Topological 3D reconstruction of multiple anatomical structures from volumetric medical data

Sylvain Gerbaud, Arthur Cavalier, Sébastien Horna, Rita Zrour, Mathieu Naudin, Carole Guillevin, Philippe Meseure

Topological 3D reconstruction of multiple anatomical structures from volumetric medical data

Sylvain Gerbaud, Arthur Cavalier, Sébastien Horna, Rita Zrour, Mathieu Naudin, Carole Guillevin, Philippe Meseure

Topological 3D Reconstruction of Multiple Anatomical Structures from Volumetric Medical Data

Sylvain Gerbaud a b, Arthur Cavalier b, Sébastien Horna a b, Rita Zrour a b, Mathieu Naudin a c, Carole Guillevin a c, Philippe Meseure b

a Labcom I3M, University Hospital Centre, CNRS, Siemens Healthineers, University of Poitiers, 86000 Poitiers, France

b University of Poitiers, CNRS UMR 7252, XLIM, 86000 Poitiers, France

c University of Poitiers, CNRS UMR 7348, Laboratory of Applied Mathematics LMA, 86000 Poitiers, France

Article info

Article history:
Received June 21, 2024

Keywords: Volumetric reconstruction, Generalized Maps, Topology-based modeling, Medical Images

Abstract

In the medical field, usually, practitioners mainly base their analysis on 2D slices produced from MRI or CT-scans that correspond to restricted views of a pathology. To facilitate the work of doctors, increase diagnostic accuracy and cross-reference multi-modal data, a 3D reconstruction is required. However, most of the time, reconstruction methods fail at visualizing complex and noisy data made up of several tissues. Indeed, these methods often build each tissue independently so that the consistency of the global model is not ensured: overlaps may appear between segments whereas some disjointed volumes exhibit empty spaces. This paper presents a complete topologically consistent reconstruction system from 3D medical acquisitions such as MRI or CT-scans. Compared to other methods, our system offers a single volumetric representation of an organ corresponding to a 3D space partition, where a semantic label is associated to each volume to identify the represented tissue and adjacency between volumes is explicitly and precisely defined. This partition is controlled and free from topological and geometric defects usually found in other 3D reconstruction approaches. Experimental studies were conducted on MRI datasets of brains resulting in consistent reconstructions. An application of the model for calculating the distribution of physiological data in brain tissue is also shown.

1. Introduction

Imaging technologies such as Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) are now commonly used in medicine. Usually, practitioners mainly base their analysis on 2D slices produced from those acquisition systems, that only correspond to restricted views of a pathology. Recently, visualization and operations on 3D reconstruction from medical acquisitions have been expanding since they supply practitioners with a more global representation of organs, improving the identification of their components and allowing them to perform additional analysis of anatomical functional activity.

For that purpose, a voxel-based model [1] is directly generated from a segmentation process that determines to which physiological element each voxel belongs to (each anatomical tissue correspond to one segment). However the use of such a discrete representation presents many disadvantages: (1) it is difficult to deal with holes and cavities, (2) the connectivity should be defined for both the object and its complementary (as required in discrete topology [2]) and is not easy to handle, (3) the borders are not smoothed, (4) the computation of geometric properties are limited to the size of voxels [3].

To overcome these limitations, the use of 3D meshes was widely studied in the literature as this kind of representation disregards voxels resolution. Many frameworks provide reconstructed (surface) meshes from medical data for precise visualization and analyses [4, 5, 6]. However they are not adequate for correctly visualizing noisy data composed of multiple tissues corresponding to anatomical elements. They usually require a very long runtime and use extensive processes that often need a user intervention to correct defects (overlaps of tissues, non-labeled volumes) that prevent a final mesh to be consistent within its domain. For example, the widely-used *FreeSurfer* software is the primary framework for neuroscien-

tists to build continuous models of brains [7]. It provides many tools for computing 2D and 3D information to analyze MR images [3], such as an accurate cortical thickness computation [8], a skull-stripping algorithm [9] and an automated segmentation of neuro-anatomical structures [10]. Nonetheless, it suffers from the drawbacks mentioned above, especially in case of noisy input MRI data. It does not compute any topological neighborhood between reconstructed components. In the best case, their surfaces are juxtaposed and may exhibit some approximate "contact zones", but in the worst case, their volumes can overlap, thus not preserving the topology of the segmentation.

To obtain a correct reconstruction of anatomical structures with several tissues, it is necessary to know and explicitly represent the topological relationships between the distinct regions (adjacency, inclusion, etc.) according to physiological constraints. More precisely, it is essential that the global volumetric mesh corresponds to a 3D space partition in which each component has its own disjoint space, neighborhood adjacency zones and characteristics (i.e. a space partition with no overlapping regions nor discontinuities between volumes to closely match the anatomical reality). For example, in the context of a 3D brain reconstruction, it is crucial to accurately separate white matter from gray matter and cerebrospinal fluid.

In this collaborative research work with a clinical partner, we propose a new reconstruction system to extract a volumetric model of segmented anatomical objects composed of several tissues. Our contribution resides in the 3D mesh consistency with topological and semantic constraints, in the explicit representation of adjacency relations, and in the cleaning phase able to leverage voxel information efficiently compared to other methods. More specifically, our main contributions are summarized as follows:

- The construction of a 3D space partition corresponding to a segmentation of the multiple input structures. This space partition contains no empty spaces (that is, volumes with unidentified content), is free of overlaps and has its own neighborhood adjacency zones and all properties are defined by a formal topological model;
- All adjacency relationships between physiological volumes are given explicitly. Thus, all inconsistencies like neighborhood incoherence in the anatomical structure or artifacts generated by the acquisition are detected and corrected automatically;
- The process relies on a minimum of geometrical computation and uses local topological operations which reduces the computation time. It offers a reconstruction respecting the anatomical structure and the topological constraints.

Finally, to illustrate the strength of our model, we highlight a potential application, involving complex geometric manipulations, for improving the visualization of data acquired by Magnetic Resonance Spectroscopy with respect to anatomy.

2. Related Work

To obtain a precise and topologically correct mesh, we have chosen to combine a parallel Marching Cubes algorithm with a formal topology-based volumetric representation. Related work on these two key elements are presented in this section.

2.1. 3D Geometric reconstruction method

Reconstruction methods aim at building a surface in 3D corresponding to the boundary of a single Region Of Interest (**ROI**). Methods can be grouped into several categories.

Deformable models. This kind of reconstruction methods have been extensively used in medical images. There are basically two types of deformable models: parametric deformable models and geometric ones. Parametric models [11, 12] generate an initial mesh in the continuous domain and apply geometric transformations to make its surface match the **ROI**. Thus, it has a subvoxel precision but expensive computational is needed to avoid self-intersections. These approaches require a target topology at the beginning or the end of the deformation. Thus, they can only be used in the case of an known topology, which makes the assumption that the reconstructed organ is healthy. Geometric deformable models give solutions to address their limitations. They are based on curve evolution theory [13] and level set methods [14, 15, 16], representing curves and surfaces implicitly. These models can easily avoid self-intersection [17, 18] and can suffer topological changes along evolution. However, with level set methods, we are not guaranteed to obtain a mesh with a desired topology. To address this limitation, topology correction [19] and topology preservation [20, 21] tools have been proposed, but this limits the use of geometric deformable models to topology-known applications.

Deep learning approach. Two classes, implicit and explicit methods, categorize the widespread application of deep learning approaches. The former use a deep neural network (DNN) to learn an implicit surface representation [22, 23, 24], then triangulate the final mesh using an isosurface extraction method (such as the Marching Cubes method explained below). Explicit methods operate a DNN to deform an initial mesh to a target [25, 5], to produce an explicit mesh directly. The same limitation as geometric deformable models appears: Deep learning approaches need to train on a initially-known topology. For instance, a neural network of cortical reconstruction methods is trained to reproduce meshes that are homeomorphic to a sphere, which is a wrong assumption if a tumor is present or in case of post surgery.

Isosurface reconstruction. One of the widely-used algorithms for 3D reconstruction is Marching Cubes (**MC**) [26]. From an input (acquired) voxel grid, the MC grid is constructed as its dual representation. This means that each cube in the MC grid is formed by considering the midpoints of eight neighboring voxels from the input grid. Edges in the MC grid connect the centers of adjacent voxels from the input grid. The simplicity, robustness, and speed of MC explain its popularity. However, guaranteeing topological correctness is a challenging issue when the acquisition resolution is not sufficient to clearly

identify local topology and causes ambiguities. Many methods with asymptotic decider have been proposed to deal with this problem [27, 28, 29]. Besides, new approaches emerged, such as Dual Marching Cubes [30] that replace triangles by quads to eliminate poorly-shaped triangles existing in **MC** surfaces. Some improvement has been made [31] to circumvent the topological ambiguities. Extra research has been done with neural networks to refine the mesh produced by Marching cubes [32, 33]. Moreover, each produced mesh is smoothed independently, leading to a loss of connectivity when multiple objects are reconstructed.

