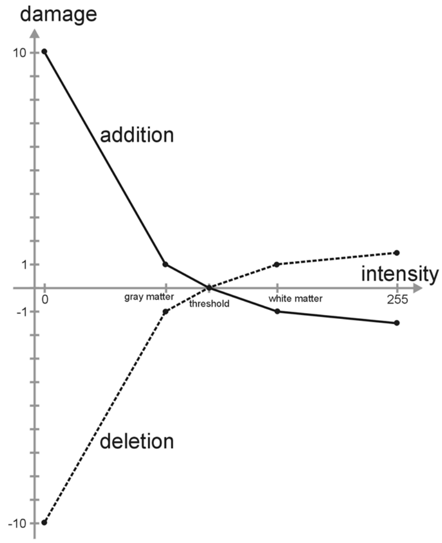

FIG. 5. Heuristic misclassification-damage function. The function provides a heuristic estimate of the misclassification damage caused by inverting a voxel in the presegmentation. In order to choose between corresponding solutions (cutting and filling, Fig. 3) for each locus of topological error, the damage each solution would do to the presegmentation is estimated by summing the misclassification damages of the inverted voxels.

As defined in the previous section, $\{C_1, C_2, \ldots C_k\}$ is the set of positive and $\{C'_1, C'_2, \ldots C'_l\}$ the set of negative cuts. When this procedure terminates, S is the set of all choices and each choice in S is a nonempty set of positive and negative cuts. Most choices will contain exactly one positive and exactly one negative cut, but the number of cuts in a choice set can occasionally be larger than two and the number of positive cuts need not be equal to the number of negative cuts.

To decide, for each choice separately, whether the positive or the negative cuts should be made, we estimate the damage done by each of the two solutions in terms of misclassification of voxels in the final segmentation and choose the solution doing less damage. The misclassification damage is estimated by the following heuristic: The damage done by a set of cuts is defined as the sum of the damages done by the cuts and the damage done by a cut is defined as the sum of the damages done by the voxels the cut comprises.

$$\text{deletion damage(choice } E) = \sum\nolimits_{\text{cuts } D\in E|D\in\{C1,C2,\ldots Ck\}} \text{damage(cut } D)$$

$$\text{addition damage(choice } E) = \sum\nolimits_{\text{cuts } D\in E|D\in\{C'1,C'2,\ldots C'l\}} \text{damage(cut } D)$$

$$\text{damage(cut } D) = \sum\nolimits_{\text{voxels } v\in D} \text{damage(voxel } v)$$

The misclassification damage of a voxel depends on whether the voxel is to be deleted (positive cut) or added (negative cut), on the intensity of the voxel in question, the intensity of prototypical gray and white matter voxels in the original unsegmented MR volume (as identified before the presegmentation by histogram analysis), and on the threshold intensity used in the presegmentation. Deletion (positive cutting) means that a voxel classified as white matter in the presegmentation is reclassified as an outside-white-matter voxel, and vice versa for addition (negative cutting). Ideally, voxels whose intensity is above threshold should be classified as white matter, and voxels whose intensity is below threshold should be classified as outside-white-matter. (This is not strictly the case after presegmentation, because in the presegmentation thresholding is followed by binary-voxel-object smoothing.) The rationale of the heuristic used to estimate the misclassification damage caused by inverting a voxel (Fig. 5) is as follows: If a voxel whose intensity is at threshold is inverted, no damage is done (damage value 0). If a voxel whose intensity is prototypical of white matter is misclassified as outside-white-matter, the damage is as great as that of misclassifying a voxel whose intensity is prototypical of gray matter as white matter. These outright misclassifications are arbitrarily assigned a damage value of 1. If a voxel whose intensity is prototypical of white matter is correctly classified as white matter after inversion, the inversion clearly corrected an outright misclassification of the presegmentation process and is therefore assigned a damage value of -1. By the same logic, the classification of a voxel whose intensity is prototypical of gray matter as outside-white-matter is assigned a damage value of -1. A voxel intensity close to 0 (black) indicates that the voxel represents a point in the brain that is not only outside of the white matter, but also outside of the gray matter and, thus, even further away from the gray-white matter boundary, which we aim to recover, than a prototypical gray matter voxel. The damage done by adding such a voxel should therefore be markedly greater than that of adding a prototypical gray matter voxel. In all empirical tests presented in this paper we assigned a damage value of 10 to the addition of a voxel of intensity 0. By symmetry, the reclassification of such a voxel as outside-white-matter (deletion), correcting a previous misclassification of the same severity, is assigned a damage value of -10. In contrast to the case of black voxels, white voxels whose intensity exceeds that prototypical of white matter cannot be inferred to be further away from the gray–white matter boundary than prototypical white matter voxels. Such voxels are merely to be classified as white matter with greater confidence. Therefore the damage values assigned to reclassification of such voxels are only slightly greater in absolute value than those for reclassification of prototypical white matter voxels. The damage values assigned for reclassification of a voxel of intensity 255 are 1.4 for the case of deletion and −1.4 for the case of addition. Between the points just motivated, the misclassification-damage functions for addition and deletion are linear as shown in Fig. 5. In the present version, the total damage value of each voxel that is to be inverted is obtained by adding 1 to its misclassification-damage value, ensuring that among corresponding solutions (cutting versus filling) the smaller set of voxels is chosen if both sets cause similar amounts of misclassification damage.