Reconstruction with control. Some reconstruction methods take into consideration connectivity and global topology to reconstruct data with multiple labels. They are not limited to specific surface reconstruction contrary to deep learning approaches [22, 24]. This kind of reconstruction methods allows for a topology control of a reconstructed surface by defining the exact property to match during the building process. Lazar et al. propose a surface reconstruction [34] that follows, in addition to a fixed genus, connectivity constraints on each sub-surface separately. Without using such constraints, Bruel et al. propose, based on persistent homology, to refine the implicit function from which the final surface is extracted afterwards [35]. Their approach is guaranteed to work with multi-label models only if they share the same topological properties. Based on dual contouring, Frisken et al. keep sharps boundaries to produce smooth meshes without discontinuities, by preserving the topology of the segmentation [36]. However, their method does not apply any kind of correction, and does not take into account the anatomical knowledge.

Current limitations. The main issue faced by the presented methods is that an accurate reconstruction of a non-arbitrary number of segments is not guaranteed. From medical data, a 3D space partition is needed, but overlaps and discontinuities may appear in some cases with these methods, especially between volumes of different segments. The various kind of reconstruction methods reviewed before can be used in a system, but will suffer from the same limitations. The methods that only consider a single surface, independently, are not suitable to connect multiple surfaces (overlaps, holes). That is why systems that incorporate multiple tissues in their reconstruction process are important. Under these conditions, in the medical field, we find exclusively systems such as the well-known FreeSurfer[7], along with any deep learning approach (DeepCSR[4], FastSurfer[22, 12, 24, 25]), which produce meshes homeomorphic to a sphere, without any adjacency relationships. These systems are useful for cortical reconstruction, but are not suitable for the reconstruction of a whole brain, especially in the presence of tumor tissue, because of the cavity created by the tumor (the mesh is no longer homeomorphic to a sphere). To our knowledge, no other tools are available, thus more generic tools that reconstruct multi-segment meshes and associate a topological structure are considered. Reconstruction methods with control can reconstruct any number of segments, but either the lack of control after the reconstruction step generates artifacts (superfluous connected components) and cavities [36] or some volumes are deleted without considering the anatomy [34]. The library CGAL [37] provides a system to reconstruct multi-label mesh with its corresponding topological model [38]. However, it overlooks medical data, leading to likely incorrect adjacency relationships between anatomical structures. Finally, isosurface reconstruction methods also offer the possibility to reconstruct any number of segments, by interpreting each segment as an implicit surface defined as $F : \mathbb{R}^3 \rightarrow \mathbb{R}$, such that the set of point $\{X \in \mathbb{R}^3 : F(X) = constant\}$ represents the surface (in most case *constant* corresponds to an isovalue). Many approaches have been proposed to do so [39, 40, 41], but it is necessary to ensure that the method builds manifold and watertight surfaces, without geometric interpolation. Also, it needs to avoid creating overlaps between distinct reconstructed surfaces. To control the topology of the model, a reconstruction relying on a topology-based model offers all tools to help building a correct partition of 3D space, by dealing with discontinuities, connected components and cavities due to artifacts or insufficient acquisition sampling.

2.2. Topological models

The concept of topology encompasses major properties such as the number of connected components, the adjacency of objects, the inclusion of an object in another, and so on.

Discrete approach. To study topology directly in voxel space, a discrete approach can be used [42]. However, it is topologically ambiguous so that it can lead to different interpretations. For instance, when two pieces of any segment are only connected by the corner of two voxels (also known as 26-connectivity), it is not immediately possible to state if these pieces are connected or just close or even in contact. Furthermore, despite the efficiency of segmentation methods, the content of some voxels, at the border of an area of interest, may remain ambiguous. For instance, in medical data mining, any voxel is considered to contain only one type of tissue after a segmentation, despite it may actually cover more than one type of tissue. This property comes from the resolution of the MRI images acquired that cannot capture precise details. Typically, the cerebral cortex is a very fine structure with very fine folding patterns only partially represented by a voxel-based discretization. As a consequence, a segmentation method suffers from imprecision, unclear adjacency and small isolated volumes generated mainly from the acquisition noise.

Continuous approach. Using a continuous method seems more appropriate to control topology since it represents the shape of reconstructed objects more precisely. Unfortunately, even if each segment is reconstructed correctly, merging all of them does not necessarily provide a correct global topology. In practice, overlaps or discontinuities (empty spaces) between volumes may appear, due to voxel ambiguities. To reconstruct a consistent multi-segment object, a powerful, topology-based, volumetric model must be used. Topology-based models represent meshes as a subdivision of cells of different dimensions (0 for vertices, 1 for edges, 2 for faces, 3 for volumes) that are

linked with explicit adjacency relations [43]. Some data structures such as winged-edges and half-edges [44, 45] are well-known, but they are restricted to surface meshes, that is, objects are only represented as a set of faces. Among these objects, volumes correspond to closed sets of faces. Using 2D topological models, it is easy to look for these sets and find volumes by following adjacent faces, but it is not possible to link these volumes to represent a 3D partition (the same way as faces are linked to create a surface). We therefore need a model that (1) explicitly represents volumes, (2) does not constraint the number of faces in volumes and the number of edges in faces and (3) links volumes to create complex 3D objects or partitions. Combinatorial maps suit these objectives well. This family of *nD* models includes oriented combinatorial maps (an extension of half-edges) and generalized maps (**G-maps**) [46] (Appendix A for more details).

From volumetric medical data, a 3D mesh is generated, and an explicit representation of its volumes is needed. Connectivity and thus discrete approaches are not enough to represent them. A continuous representation is more appropriate in this regard and can also topologically manage multi-segment object with explicit adjacency links. Even if the reconstructed mesh is dense (i.e., composed of a large number of faces and volumes), *nD* models can process it efficiently by doing all the operations locally, thus reducing the computation time. In this context, we chose to use generalized maps based on these criteria.

3. Overview

Discrete data given by medical acquisition systems consist in a bounded part of 3D space represented as stacked slices. Each slice appears as an image with a given thickness. It is composed of a number of voxels which size is in accordance with the spatial resolution of the acquisition (the smaller the voxels, the higher the resolution). In many systems (MRI, CT, etc.), each voxel is characterized with only a single value. A segmentation algorithm uses a duplication of an input voxel grid but separates the copy into multiple tissues, that we will refer to as Regions of Interest (**ROI**) or more simply, segments in our work. They usually correspond to different anatomical tissues. After a segmentation process, voxels are annotated with the label of the **ROI** to which they belong. Thus, the representation of segmented brain tissues is a grid of voxels where each voxel has a single value that determines its ownership to a tissue. However, since, in practice, a voxel may cover several tissues, the segmentation processes can also give probabilities of ownership for each tissue. This is called the Partial Volume Effect (later referred as **PVE**)[47]. In our case, we chose the highest value in the **PVE** to determine the final ownership of a voxel.

To obtain an accurate and topologically-correct mesh, we need to convert the discrete data into a 3D volumetric mesh associated with a formal topology-based representation. For this starting step, the required isosurface is extracted from the segmented grid using the *Flying Edges* (**FE**) algorithm [48]: a parallel version of the **MC** algorithm.

Some important properties of the built surfaces are expected. First, the algorithm must ensure that built surfaces are manifold and watertight to form closed volumes, and that every vertex must be positioned at the midpoint of cube edges, to ensure that two surfaces from different segments intersecting a same cube edge, share the same vertex. Second, the algorithm needs to prevent overlap in cube configurations with multiple segments, where several surface constructions are possible, and no hole should appear either (ambiguous configurations of **MC** [49]). Although another Marching Cubes implementation could have been chosen, **FE** algorithm meets all these requirements, it is parallelized and thus fast in terms of computation time, so it is a good candidate for our method.

The different steps of our method are presented in Figure 1 and the following gives its overview.

Starting from a reconstruction based on the **FE** algorithm that produces a continuous closed surface for each **ROI**, the first stage consists in a topological reconstruction to associate those generated surface to G-maps and consider them as volumes thereafter. Semantic information is stored to preserve the nature of the tissues and a cleaning process is applied to remove unwanted elements corresponding to artifacts from the acquisition (see Section 4.1).

During the second stage, segments are integrated in the same space (a single G-map) and then topologically linked by their common faces when they are adjacent. This step uses topological properties to detect and correct neighborhood inconsistencies, that are (i) empty spaces, more precisely, volumes that do not belong to any segment, which results in discontinuities between identified volumes and (ii) anatomical incorrect neighborhood. This step only requires topological and semantic information to make these corrections. Since no new elements are created, we do not encounter the case where the insertion of a vertex is too close to an existing one, and thus avoiding the numerical instabilities inherent in geometric calculations (see Section 4.2).

In the last stage, a transformation that preserves topology (homeomorphism) and semantic constraints is applied to compute a shape as close as possible to the original data (see Section 4.3). By deforming according to voxel information, we ensure a better fitting of the surface to the input data. This voxel information can be statistical data calculated from voxel intensity.

To summarize, our method produces automatically a partition of 3D space corresponding to a volumetric mesh from several segmented **ROI**; in other words, the mesh is free from empty spaces and volume overlaps. This partition shows explicit adjacency relationships defined by a formal topological model (G-map).