EMPIRICAL VALIDATION

Our algorithm is guaranteed to remove all handles, rendering the surface topology of the segmentation simple. Empirical validation is required to go beyond this analytical fact and answer the question whether the corrections reflect the true geometry of the subject's cortical sheet at the locations where the presegmentation contained topological errors. We validated the algorithm's behavior in three ways: first, by subjecting its topology correction to a handle-by-handle and voxel-by-voxel comparison to a human expert's correction performed independently; second, by inspecting the corrections of about a thousand topological errors in the context of the polygon-mesh surface representation as well as in the context of the voxel volume of the original anatomical MR scan, and finally by crossvalidation between segmentations based on different scans of the same subject's brain.

Comparison to Expert Performance

As our algorithm automatizes a task previously performed by an expert, the natural way to validate the algorithm is to compare its performance to the expert's. The manual topology correction, as it was routinely performed in our lab before the algorithm was developed, takes about 30 min per subject, matching the duration mentioned in Dale *et al.* (1999). Application of our algorithm as a test of topological integrity to presegmentations manually corrected previously, however, revealed that small handles had frequently been overlooked. For the purpose of this validation study, the expert (author R.G.) therefore took particular care in his manual correction of the presegmentations, frequently requiring more than a full hour per subject. The presegmentations corrected by the expert were tested for topological integrity before the comparison, allowing us to compare the corrections chosen by the expert and by the algorithm for every single topological error. The expert and the algorithm independently proceeded from the same presegmentation of each scan, allowing a precise voxel-by-voxel comparison of the changes. Like the algorithm, the expert inspected the original grayscale MR scan to decide how to correct the topology in the presegmentation. Unlike the algorithm, the manual correction procedure also included repeated polygon-mesh reconstruction and partial inflation based on the topologically incorrect presegmentations to locate topological errors and assess the local 3D geometry. This labor-intensive form of validation has been performed for 18 hemispheres with a total of 326 topological defects. The results are summarized in the table. The part of the medioventral region where the segmentation's boundary does not represent the cortical sheet but the subcallosal mask used in the presegmentation (see Method) has been excluded from the comparison. The number of handles found in this region is given in Table 1 (*s*).

The comparison of the changes was carried out at two levels of analysis. First, on the level of single voxels, we counted voxels inverted only by the algorithm, by both, algorithm and expert, and only by the expert. Second, on the more abstract level of topological errors, we counted the number of handles corrected "congruently" and the number of handles corrected "incongruently." The corrections chosen by algorithm and expert to correct a particular handle were counted as congruent if and only if: (1) the same method (deletion or addition of voxels) was chosen by both and (2) the cut (or inverse cut) was placed at the same location along the ring. Two nonidentical cuts (contiguous sets of inverted voxels) were considered to be "at the same location along the ring" only if they overlapped, i.e., if their intersection was a nonempty set of voxels.

The analysis on the level of topological errors shows that the algorithm chooses the same kind of correction (deletion or addition) and the same location for most handles (bold in the table). Figure 7 shows several examples of how the algorithm and the expert corrected the same topological errors in the scan of subject CG. If we exclude the extreme case of subject TI, in which the presegmentation failed catastrophically due to an inhomogeneity artifact in the inferotemporal region causing the expert to invert 5518 voxels, the algorithm corrected 84% of the handles congruently. Close scrutiny was given to the 16% incongruent corrections chosen by the algorithm (b in Table 1). Each handle corrected incongruently was reexamined, revealing, first, that most of them occurred in regions where noise and complex geometry rendered the situation ambiguous even to the expert and, second, that where the same method (deletion or addition) of correction was chosen by expert and algorithm, the two sets of inverted voxels frequently touched without overlap. Though they were counted as incongruent, the latter changes are qualitatively identical and quantitatively so similar that the difference in the initial reconstruction they entail is likely to fade when the surfaces are morphed to precisely represent the inner or the outer boundary of the gray matter.

The conclusion that changes made only by the algorithm are few is also supported by the analysis on the level of single voxel inversions. The number of voxels

TABLE 1

Comparison between the Algorithm and a Human Expert

		Automatic segmentation			Comparison between algorithm and expert: **Handle corrections**			Comparison between algorithm and expert: Voxel inversions			
Subject, hemisphere		Genus (number of handles) of the presegmentation (g)	Number of handles corrected by deletion, addition	Number of handles intersecting the subcallosal mask (s), excluded from the comparison	**Congruent** (a)	**Incongruent** (b)	Additional changes made by the expert that do not affect topology (c)	Performed only by the expert (d)	Performed by both, expert and algorithm (e)	Performed only by the algorithm (f)	Scanner (for models and sequences, see footnotes)
VA[a]	L	10	3, 7	3	**7**	**0**	4	38	39	0	GE 1.5T[b]
	R	7	0, 7	2	**3**	**2**	2	33	15	9	
CG[c]	L	11	4, 7	0	**9**	**2**	9	99	40	26	Siemens 1.5T[d]
	R	14	5, 9	5	**9**	**0**	22	168	32	25	
JH	L	7	2, 5	1	**3**	**3**	11	552	27	16	Siemens 1.5T[d]
	R	2	1, 1	0	**1**	**1**	7	47	2	4	
EF	L	14	2,12	1	**12**	**1**	64	1097	40	23	Siemens 1.5T[d]
	R	22	8,14	5	**13**	**4**	36	721	33	33	
TI[e]	L	63	31,32	7	**36**	**20**	58	4646	222	170	GE 3T[f]
	R	33	8,25	3	**23**	**7**	44	593	57	32	
RG	L	18	3,15	2	**12**	**4**	16	93	22	12	Siemens 1.5T[d]
	R	19	9,10	2	**14**	**3**	7	75	20	7	
SV	L	22	4,18	4	**15**	**3**	14	210	27	39	Siemens 1.5T[d]
	R	11	0,11	2	**8**	**1**	12	50	26	9	
MC	L	17	0,17	1	**13**	**3**	9	98	39	14	Siemens 1.5T[d]
	R	20	0,20	0	**18**	**2**	12	142	49	31	
FS	L	24	4,20	1	**21**	**2**	13	681	95	37	Siemens 1.5T[d]
	R	12	4, 8	1	**10**	**1**	0	293	55	14	
					Numbers of contiguous voxel sets that have been inverted[g]			Numbers of single voxel inversions[h]			