4. Topological reconstruction

In this section we describe the different steps of the method for constructing a 3D mesh following topological and semantic constraints. It first starts reconstructing and cleaning every segment before merging all of them together while applying topological operations to resolve inconsistencies. Finally, a fine geometric deformation is applied on the whole model to be as close as possible to the acquired data.

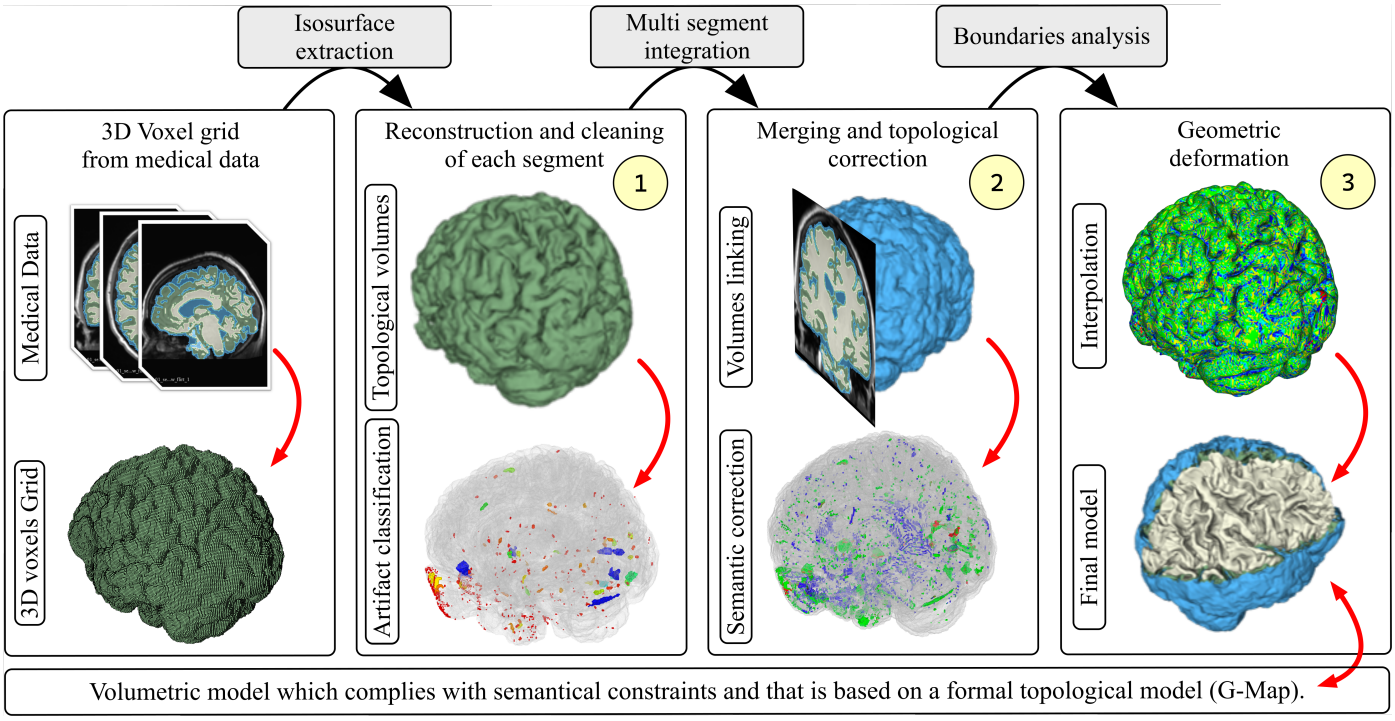


Fig. 1. Our reconstruction system is described as follows: (1) Topological reconstruction based on Marching Cubes and artifacts cleaning for each ROI. (2) Merging of all reconstructed segments, detecting and correcting semantic and neighborhood inconsistencies. (3) Continuous deformation of the curvature of the mesh to adjust its surface as closed as possible to the original data.

4.1. Reconstruction and cleaning of one segment

To obtain a clean volume representation of each segment, two steps are performed. First, each **ROI** is reconstructed independently. A **ROI** is usually composed of more than one volume, therefore each of its volumes is generated separately, geometrically and topologically. Then, a classification is applied to remove undesired volumes (e.g., artifacts) and keep only relevant ones.

4.1.1. Topological reconstruction of volumes

Starting from a segmented voxel grid of MRI images, the **FE** method is applied to reconstruct a closed surface triangle mesh (Figure 2) for each tissue. The voxel grid is divided as follows: Each voxel is labelled by a single value, either 0 if it is a background voxel or a number up to the number of segmented regions (information given by the segmentation). To reconstruct a segment, a corresponding black & white grid is automatically generated: each voxel is binary, depending on whether or not it belongs to the current segment.

The chosen reconstruction algorithm, **FE**, ensures that each voxel edge is only intersected once, to produce a manifold and watertight mesh from each segment. In fact, the input voxel grid given to **FE** is a grid composed of 0 and 1 values. An imposed isovalue of 0.5 during the reconstruction of the isosurface generates vertices at the midpoint of each **MC** intersected edge. More formally, the surface built by **FE** corresponds to the isosurface associated with isovalue 0.5, represented by the set of points $\{X \in \mathbb{R}^3 : F(X) = 0.5\}$, for a field sampled at each cube vertex, with only 0 and 1 values. Also, since the isovalue of the isosurface is different to the one associated to the segmentation voxel grid, triangles that have (at least) one null edge (at least two coincident vertices) are not created. Although such degenerate triangles do not raise issues during our reconstruction process, they are not desirable for further modeling operations (for instance, the cutting method described in Section 6). The choice of not dealing with geometric aspects (finding a more precise intersection position) at this point of the method ensures that we can make the merge phase possible in the next stage without dealing with computation accuracy.

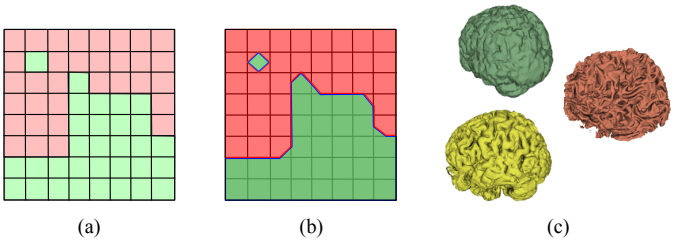


Fig. 2. In a classical way, the MC method is applied to the different regions (a), and produces (in 3D) a set of faces (b). Here, face's vertices are always placed at midpoint of edges of the MC grid (these edges link voxel centers). This process is applied independently to each ROI producing a 3D model of each structure (c).

Geometry can indeed be corrected, and even refined, after all topological corrections, to ensure a right match of the real structure shape. As explained in the **MC** method requirements in Section 3, the resulting model is guaranteed to have neither self-intersection nor overlap.

Once meshes are produced, their corresponding G-map is constructed by using the workflow described in [50]. This workflow applied to the surface mesh provided by the FE

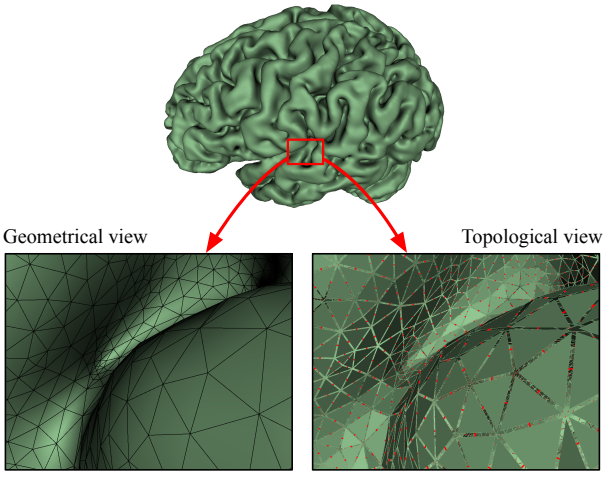


Fig. 3. Mesh is associated to the corresponding 3G-map. Topological view shows links of dimension 2 between faces.

method generates topologically closed, watertight volumes. In these volumes, adjacent faces are explicitly bound at their common edge by topological links of dimension 2 (Figure 3), called 2-links or α_2 involution (see Appendix A). Note at this point that inclusion links (volumes completely included in another volume), are not reconstituted.

4.1.2. Classification and cleaning

The second step consists in the removal of volumes corresponding to artifacts produced during the acquisition phase for each segment separately. At this stage, each volume (or 3-cell, see Appendix 2) is identified and recognized directly as a single connected component. The characteristics of a volume are local, so any geometric or topological properties, such as its size or its number of faces, can be computed locally. The computation time is thus strongly reduced.