[a] Courtesy of R. Malach, Weizmann Institute.
[b] GE Signa Horizon LX 8.25, 1.5T, IR-prepared fast GRE T1-weighted sequence.
[c] Scan 1 in the crossvalidation (see Figs. 8 and 9).
[d] Siemens Magnetom Vision, 1.5T, T1FLASH.
[e] Courtesy of B. Wandell, Stanford University.
[f] GE Signa Horizon LXII, 3T, SPGR.
[g] Each handle has either been excluded (those intersecting the subcallosal mask) or classified as congruently or incongruently changed, thus $g = s + a + b$.
[h] Outside the region defined by the subcallosal mask, the number of voxels inverted by the expert is $d + e$ and the number of voxels inverted by the algorithm is $e + f$.

inverted by the algorithm that have not been inverted by the expert is generally small (f in Table 1). Again excluding subject TI, the average number of voxels changed by the algorithm but not by the expert is 1.5 per topological error. This attests to the similarity of the topology-correcting changes chosen by expert and algorithm. The voxel-level analysis also shows that the algorithm's corrections are extremely parsimonious. The algorithm inverted an average of only 4.3 voxels per topological error (subject TI excluded).

While parsimony, resulting from the thinnest-part heuristic (see Method and Discussion) implicit to our method, is an essential property of the algorithm's behavior, this does not hold true for the expert. The expert makes many changes (contiguous sets of inverted voxels) that do not affect the topology (Fig. 6: changes outside the white circles, c in Table 1) and where he corrects the topology he usually changes more voxels than the algorithm (Fig. 7). The reasons for this are as follows. The 2-D slice representations of the presegmentation, in which the expert actually inverts the voxels, visually reveal a ring structure only if it happens to be oriented parallel to the plane of the slices. Conversely, ring-shaped configurations frequently appear in a 2-D slice where there is no handle in the 3-D object. Thus, it is usually not obvious to the expert how a given topological defect can be removed changing a minimal number of voxels. Having located the defect in the polygon-mesh representation of the cortical sheet, the expert combines visual inspection of the original MR voxel intensities and knowledge about the spatial properties of the human cortex to improve the geometrical accuracy of the segmentation locally. Whether this removes the topological defect will become apparent only when topological integrity is checked in the next cycle of reconstruction and inflation. Since a geometrically accurate representation of the cortex is necessarily also topologically correct, im-

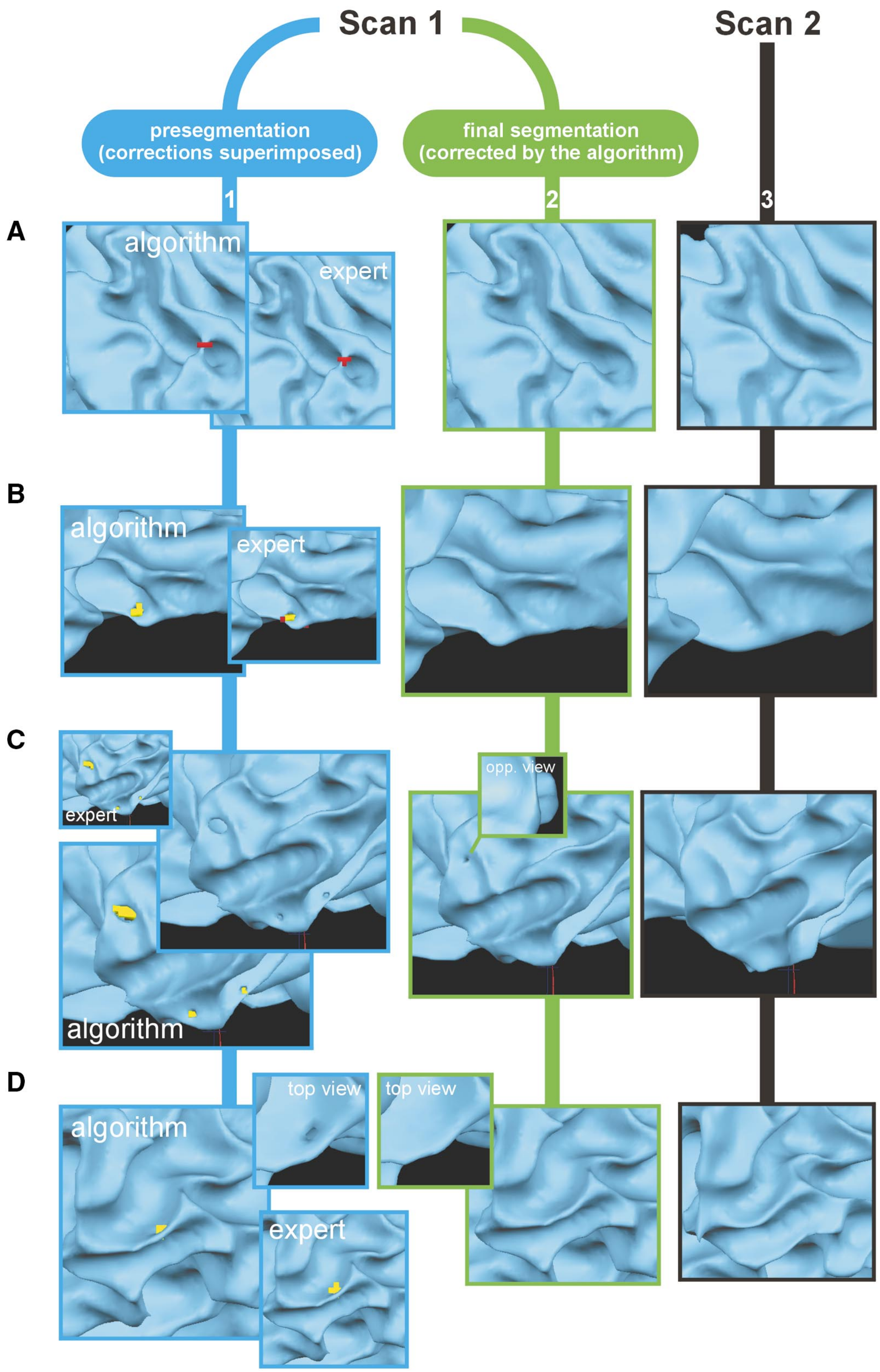

FIG. 7. Validation by comparison to a separate scan of the same subject (segmentation of scan 1 validated by scan 2). Smoothed reconstructions (blue) of the presegmentation (blue frames: 1), the final segmentation including the algorithm's corrections (green frames: 2) and a presegmentation of a different scan (scan 2) of the same subject (black frames: 3). Reconstruction has been performed completely independently for the two scans. Corrections by the algorithm and by the expert are inserted in column 1 (deleted voxels in red, added voxels in yellow). The figure shows medial views of the left parietal lobe (A), lateral views of the left occipital lobe (B), lateral views of the left

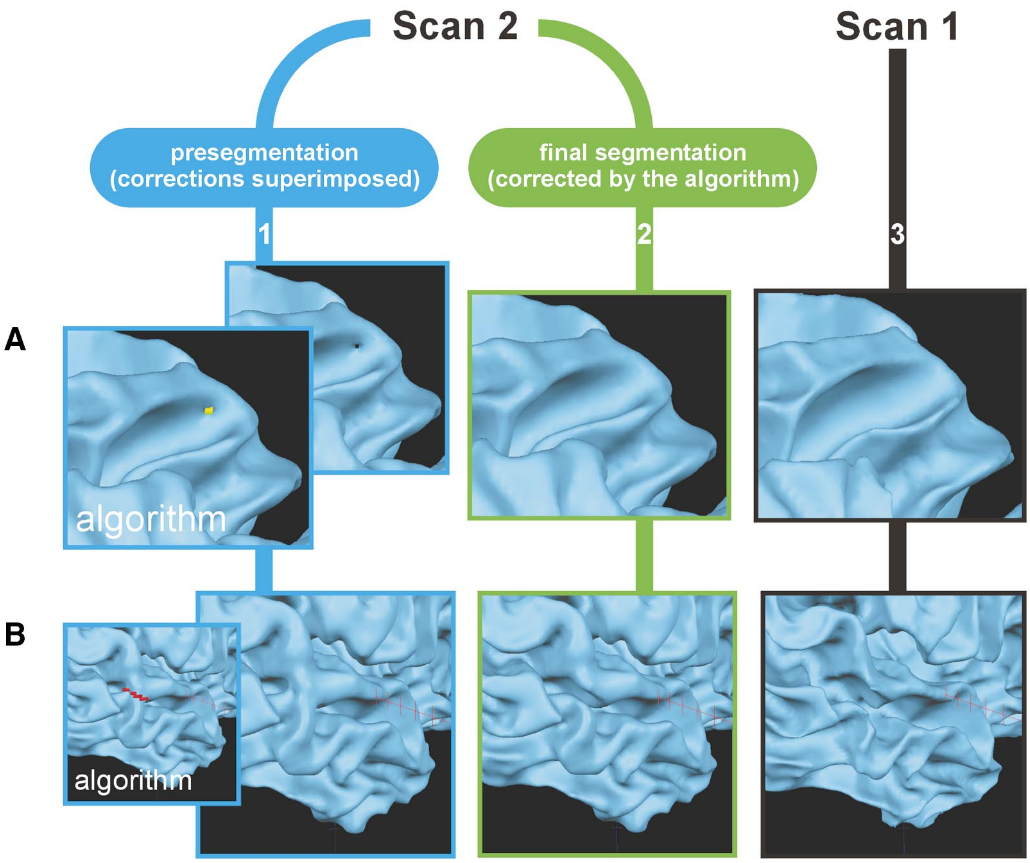

FIG. 8. Reverse validation (segmentation of scan 2 validated by scan 1). Here, the algorithm has been applied to scan 2 and the results are validated by examination of scan 1. The figure shows lateral views of the left parietal lobe (A) and of the right temporal lobe (B, topological error in the superior temporal sulcus). Axes, where visible, are Talairach axes.

proving local geometry is a good method of removing the topological defects.