To clean up the artifacts present in the reconstructed models of each tissue, a classification of the topological volumes is carried out. In our case, the term "classification" is used to separate volumes that are considered as artifacts, and the ones that have to be kept and also be labeled as *main components*. In particular, *main components* are determined as the largest objects in terms of volume and number of faces. In our case, smaller volumes located far from the main components are classified as artifacts, whereas larger volumes in close proximity to these primary elements are retained, as they are likely to result from a detachment or separation of said components. The distance between two volumes is calculated by computing the shortest Euclidean distance between any vertex coming from the first surface mesh and any vertex from the second. Figure 4 shows the result of the multi-criteria classification where the volumes that are considered as useless can be deleted. For instance, in accordance with medical experts, a volumes is considered "small", if its size is less than 4mm, and "distant" if the gap from a main component is greater than 2mm (in our case, these sizes respectively correspond to 4 and 2 voxels in the case of MRI images from 3T scanners). In other words, it is a conjunction of boolean expressions for each criterion. So in this case, if both condition are respected, it will give a high probability to keep elements that have been detach of a main component during the segmentation process. The removal of these unwanted volumes, seen as single connected components, relies on a local deletion (see Section Appendix A.2).

Finally, for each segment, a set of volumes is identified and associated to a semantic label corresponding to its anatomical structure. Without a topology-based model, these usually tedious treatments require a fine-tuning of lots of parameters. In our case, they are treated automatically while maintaining all topological and semantic properties. Note that some remaining connected components correspond to cavities that are handled in a further step (see Section 4.2.2).

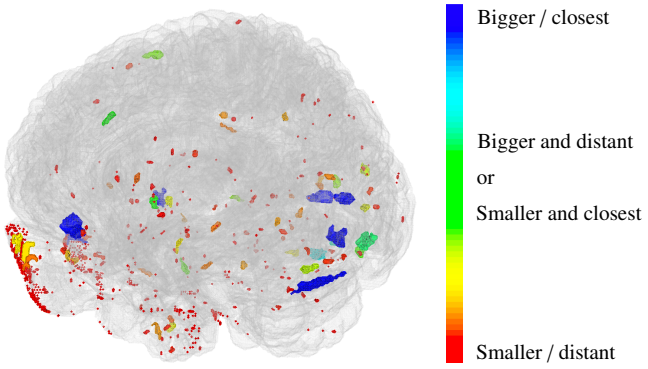


Fig. 4. Multi-criteria for artifacts detection: smaller volumes in size and distant from the main components are classified as artifacts and can be deleted.

4.2. Merging all segments

To construct a 3D space partition corresponding to a segmentation of the multiple input structures, all the possible sets of volumes corresponding to each segment are integrated in the same space. A first topological operation is applied: It aims at joining adjacent volumes by their identical faces, through links of dimension 3, called 3-links or α_3 involution (see Appendix A). Then defects such as discontinuities between volumes (those including faces with no α_3 link) are detected and corrected.

4.2.1. Adding topological links between adjacent volumes

The previous step has created a volumetric mesh with our topological model (G-map), possibly including several volumes, but rid of artifacts, for each segment. The end result of the merging process is a global, unique, G-map representation of the multi-object of a segmentation altogether (in other words, a volumetric partition).

Since the **FE** method reconstructs all segments from the same segmentation and with the same isovalue, this implies that there is no overlap between built volumes. Indeed, reconstructed surface meshes always intersect the **MC** grid edges at their midpoint (isovalue at 0.5, see Section 4.1). This property guarantees that adjacent volumes are built using the same vertices and faces in their adjacency zones (Figure 5.a). Therefore, the

merging process consists in identifying and linking the identical faces of different adjacent segments by α_3. Thus, each face of a segment is compared to the faces of the other segments (it is useless to test a segment with itself). To avoid error-prone equality tests between floating-point values, the matching test between two faces is made by comparing the identifiers of the **MC** edge supporting each of their vertices. If they match, this means that their volumes are adjacent and can be α_3-linked.

Figure 5.b shows also the topological sewing process corresponding to the segmentation present in Figure 2. Besides, faces with no match are left free (i.e. without adjacent volume). They are handled in the topological correction (Section 4.2.2).

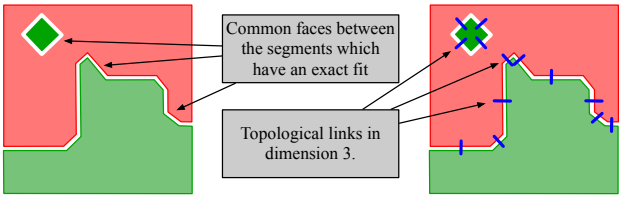


Fig. 5. Topological link of volumes corresponds to (a) identify and (b) link the identical faces between the different segments. In the following, topological links are not represented for a better readability.

4.2.2. Topological correction

During the merging process, the correctness of the adjacency relationships in the merged mesh has not been controlled. Indeed, some possible free spaces might be present in the model. This results from three possible causes. First, the segmentation can misclassified some voxels as "*background*" and, due to acquisition noise, some of these voxels might, incorrectly, correspond to cavities included in other segments. For a homogeneous treatment of topological correction, our method does not build the "background" segment and leave these cavities as empty spaces. Second, during the artifacts correction, some small volumes can be deleted, but, in the merged model, they can correspond to cavities enclosed in another segment. After deletion, only empty cavities remain in the containing segments. Third, as explained in the **MC** requirements, some ambiguous cube configurations including several segments may exhibit non covered area. These appear as empty spaces in the merged model. The goal of this step is to properly fill these spaces, with the correct semantic. The fill operation is only topological and locally computed, so our model is not prone to geometric accuracy issues (no insertion of new vertices which could be very closed to existing vertices).

Thanks to 3D topological neighborhoods, the spaces of unknown semantic are directly localized (they are defined by faces without α_3-links) and a topological closure operation (implemented over the generalized map algebra), applied on the mesh, permits to directly build the volume corresponding to the empty space. Then, these volumes are labeled "*unnamed*" and surrounding faces are no longer free. Using α_3-links to study the neighbourhood of "unnamed" volumes, we can classify them into three distinct cases (Figure 6).

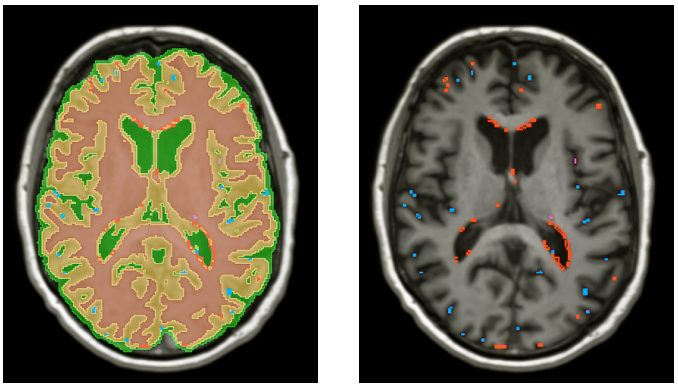


Fig. 6. Empty volumes identified during topological correction. (a) MRI image, segmented ROI and voxels marked in blue (case 1), red (case 2) and purple (case 3). (b) MRI image and voxels in blue (case 1), red (case 2) and purple (case 3).

In some cases, these inconsistencies are corrected by performing a local stochastic study. This process finds the tissue with which to fill this gap both by collecting and comparing the types of its adjacent tissues and by anatomical knowledge.

- Case 1: The neighborhood of an unnamed volume corresponds only to a single segment. This comes from the fact that, at least, one of the segments has not been topologically constrained in previous step. Each volume has its semantic modified by the one of the volumes that surrounds it, leading to its suppression (Figure 7).

- Case 2: The neighborhood of an unnamed volume corresponds to several segments and this volume could reconnect two volumes that must, by anatomical knowledge, be linked (Figure 8 - case 2).

- Case 3: The neighborhood of an unnamed volume corresponds to several segments without explicit anatomical knowledge to exploit. In this case, corrections are done by using a probabilistic information, as such as **PVE**[47] already explained in Section 3. The most probable segment (corresponding to the highest value in the set of voxels covered by the unnamed volume) is selected for reconnection (Figure 8 - case 3).

At this point, the most probable semantic content of each empty space has been identified and a new label corresponding to an anatomical structure has been applied to every unnamed volume. It is therefore possible to topologically merge these volumes with their neighbors with the same label, by erasing their common face(s).

Throughout the correction process, semantic and topological information is updated and remains consistent. At the end of this stage, we have a 3D space partition corresponding to a segmentation of multiple input structures. All volumes have a semantic information and the volumetric mesh is free from topological and geometric defects.

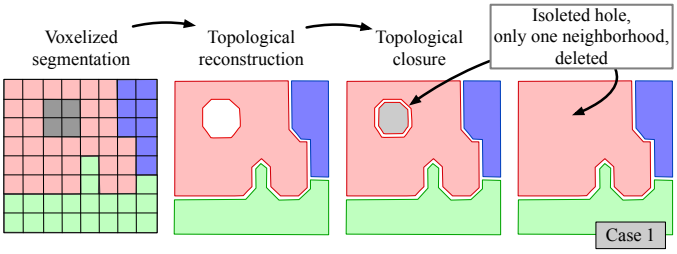


Fig. 7. Isolated cavity: "unnamed" volumes with only one neighbor segment that can be directly deleted.