Although the expert, thus, performed a more comprehensive task, essentially improving geometrical accuracy and removing topological errors as a side-effect, his behavior can serve as a reference in evaluating the algorithm, because we can show that the algorithm's changes constitute a subset of the expert's. The fact that changes made by the algorithm were, in general, also made by the expert represents evidence for correct behavior of the algorithm (a and e in Table 1). Since the algorithm is guaranteed to correct all topological errors, any additional changes made by the expert but not by the algorithm, though they may improve geometrical accuracy, are known to be unnecessary for topology correction and, thus, must not be considered in evaluating our algorithm's performance. Only the few changes made by the algorithm, that the expert did not make (b and f in Table 1), indicate divergence between the topology-correcting changes effected by the algorithm and the expert.

In summary, the comparison revealed that the expert and the algorithm correct topological defects in very similar ways, except for the fact that the algorithm is consistently more parsimonious in its modifications than the expert.

Visual Inspection of the Algorithm's Corrections

To be able to test our conclusions about the correctness of the algorithm's behavior on a larger sample of anatomical MR volumes, we next adopted a more efficient method of expert validation. We ran the complete segmentation and reconstruction procedure on a total of 30 anatomical MR volumes. The automatic correction of about a thousand topological errors was as-

temporal pole (C), and lateral and dorsal views of the right frontal lobe (D) of subject CG. Note the biggest of the three holes near the pole of the left temporal lobe (C1). There is a little dark spot in its place in the reconstruction of the corrected segmentation (C2, bottom). Examination of the opposite side of the hole (C2, top), however, shows that the topology is correct (see C3) and the dark spot is merely a small invagination of the surface. Axes, where visible, are Talairach axes.

sessed, first, by comparing the polygon-mesh reconstructions of the presegmentation and the final topology-corrected segmentation of each of the 60 hemispheres and, second, by inspecting an additional voxel volume written for validation purposes by our implementation of the algorithm, in which every voxel inverted in the presegmentation as well as every voxel inversion considered but rejected (the inverse solutions) is color-coded in the context of the grayscale intensity values of the original MR scan. We found that, where corrections were counterintuitive, it was the method of correction (deletion or addition), rarely the placement of the cut along the ring that seemed incorrect.

In the quantitative comparison to expert performance described in the previous section, there had been topological errors, whose adequate correction the expert felt unable to judge. This implies that a portion of the congruent corrections chosen by the algorithm might be inadequate and a portion of the incongruent corrections might reflect the true structure of the cortical sheet. During visual inspection we therefore restricted classification of the algorithm's corrections to topological errors whose adequate correction was apparent, determining the cleaner measure of the proportion of modifications chosen by the algorithm that could unequivocally be classified as inadequate. The proportion of corrections judged as inadequate was merely 3%.

Crossvalidation Using Duplicate Scans of the Same Brains

The approach to validation described so far relied on expert judgement on what constitutes the true structure of the cortical sheet in regions where the presegmentation is topologically incorrect. To go beyond expert judgement, we performed a second anatomical MR scan for three of our subjects. Segmentation and polygon-mesh reconstruction was performed independently from the two scans of these subjects. Since the topological errors of the presegmentation are mainly caused by noise, they come to lie in different locations, allowing a crossvalidation between the two scans.

For subject CG the results are documented in Figs. 7 and 8. Figure 7 shows the validation of the algorithm's segmentation of scan 1 by inspection of the presegmentation of scan 2. The reverse validation is shown in Fig. 8. Note that the surfaces in these figures are initial reconstructions of the segmentation, not yet morphed to precisely represent the geometry of the inner or the outer boundary of the gray matter. They serve to show how the algorithm corrects the topology in voxel space. To ease visual orientation, the surfaces representing the pre- and final segmentations (blue) have been smoothed slightly, whereas the surfaces representing the changed sets of voxels (red for deletions, yellow for additions) are inserted in their original cubic voxel shape. Figure 1 shows the final precise representations of the inner and outer boundaries of the gray matter obtained by morphing these initial reconstructions.

The crossvalidation performed for 6 hemispheres confirmed our finding that inadequate solutions are rare. The three outright errors found all concerned the method of correction (deletion or addition), confirming our intuitive judgement during visual inspection. Examination of the original anatomical MR volume revealed spatial inhomogeneity of the anatomical images as the cause of the failure of the heuristic choice between the two solutions. For each of the three incorrect solutions, enforcing the inverse solution rendered the resulting reconstruction correct. Furthermore, for all other handles, which had been removed by the correct method, the location along the ring of the object (or inverse object) where the cut was placed, reflected the true anatomy very precisely (see Figs. 7 and 8), lending further support to the thinnest-part heuristic implicit to the algorithm (see Method and Discussion).

LINEAR TIME COMPLEXITY IMPLEMENTATION

In this section, we show that the time complexity of the core components of the algorithm (distance-to-surface mapping and selftouching-sensitive, distance-to-surface prioritized region growing) is linear in the number of voxels of the input. Establishing the sets of interacting cuts, choosing which cuts to make and making the changes is also possible in time linear in the number of voxels, though this is a less important issue because the number of handles, in the domain of cortex segmentation in anatomical MR volumes, is usually small compared to the number of input voxels, rendering the time required for these components of the algorithm negligible compared to that required by the core components.