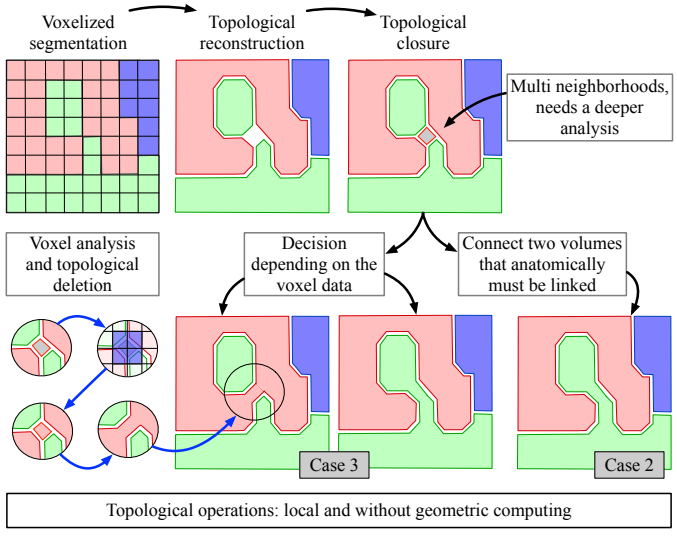


Fig. 8. Multi-connected cavity: after topological closure of the g-map associated to the mesh, if the generated volume is required to connect two volumes that, anatomically, must be linked, corresponding semantic is directly affected (case 2). Otherwise, a voxel analysis permits to define the right semantic (analysis on bottom left and case 3). Finally, useless faces are detected (face between the same structure) and deleted with a topological operation.

4.3. Geometric interpolation

By using **FE** to process the input data, the vertices of the generated meshes are conveniently computed at the midpoint of the **MC** grid edges that the surface to build intersects (Section 4.1). This procedure allows us to obtain an exact match between adjacent segments during the merging step (Section 4.2). However, this geometric positioning is arbitrary and only roughly approximates the real anatomical shape of segments (error is however bounded to half the size of voxels).

More precisely, the duality between the **MC** grid and that of the segmentation must be considered to understand how isosurface vertices should be positioned. Vertices of the **MC** grid are placed in the center of voxels from the segmentation. Therefore, an edge of the **MC** grid links centers of two adjacent voxels (Figure 9.b). The isosurface can actually intersect this edge anywhere between these centers.

To compute a smoother reconstruction without overlapping respecting the anatomical structure and the topological constraints, it is possible to use a statistical information or scalar value extracted from these voxels. For instance, voxel intensity could be used. In brain imaging, such information can be computed from **PVE**. By gathering locally the average probability of ownership from voxels which supports a vertex, the nature of the tissue of the highest value is also used to modify the position of the vertex. We use this data to define a new isovalue associated with the vertex and thus modify its coordinates. Let us call P, a vertex, defined as the intersection between the isosurface and the segment P_1P_2, where P_1 and P_2 are midpoints of adjacent voxels. The position of P can be approximated using linear interpolation. Actually, by knowing the voxels Vox_1 and Vox_2 of P_1 and P_2, we can have access to the PVE value of all tissues within, and the nature of the tissue in each voxel. Using these information, we compute the mean **PVE** of each tissue, assign the highest mean value to v_p and keep the associated tissue type. If the tissue type of the highest mean **PVE** value is the same as the one of Vox_1, we consider v_1 as 0 and v_2 as 1, and the other way around in the other case. Finally, by linear interpolation, we have $P = \frac{v_p - v_2}{v_1 - v_2} P_1 + \frac{v_p - v_1}{v_2 - v_1} P_2$.
It is important to note that the calculated isovalue v_p is strictly in between v_1 and v_2. This modification is carried out within the voxels, preventing self-intersections. As shown in Figure 9, the corresponding pair of voxels are stored for each vertex to keep the relation between the built mesh and the voxel grid.

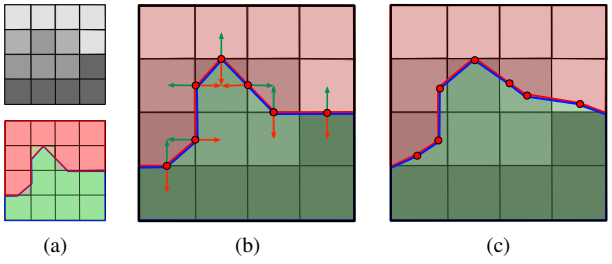


Fig. 9. Geometric warping: (a) original voxel data and corresponding segmentation and topological 3D reconstruction. (b) Displacement of each vertex by estimating its position based on an interpolation of voxel statistical information. (c) New geometric position of these vertices, closest to the real anatomy.

The adjacency relationships between segments established previously are kept during this deformation process. Neither volumes discontinuity nor overlap can appear. Indeed, if a vertex belongs to one or more shared faces of adjacent volumes, the shape of these volumes is modified when this (shared) vertex is moved. Figure 10 shows a visualization of the geometric warping displacement, the average variation is of the order of an seventh of a voxel length, depending of the direction.

5. Results and comparisons

This section presents results of our volumetric reconstruction process on synthetic MRI scans and real ones provided by our clinical partner (numbered $Brain_1, \ldots, Brain_{42}$). All real images are acquired on a Magnetom Skyra 3 Tesla (Siemens Healthineers, Erlangen, Germany). The sequence used is a 3D T1 MPRAGE 0.9 mm isotropic (TE= 2.41 ms, TR=1950 ms, TI= 816 ms, FOV = 256×213 mm2, matrix: 240×288, slices: 192, turbo factor: 224). Original DICOM data are converted to NIFTI to be used in our homemade automated pipeline. As a

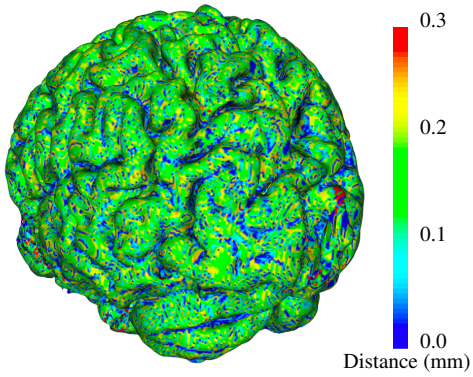


Fig. 10. Transformation of the vertices geometry due to the voxel data, so that the model is as close as possible to the real anatomy. The displacement distance cannot exceed half a voxel, i.e. 0.5mm in this case. Most variations are around 0.1/0.2mm.

post-processing step, FSL-BET [51] is applied to remove subcutaneous fat to create a mask of brain only. Then, FSL-FAST [52] is launched to obtain the 3 segmentation classes: (i) Cortex also known as Gray Matter, (ii) White Matter, (iii) Cerebral Spinal Fluid.

All statistics presented in this section have been produced on an Intel core(TM) I9-9900 3.10 GHz processor, with 8 cores and 96 GB of memory. Our method was implemented by integrating the kernel of a topological modeler *Moka* [53] within the open-source software *3D Slicer* [54], in the form of a module in C++ and using the VTK and ITK libraries. *Moka* provides the topological modeling part and all necessary topological operations, while *3D Slicer* is used to handle the MRI images (and the corresponding segmentation voxel grid) and to visualise all data (images and model). Both *Moka* and *3D Slicer* communicate to keep the model and the view up-to-date, especially after the use of a topological operation. Our work is compatible with the latest stable release of *3D Slicer* (5.6.2), but also with previous release up to 4.10. We also discuss the choice of the input iso-extraction method as well as comparisons between our reconstruction with those obtained from *FreeSurfer* and the CGAL surface reconstruction package.

5.1. Statistics and performance

Model Resolution	Number of faces $seg_1/seg_2/...$	Total time
Eye 66x56x29	41 292 / 57 100	36s
Kidney 84x79x127	82 500 / 162 892	43s
M. Brain 256x256x130	504 068 / 71 800	1min 18s
$Brain_3$ 240x288x192	899 480 / 1 390 964 / 852 804	8min 30s
$Brain_8$ 240x288x192	831 376 / 1 402 964 / 998 092	10min 02s
$Brain_{20}$ 240x288x192	974 876 / 1 426 144 / 822 284	10min 35s
$Brain_{27}$ 240x288x192	1 042 564 / 1 529 272 / 1 092 692	13min 52s
Brain7T 512x512x176	2 492 868 / 4 047 448 / 2 870 800	37min 06s

Table 1. Characteristic of our set of test models and total computation time of the entire method (section 4). Note that the number of surfaces generated is not directly linked to the grid resolution, but to the density and multiplicity of the segmented tissues. The lowest number of faces reconstructed and fastest total computation time are highlighted in blue, while the highest number of faces and slowest time are in red.

Table 1 presents the number of faces generated by the isosurface extraction method used as input to our topological reconstruction for each **ROI**, as well as the total computation time of the complete processing chain. We first show statistics on three synthetic examples: *Eye*, *Kidney*, *M.Brain*. We also report performances on a representative sample of our real dataset. $Brain_3$ and $Brain_{27}$ are respectively the samples with the most and least amount of reconstructed faces, while $Brain_8$ and $Brain_{20}$ are average in this characteristic. All statistics of this dataset are available as supplementary material. Finally, we present the statistics on a higher resolution voxel grids (and noisier segmentation) from a mri 7T scan as *Brain7T*.

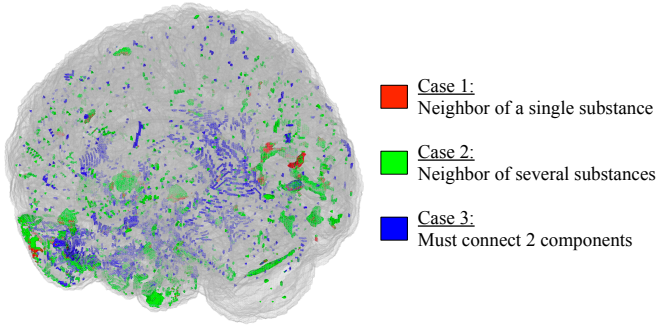


Fig. 11. Visualization of empty volumes classification as described in 4.2.2 in a complete reconstructed brain model composed of several structures.

Processing times for the reconstruction and cleaning (Section 4.1) for each segment are shown in Table 2. It can be seen that the different steps remain of the same order of magnitude in terms of computation time depending on the number of faces constituting the segment to reconstruct. The variations during the cleaning phase are due to the quality of the segmented data.

Table 3 presents the calculation times required to merge the reconstructed segments. The method detects and processes several thousand inconsistencies in a few minutes as illustrated by Figure 11. These *defect* volumes are then classified and corrected according to their semantics and topological neighborhood.

The medical accuracy of our model heavily relies on the input segmentation. However, regardless of the segmentation's quality—whether noisy or not—our reconstruction method consistently produces an accurate model aligned with it. Also, our model can detect anatomical problems (using the neighborhood relations between volumes and their semantic information), and automatically guide correction when possible.

5.2. Comparison with usual reconstruction methods

To justify our choice of using an isosurface generator and to evaluate the geometric and topologic qualities of our method, we compare the models reconstructed for a given segment using our process and three different methods of 3D reconstruction algorithm , that do not employ any topological solution for multi-segments cases.

- Dual Marching Cubes (**DMC**) method that uses an asymptotic decider and replaces triangular meshes by quad

Model	Number of reconstructed volumes	Reconstruction time	Number and percentage of deleted volumes	Classification and cleaning time	Process Time
Eye	62 / 328	2s / 5s	6 (10%) / 75 (23%)	2s / 1s	12s
Kidney	189 / 225	3s / 4s	74 (39%) / 102 (45%)	5s / 7s	25s
M. Brain	43 / 17	19s / 5s	4 (10%) / 2 (10%)	12s / 6s	1min 02s
$Brain_3$	365 / 704 / 386	47s / 1min 04s / 38s	0 / 211 (30%) / 89 (23%)	0s / 1min 32s / 46s	4min 47s
$Brain_8$	950 / 2222 / 541	35s / 1min / 41s	0 / 366 (16%) / 189 (35%)	0s / 1min 18s / 50s	6min 15s
$Brain_{20}$	256 / 501 / 316	39s / 59s / 35s	0 / 91 (18%) / 130 (41%)	0s / 1min 32s / 1min 04s	7min 16s
$Brain_{27}$	841 / 2959 / 1208	47s / 1min 02s / 47s	0 / 1298 (44%) / 225 (19%)	0s / 1min 35s / 49s	8min 04s
Brain7T	3 922 / 15 272 / 19 888	1min 35s / 2min 33s / 1min 53s	919 (23%) / 3239 (20%) / 11 261 (57%)	1min 44s / 2min 32s / 2min 51s	16min 38s

Table 2. Quantitative information and computation time of each step of Section 4.1 with our dataset. For *real brain models*, the lowest number of reconstructed volumes, percentage of deleted ones, fastest reconstruction and total computation time are highlighted in blue, while the highest are in red.

Model	Semantic/Topologic process			Voxels modified	Process time
	$Case_1$	$Case_2$	$Case_3$		
Eye	31	0	569	717	17s
Kidney	22	0	569	441	18s
M. Brain	39	0	166	626	17s
$Brain_3$	144	1688	1840	1329	3min 27s
$Brain_8$	155	4045	1798	2384	3min 27s
$Brain_{20}$	68	1698	1533	916	3min 03s
$Brain_{27}$	292	4685	3668	3329	5min 32s
Brain7T	3 493	17 872	4 907	23 127	18min 22s

Table 3. Quantitative information and computation time of each step of the method described in Section 4.2 with our dataset. Voxels corresponding to each "unnamed" volume are modified according to their semantic/topological case. For real brain models, lowest number of applied corrections and total computation time are highlighted in blue, while the highest are in red.

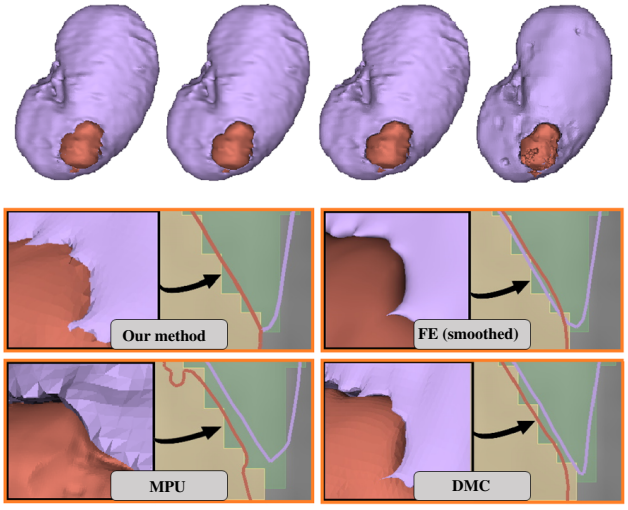


Fig. 12. Reconstruction of a kidney with a tumor from the 2019 KiTS Challenge [55, 56]: on top, the built 3D models, from left to right: our method, FE(smoothed), DMC, MPU. By using MPU or DMC, an empty space between the surface of the reconstructed kidney and tumor is exhibited, while with FE, overlaps appear due to surface smoothing (FE, DMC and MPU do not use any topological solution).

meshes which tends to eliminate the poorly-shaped triangles occasionally present in **MC** surfaces [31];

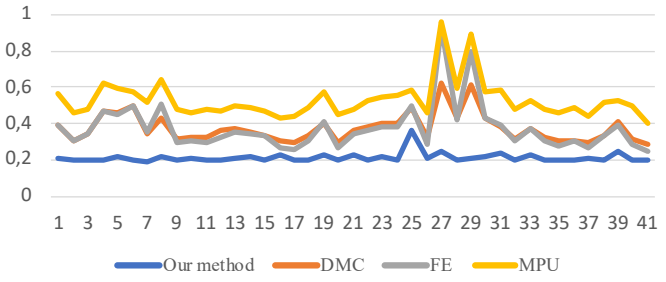
- Multi-level Partition of Unity (**MPU**) method that uses implicit models [14] and interprets the contours (i.e., segmentation boundary inside the volume) as points in $\mathbb{R}^3$;

- Flying Edges (**FE**) used at the start of our process. It is interesting to study the differences between **FE** (especially when the built mesh is smoothed) and our method due to the geometric warping applied at the end of our process.

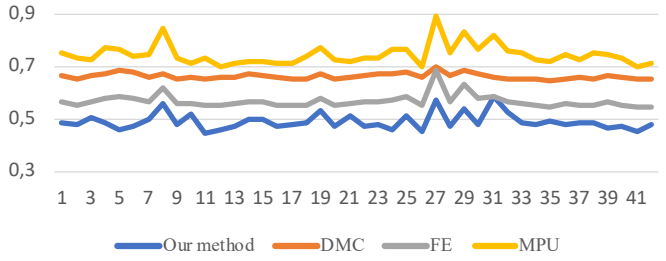
We first compare these methods from a topological point of view. As shown in Figure 12, smooth **FE**, **MPU** and **DMC** leads to inaccurate connectivity between elements (disjointed volumes or, on the contrary, overlaps), while our model preserves the one from the segmentation. More precisely, smooth **FE**, **MPU** and **DMC** are not directly usable to group and merge several volumes together, because of their construction system that does not produce faces allowing a direct topological linking of adjacent volumes. Indeed, the main source of overlapping is the post-processing step performed by counterpart approaches after surface extraction. That is why, **FE** (without smoothing) does not produce overlap/discontinuity (since any surface that crosses a given cube edge always intersect it at its midpoint, surfaces from different segments match exactly). Nevertheless, this method still exhibits unidentified empty volumes that our algorithm fills in. Topologically speaking, our model gives better results than **FE**, **DMC** and **MPU** since it does not include empty spaces, overlaps and artifacts.

As a comparison from a geometrical point of view, the figure 13 shows the standard deviation and arithmetic mean of the distance computation from the object to the segmentation done on our real medical dataset (forty brains). To obtain the distance value, we generate a point cloud corresponding to the midpoint of each voxel of the segmentation, and compute for each point of the reconstructed mesh the shortest Euclidean distance to the point cloud generated from the segmentation. It can be noticed that our method is better when compared to **FE**, **DMC** and **MPU**. This behavior is independent of the complexity of the model or the noise level since, for all forty brains, our method still gives better results. The goal of the proposed method is to follow, as closely as possible, the segmentation used as entry and supposed to be correct. Indeed, it is important to note that the reconstructed model is completely based on the segmentation result and can build incorrect tissue surfaces if the segmentation is wrong.