Region growing in 3-D voxel space has linear time complexity. It can be viewed as analogous to graph searching, the vertices representing the voxels and the edges the side-adjacency relation. Each vertex of a graph representing a voxel object in this way has at most 6 edges. Since the total number of edges in this special type of graph is, thus, proportional to the number of vertices, the searching can be performed in time linear in the number of vertices.

In our algorithm, the voxel object is, of course, represented by a 3-D occupancy grid. Fringe voxels as well as object voxels already included in the growing region are marked in the grid. Fringe voxels are additionally stored in a linked list. Time complexity is linear because every object voxel is included only once, and all operations associated with the inclusion of a voxel, i.e., marking it as included and updating the fringe set, can be performed in constant time using the occupancy grid (for the object, the region and the fringe) and the linked list representation of the fringe, which allows constant time access to the next fringe voxel to be included. Updating the fringe takes constant time because the number of object voxels to be considered has a constant upper bound of 6, the number of side-adjacent neighbors of the included voxel. To decide whether a voxel side-adjacent to the included voxel needs to be added to the fringe, all that is required is checking if it is an object voxel and if it is already present in the fringe set.

Region growing in 3-D voxel space can be used not only for the selftouching-sensitive, prioritized region growing process, but also for the distance-to-surface mapping: The growing in each step of the erosion (see above, Formal definition of the algorithm) is constrained to voxels representing the outer layer of object voxels not yet mapped. Each region growing pass maps one layer of the object and marks it as mapped. Since each pass takes time linear in the number of voxels contained in the layer it maps, and the numbers of voxels of all the layers add up to the number of voxels of the object, the distance-to-surface mapping takes time linear in the number of object voxels.[2]

To find the thinnest part of each ring structure, two features need to be added to a 3-D region growing process that merely floodfills the whole object: distance-to-surface prioritization and selftouching sensitivity.

Distance-to-surface prioritization can be added preserving linear time complexity by using a separate linked list or stack representation for fringe voxels of each priority level and maintaining a pointer to the highest non-empty fringe subset. This allows a highest priority fringe voxel to be accessed in constant time at each point in the process.

Adding the feature of selftouching sensitivity in a way that keeps the time complexity linear is a little trickier. Whereas the distance-to-surface of a voxel remains constant throughout the process, whether the region selftouches in a voxel or not can change several times as the region grows. In a naive implementation using only the representations introduced so far, this can lead to extensive searching through the fringe set for a fringe voxel the region does not selftouch in, rendering the time complexity quadratic in the worst case. To avoid searching for an includable fringe voxel, a separate fringe set for voxels known to be selftouching can be used at each priority level in conjunction with a voxel position grid of pointers. At each voxel position currently representing a fringe voxel known to be selftouching, the grid contains a pointer to the linked list element representing the fringe voxel. This is advantageous because the selftouching status of a voxel can only change when a point-adjacent voxel is included into the region. Fringe voxels known to be selftouching (which are stored in a separate linked list) are not considered for inclusion into the region. When a fringe voxel not known to be selftouching is considered for inclusion, its selftouching status must be checked, which takes constant time because the checking operates only within the 26-voxel neighborhood of the voxel in question. If the region does not selftouch in the voxel, the voxel is included into the region. If it does selftouch in the voxel, the voxel is moved to the separate list containing only fringe voxels known to be selftouching, and a pointer to its list element is inserted into the 3-D grid of pointers indicating that the voxel is known to be selftouching and where its list element is. A fringe voxel known to be selftouching is moved to the corresponding list of fringe voxels not known to be selftouching when a fringe voxel in its point-adjacency neighborhood is included into the region because this and nothing else can change its selftouching status. To be able to move fringe voxels between the two subsets of fringe voxels maintained for each priority level in constant time, we use doubly linked lists.

This scheme preserves linear time complexity because each object voxel eventually included into the region is handled in constant time. An object voxel is handled for the first time when it is included into the fringe. It is initially added to the list of fringe voxels not known to be selftouching. While it is in the fringe, it may be moved to the list of fringe voxels known to be selftouching and back to the list of fringe voxels not known to be selftouching a number of times before finally being included into the growing region. Each such move is handled in constant time and the number of times the voxel is moved back and forth between the two fringe subsets cannot exceed 26, because a voxel's selftouching status can change only when a point-adjacent voxel is included into the region, which takes place only once for each object voxel. Once all point-adjacent voxels are part of the growing region, the voxel can no longer be selftouching and is therefore included into the region the next time it is considered for inclusion. Actually the maximal number of times a fringe voxel can go through the cycle (become selftouching and nonselftouching again) is much lower than 26, but any constant upper bound suffices to show linearity. Most voxels will be included into the region either

[2] Performing the distance-to-surface mapping by region growing has an additional advantage: during each step of erosion it can be detected, whether the eroded object still contains rings. If it doesn't contain any more toroidal structures, the distance-to-surface mapping can terminate because how the region grows into the ringless core of the object in the main step of the algorithm is inconsequential for the result. If the eroded object still contains rings can be detected during the erosion by making the region growing marking the outer layer selftouching-sensitive. If the 2-D outer layer selftouches, the 3-D eroded object selftouches, i.e., contains rings. How selftouching sensitivity can be added to voxel space region growing keeping the time complexity linear is described below in this section.

when they are considered for inclusion for the first time or after very few moves.[3]

Selftouching-sensitive, distance-to-surface prioritized region growing, thus, can be performed in time linear in the number of object voxels, as can the distance-to-surface mapping. The same process applied to the inverse object is linear in the number of non-object voxels in the input, rendering the core of the whole process linear in the number of input voxels.