Finally, to visualize the impact of the geometric deformation step (see Section 4.3), we propose to use a synthetic object with a known geometry, namely a sphere. The aim is to com-



(a) Standard deviation



(b) Arithemic mean

Fig. 13. The standard deviation (a) and arithmetic mean (b) of error values for all forty cerebral cortices reconstructed with our method, FE, DMC and MPU.

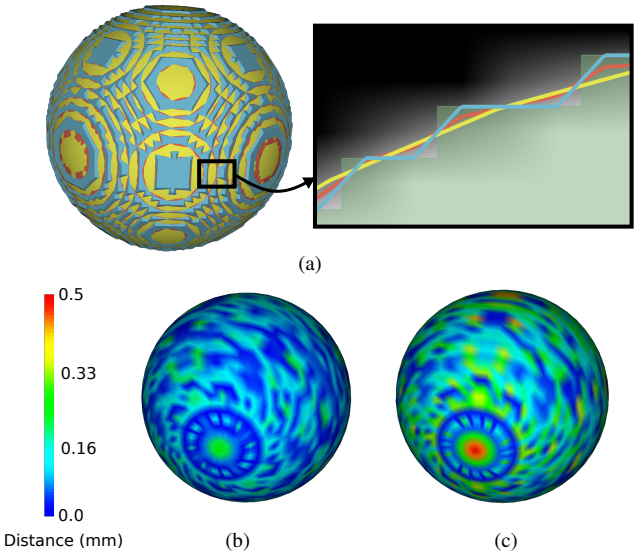


Fig. 14. Test of geometric warping on a discretized sphere: (a) superposition of the three 3D models (yellow for the initial sphere, blue for FE's one and orange for ours); calculation of the Euclidean distance between the reconstructed models from the geometry of the real sphere (b) for our method and (c) for FE.

pare the Euclidean distance between the objects reconstructed by our method and then by **FE** in comparison to the sphere's real geometry. To achieve this, the sphere is discretized into a regular grid: voxels that are entirely contained within the sphere have the value 1, those entirely outside have 0. For a voxel partially covered by the sphere, a coverage rate calculation (statistical value) is performed, based on a random selection of several thousand points located inside this voxel to calculate the ratio of the number of points located inside the sphere. This ratio is stored as the density associated with the voxel. Figure 14 shows the result of the comparison between our reconstructed model and the sphere, and similarly for the FE. We notice that our model is closer to the real geometry, thanks to the statistical value used in the geometric deformation step (the majority of distances are below 0.2mm, compared to 0.33mm for the **FE**).

5.3. Comparison with FreeSurfer

FreeSurfer is a set of software tools for the study of cortical and subcortical anatomy [7]. Currently, it is the main framework for neuroscientists to take benefit from 3D [57, 58]. In its cortical surface reconstruction stream, provided tools construct models of the boundary between white matter and cortical gray matter as well as the pial surface [8]. The volume-based stream is designed to preprocess MRI volumes and label subcortical tissue classes. A high dimensional nonlinear volumetric alignment is performed on a given atlas and volumes are labeled (fully described in [59]).

Qualitatively, the *Freesufer* reconstruction of the cortical area is accurate thanks to the use of data atlases. However, as the method is based on an independent reconstruction of each hemisphere, it induces a high degree of inaccuracy at the junction of the two lobes (Figure 15). The tissues are ultimately constructed independently and it is not possible to easily calculate the topological neighborhood between the meshes of the different structures. Moreover, this method can only produce a single model representing the mixture of cortical and subcortical interfaces. It cannot discriminates between all brain segments. An automatic cleaning process is available, but in the case of excessively noisy data, manual intervention is required and leads to a tedious semi-automatic process for reconstruction.

When compared to FreeSurfer, our method not only preserves distinctions between all brain segments but also outperforms FreeSurfer in terms of computation time. Few minutes are needed to reconstruct the whole brain, compared to several hours with FreeSurfer pipeline.

5.4. Comparison with 3D Mesh generation of CGAL

The Computational Geometry Algorithm Library (CGAL) [37] is a software project that provides easy access to efficient and reliable geometric algorithms used in many fields. Among the available operations, we are interested in the *3D mesh generation* [38]. This package is designed to generate a 3D mesh from a point cloud [60]. A Delaunay refinement is applied to produce a volumetric tetrahedral mesh. The main difference with a standard Delaunay triangulation is the addition of Steiner points to the input data to improve the quality of the final mesh. The reconstruction is highly customizable with different parameters for surface *facets* and mesh cells (more details [61]). The result is described as a 3D model based on tetrahedra, with much less information than our topological model. It can be converted to a surface model (a *FaceGraph*) by extracting triangles and describing their connectivity using a refined version of half-edges (a topological model used for 2D manifolds only) to describe faces.

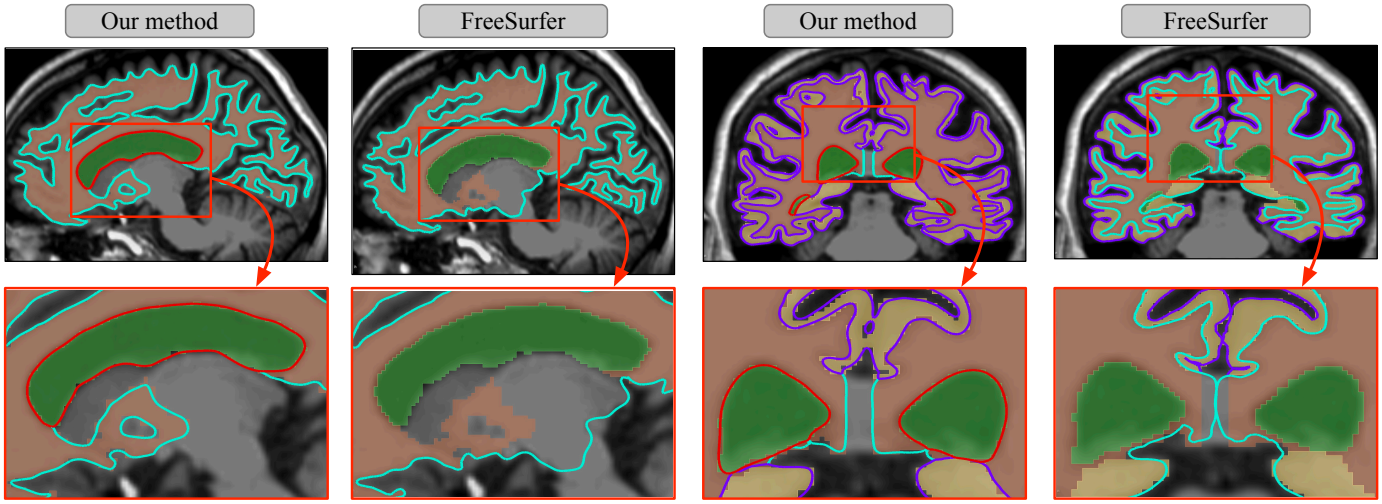


Fig. 15. Comparisons between our method and FreeSurfer's on two cross-sectional views of the brain (top row) and closeups as insets (bottom row). Starting from the same segmentation, we note that FreeSurfer is unable to distinguish between ventricules (green) and the white matter (orange) whereas ours clearly shows the contours of the two segments (left). FreeSurfer also suffers from surface overlaps between the two lobes (right).

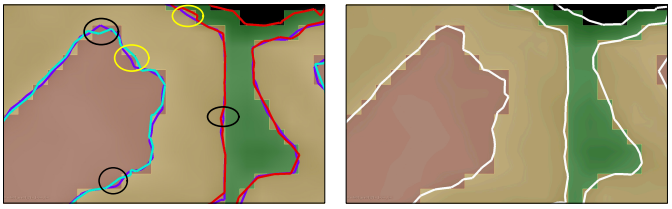


Fig. 16. Examples of the reconstruction of all 3 segments by CGAL 3D Mesh Generation : (a) ROIs independently reconstructed (the color matching of the surfaces are: orange for cerebral spinal fluid, white for white matter and green for gray matter) showing overlaps and empty spaces (respectively in yellow and black); (b) All ROIs at once, in one mesh, without tissue differentiation.

Using CGAL, a first possible strategy consists in building each segment one after the other. Nevertheless, figure 16.a shows that the different **ROI** of the segmentation cannot be reconstructed separately because it suffers from overlaps and gaps created by unadjusted surfaces. Therefore, the mesh must be reconstructed in one step, including every **ROI**, which is another strategy proposed by CGAL. Since all segments are built using the same Delaunay refinement, this leads to the absence of voids and overlaps as in Figure 16.b. However, the reconstruction process does not produce a mesh for each segment, but, instead, a global mesh that merges all segments and loses tissue information. The tissues cannot be separated in the obtained model. Besides, the refinement process (additional Steiner points) aims at producing a high quality 3D mesh but approximates the shapes and leads to an inaccurate anatomical reconstruction. To sum up, due to loss of semantic information and inaccuracy of the geometry, CGAL's reconstruction method appears unsuitable for medical uses.