DISCUSSION

We have proposed an algorithm that makes changes in a binary voxel object to enforce simple topology of its boundary polyhedron. The intuitive rationale of the algorithm—minimizing the damage done to the presegmentation—makes our approach attractive for segmentation of any object whose topology is known to be simple. We have demonstrated and validated the algorithm's operation for the particular domain of recovering the spatial structure of the human cerebral cortex from an individual subject's anatomical MR scan. We have explained how the algorithm can be implemented such that its time complexity is linear in the number of input voxels.

While our algorithm is guaranteed to output an object whose boundary is a simple polyhedron, its choice of topology correction relies on two heuristics: the thinnest-part heuristic, which is implicit to the idea at the core of the algorithm, and the more easily replacable heuristic of choice between correction by addition and deletion. We will discuss each of them in turn.

The Thinnest-Part Heuristic

We have motivated the thinnest-part heuristic above by stating that it is more likely that a given ring structure resulted from erroneous addition or deletion of a small contiguous set of voxels than from addition or deletion of a larger one. Note, however, that the algorithm does not necessarily minimize the number of voxels changed to cut a handle. Instead, the location of the cut along the ring is the location where the ring breaks first during repeated erosion.[4]

Minimal cutting would be computationally much more expensive than our method of selftouching-sensitive, distance-to-surface prioritized region growing. For the case of anatomical MR scans of the human cerebral cortex, we would not expect the results to differ much if the algorithm strictly minimized the number of cut voxels. Either approach is heuristical and, thus, has to be validated empirically for the domain it is to be applied in, which we have done for our approach as applied in the domain of anatomical MR scans of the human cortex. The thinnest-part heuristic emerges as very powerful in choosing the location of the cut along the ring.

Heuristic Choice between Deletion and Addition of Voxels

An important property of our approach is the adaptive choice between two complementary solutions for each topological defect. Let's simplify a little and assume that there is a correct binary voxel representation of the object to be recovered whose genus is 0. After presegmentation the voxel object contains two types of error: voxels that are set but should be vacant (positive errors to be eliminated by positive cutting, i.e., by deletion) and voxels that are vacant but should be set (negative errors to be eliminated by negative cutting, i.e., by addition). The intensity threshold used in the presegmentation process influences the frequency of these complementary types of error. If the threshold is close to the intensity of the gray matter, there will be many positive errors and comparatively few negative ones. If the threshold is close to the intensity of the white matter, the reverse situation is to be expected. Positive cutting eliminates toroidal structures by removing voxels from the object (cutting handles). Negative cutting eliminates toroidal structures by adding voxels to the object (filling holes). The algorithm integrates both methods by choosing the better solution for each toroidal structure. It is, thus, adaptive in two important ways. First, it adapts to the local situation in the region of each toroidal structure. Second, it adapts to the threshold used globally in the presegmentation process. The threshold can thus be chosen independently of the topology correction step. This is an important property, because it allows other

[3] Note that a fringe voxel's selftouching status may change back and forth while the voxel is in the list of voxels not known to be selftouching without this entailing any move between the two sets, because selftouching status is only checked when the voxel is considered for inclusion into the region. An alternative implementation maintaining a fringe subset of voxels known not to be selftouching in addition to the subset known to be selftouching would, by the same logic, have linear time complexity as well. Since such an algorithm, as opposed to ours, tracks all changes of selftouching status for all fringe voxels, however, it would be considerably slower.

[4] The chosen cut $C(h)$ is a set of voxels whose maximal distance-to-surface is minimal within the set PC of all possible cuts removing the ring structure:

$$\max\{d(v)|v \in C(h)\} = \min\{\max\{d(v)|v \in C'\}|C' \in PC\},$$

where $d(v)$ is the distance-to-surface of voxel v. It is easy to imagine a ring structure for which the cut comprising the minimal number of voxels does not satisfy this criterion. If the cut of minimal size has the shape of a disk, for example, an alternative cut of a different shape can have more voxels that are all closer to the object surface than the maximal distance-to-surface within the disk-shaped minimal cut.

constraints to be taken into account in choosing the threshold.

The particular heuristic cost function by which the "better" solution is to be chosen is a separate issue. Here we chose the approach of minimizing the "damage" done to the presegmentation. The validation has shown that, though errors were rare, most of those that occurred concerned this heuristic choice, suggesting that a different cost function, e.g., based on local histogram analysis (to account for spatial inhomogeneity of the signal) or on geometrical *a priori* constraints might further improve the quality of our automatic topology correction.

Future Directions

Though the cortical surface reconstruction process as described in this paper is already very efficient and yields satisfactory results, we are following two lines of development. The first concerns the nature of the other segmentation operations and the stage within the segmentation process at which the topology correction algorithm is invoked. Before topology correction in voxel space was available, it was necessary to reduce the number of topological defects by applying binary voxel object smoothing (as described in the section Method) in voxel space. For the purposes of this paper, the presegmentation has been performed in the same way, with the topology correction inserted as the final step of the segmentation, immediately preceding polygon-mesh reconstruction. Though smoothing is appropriate as it implements the *a priori* constraint of finite curvature of the cortical sheet, it may be beneficial to reduce the amount of smoothing performed in voxel space. Furthermore, it may be preferable to perform the smoothing in a topology-preserving way after topology correction, as this would allow us to solve the problem of "needles." Needles are protuberances of the object surface that have a 1-D structure (a single voxel thick). Needles are not necessarily topological errors but clearly incompatible with what is known about cortex curvature. They occur due to noise in the original MR scan, and in the region of handles they sometimes resist smoothing if it is performed before the handle is removed, suggesting the iterative application to convergence of a topology-preserving binary voxel object smoothing operation after topology correction.

The second line of development concerns the self-touching-sensitive region-growing algorithm itself. We plan to develop the algorithm toward greater interactivity with the original anatomical MR scan: In the present version, the intensities of the original MR scan are used only in choosing, for each handle, between the previously determined complementary solutions of cutting and filling. Intensities could also be used to determine the locations of the cuts and fillings by including them in the prioritization function of the region growing process. The priority value of a voxel would reflect its classification confidence and be a function of both, the distance-to-surface (computed on the basis of a rough presegmentation) and the intensity of the voxel. Region growing would then be performed concurrently in the positive and in the negative object guided by this common priority metric, such that the two processes define the final segmentation interactively.

CONCLUSION

We have defined and validated an algorithm that segments anatomical MR scans of the human brain to recover the spatial structure of the cortical sheet. Based on a selftouching-sensitive, distance-to-surface prioritized region growing process, the algorithm enforces the *a priori* constraint of simple topology and yields geometrically and topologically correct reconstructions of the cortical sheet.

The algorithm is fast and, more importantly, has linear time complexity, making it suitable also for application to higher resolution or supersampled anatomical datasets. Combined with segmentation and polygon-mesh morphing methods similar to those described by Dale *et al.* (1999) and MacDonald *et al.* (2000), respectively, our algorithm allows us to obtain, fully automatically, topologically and geometrically accurate polygon-mesh representations of the inner and outer boundaries of the cortical gray matter at a computational cost much lower than that of previous approaches. The complete process from an anatomical MR volume of 1 mm isotropic resolution in Talairach space to polygon meshes representing the inner and the outer boundary of both hemispheres takes less than 15 min on a current PC workstation.

ACKNOWLEDGMENTS

The authors thank Miguel Castelo-Branco, Steffen Egner, and Elia Formisano (Universiteit Maastricht) for the insights and Roi Mukamel, Rafael Malach (Weizmann Institute of Science), and Brian Wandell (Stanford University) for the brains they contributed. This research was supported by the Max Planck Society and the Universiteit Maastricht.

REFERENCES

Carman, G. J., Drury, H. A., and Van Essen, D. C. 1995. Computational methods for reconstructing and unfolding the cerebral cortex. *Cerebral Cortex* **5:** 506–517.

Dale, A. M., Fischl, B., and Sereno, M. I. 1999. Cortical surface-based analysis. I. Segmentation and surface reconstruction. *NeuroImage* **9:** 179–194.

Fischl, B., Sereno, M. I., and Dale, A. M. 1999. Cortical surface-based analysis. II: Inflation, flattening, and a surface-based coordinate system. *NeuroImage* **9:** 195–207.

Formisano, E., Di Salle, F., and Goebel R. 2000. Subcomponents of visuospatial mental imagery revealed by event-related fMRI and cortex-based independent component analysis. *Soc. Neurosci. 2000.*

Goebel, R., and Singer, W. 1999. Cortical surface-based statistical analysis of functional magnetic resonance imaging data. *NeuroImage* **9:** S64.

Goebel, R. 2000. A fast automated method for flattening cortical surfaces. *NeuroImage* **11:** S680.

Kiebel, S. J., Goebel, R., and Friston, K. J. 2000. Anatomically informed basis functions. *NeuroImage* **11:** 656–667.

Lee, J. S. 1983. Digital image smoothing and the sigma filter. *Comput. Vis. Graph. Image Process.* **24,** 255–269.

Linden, D. E., Kallenbach, U., Heinecke, A., Singer, W., and Goebel, R. 1999. The myth of upright vision. A psychophysical and functional imaging study of adaptation to inverting spectacles. *Perception* **28:** 469–481.

MacDonald, D., Kabani, N., Avis, D., and Evans, A. C. 2000. Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI. *NeuroImage* **12:** 340–356.

Sereno, M. I., Dale, A. M., Reppas, J. B., Kwong, K. K., Belliveau, J. W., Brady, T. J., Rosen, B. R., and Tootell, R. B. 1995. Borders of multiple visual areas in humans revealed by functional magnetic resonance imaging [see Comments]. *Science* **268:** 889–893.

Talairach, J., and Tournoux, P. 1988. *Co-Planar Stereotaxic Atlas of the Human Brain.* Thieme Medical Publishers, NY.

Van Essen, D. C., and Drury, H. A. 1997. Structural and functional analyses of human cerebral cortex using a surface-based atlas. *J. Neurosci.* **17:** 7079–7102.

Van Essen, D. C., Drury, H. A., Joshi, S., and Miller, M. I. 1998. Functional and structural mapping of human cerebral cortex: Solutions are in the surfaces. *Proc. Natl. Acad. Sci. USA* **95:** 788–795.

Vaughan, J. T., Garwood, M., Collins, C. M., Liu, W., DelaBarre, L., Kim, S. G., Adriany, G., Andersen, P., Merkle, H., Goebel, R., Smith, M. B., and Ugurbil, K. 2001. 7T versus 4T: RF power, homogeneity, and signal-to-noise comparison in head images. *Magn. Reson. Med.*, in press.