6. Case study: Magnetic Resonance Spectroscopy

To highlight the benefits of the models produced by our reconstruction method, we present a use of these models for the visualization of data acquired by Magnetic Resonance Spectroscopy (MRS). It is a non-invasive imaging tool associated with MRI that provides metabolism data of covered tissues. MRS yields measurements with a much coarser resolution than MRI leading to data that are hard to interpret since the representation of the resulting values does not take into consideration the heterogeneity of tissues in an acquisition volume.

To cross-reference the anatomical and physiological modalities, the spectroscopic grid needs to be superimposed on the anatomical 3D model. To do this, a 3D model reconstructed using our method is perfectly suited, thanks to its geometric precision and its topological and semantic information. MRS volume is registered with MRI one, to work in the same space. However, the acquired spectroscopy grid is not, in practice, aligned with the corresponding MRI images. Working with a continuous model is more suitable, because we become independent from discretization, making it easier to compute intersections more exactly, and as we are closer to real data, it enables us to achieve smaller resolutions and give more information in this case study to practitioners. From our volumetric reconstruction, we apply successive planar cuts to extract the parts of the anatomical model corresponding to each voxel in the spectroscopic grid (Figure 17). This is a complex process due to the need to preserve topological and semantic information and manage volume inclusions, while taking into account accuracy issues [62, 63]. The strength and robustness of our model make it possible to overcome these problems by following a refinement process. We use the process described by Belhaouari et al. [50] propagating planar intersections between adjacent tissues using a topologically-based approach.

At the end of the process, for each voxel of the MRS grid we obtain the covered brain sub-region (as seen in Figure 17). This multi-modal modeling allows doctors to study the physiological

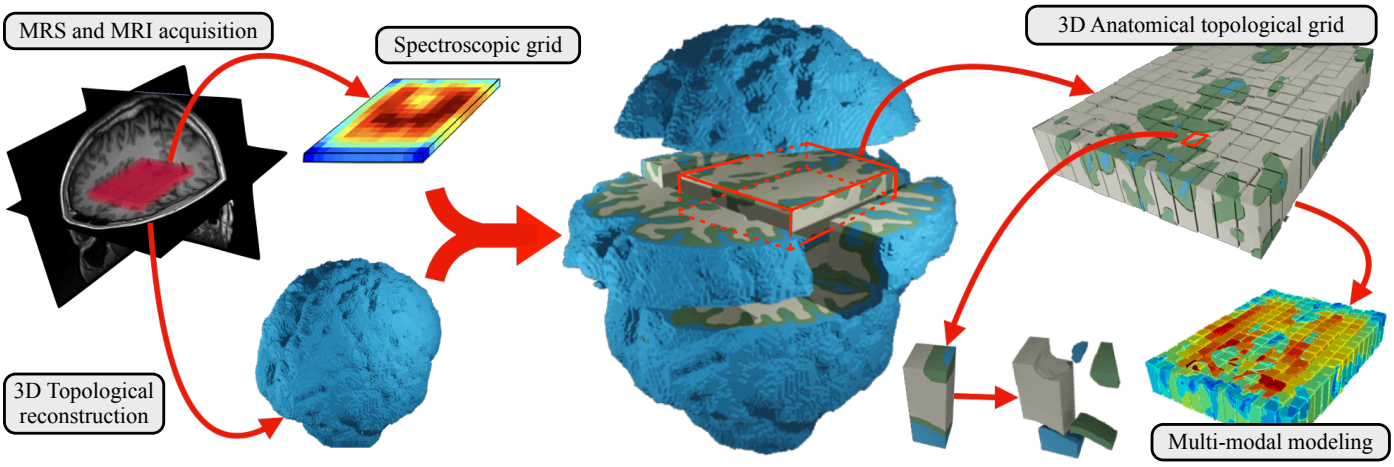


Fig. 17. 3D Anatomical topological construction matching the spectroscopy grid. Using the information of the spectroscopy grid and its voxels, the 3D brain model is split in accordance with the number of rows and columns of the initial grid. Resulting model is composed of identifiable volumes, each from a specific spectroscopy voxel and brain tissues, and used for multi-modal modeling.

distributions in the different anatomical tissues and evaluate if the measured metabolisms correspond to standard values for the covered types of tissue, or not [64]. This is a major scientific challenge for detection and analysis of brain tumors.

7. Conclusions

Our main objective is to produce high-quality volumetric meshes from discrete 3D acquisitions. In this context we presented a new reconstruction method to reconstruct a partition of space from a multi-label segmentation. A topologically consistent process is applied to merge efficiently every cleaned segment in a single volumetric model. Topological corrections are also applied using semantic data followed by a geometric warping that deforms the shape of the reconstructed segments while not altering the overall topology.

The proposed reconstruction method is a general approach that is not tied to our implementation choices. Another topological structure than Generalized-maps can be considered as long as it allows *sewing* volumes together and guarantees adjacency relations (e.g. oriented maps [43]). In the same way, using another iso-extraction method could be considered if it shares the criteria discussed in Section 5.2. A potential suitable candidate for future work could be some variants from the surface-net approaches [36].

In the future, we aim at applying our process to non-healthy brains with tumors. Since our volumes are topologically correct, we could track their evolution over time and study it on the concrete case of cancerous tumors. As stated in the article, these topologically-correct volumes could help doctors refine the measurements taken by low-resolution acquisition systems (as in the case of spectroscopic analysis) by improving data interpolation across volumes instead of coarse voxels for instance. Finally, a reconstruction pipeline should be scalable so it would be interesting to carry out a scale-up study to investigate how topological formalisms can be adapted to out-of-core approaches.

References

[1] Höhne, K, Bomans, M, Pommert, A, Riemer, M, Schiers, C, Tiede, U, et al. Voxel-based visualization of medical images in three dimensions. Medical Images: Formation, Handling and Evaluation 1992;98:145–158.
[2] Kong, T, Rosenfeld, A. Digital topology: Introduction and survey. Computer Vision Graphics Image Processing 1989;48:357–393. doi:10.1016/0734-189X(89)90147-3.
[3] Maret, D, Telmon, N, Peters, O, Lepage, B, Treil, J, Inglèse, J, et al. Effect of voxel size on accuracy of 3d reconstructions with cone beam ct. Dento maxillo facial radiology 2012;41:649–55.
[4] Santa Cruz, R, Lebrat, L, Bourgeat, P, Clinton, F, Fripp, J, Salvado, O. Deepcsr: A 3d deep learning approach for cortical surface reconstruction. Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision 2021;.
[5] Lebrat, L, Santa Cruz, R, de Gournay, F, Fu, D, Bourgeat, P, Fripp, J, et al. Corticalflow: A diffeomorphic mesh deformation module for cortical surface reconstruction. In: Advances in Neural Information Processing Systems; vol. 34. 2021,.
[6] Bongratz, F, Rickmann, AM, Pölsterl, S, Wachinger, C. Vox2cortex: Fast explicit reconstruction of cortical surfaces from 3d mri scans with geometric deep neural networks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022;.
[7] Dale, A, Fischl, B, Sereno, M. Cortical surface-based analysis. i. segmentation and surface reconstruction. NeuroImage 1999;9:179–194. doi:10.1016/j.neuroimage.2012.09.050.
[8] Fischl, B, Dale, A. Measuring the thickness of the human cerebral cortex from magnetic resonance images. Proceedings of the National Academy of Sciences of the United States of America 2000;97:11050–11055. doi:10.1073/pnas.200033797.
[9] Ségonne, F, Dale, AM, Busa, E, Glessner, M, Salat, D, Hahn, HK, et al. A hybrid approach to the skull stripping problem in mri. Neuroimage 2004;22(3):1060–1075.
[10] Fischl, B, Salat, DH, Busa, E, Albert, M, Dieterich, M, Haselgrove, C, et al. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. Neuron 2002;33(3):341–355.
[11] MacDonald, D, Kabani, N, Avis, D, Evans, A. Automated 3-d extraction of inner and outer surfaces of cerebral cortex from mri. NeuroImage 2000;12:340–356. doi:10.1006/nimg.1999.0534.
[12] Cheng, Q, Sun, P, Yang, C, Yang, Y, Liu, PX. A morphing-based 3d point cloud reconstruction framework for medical image processing. Computer methods and programs in biomedicine 2020;193.
[13] Kimia, B, Tannenbaum, A, Zucker, S. Shapes, shocks, and deformations i: The components of two-dimensional shape and the reaction-diffusion space. International journal of computer vision 1995;15(3).
[14] Braude, I, Marker, J, Museth, K, Nissanov, J, Breen, D. Contour-based surface reconstruction using mpu implicit models. Graphical mod-

els 2007;69(2):139–157. http://doi.org/10.1016/j.gmod.2006.09.007.

[15] Lu, S. A novel method for 3d reconstruction of blood vessels. Journal of Physics: Conference Series 2021;1732. doi:10.1088/1742-6596/1732/1/012110.

[16] Kigka, VI, Rigas, G, Sakellarios, A, Siogkas, P, Andrikos, IO, Exarchos, TP, et al. 3d reconstruction of coronary arteries and atherosclerotic plaques based on computed tomography angiography images. Biomedical Signal Processing and Control 2018;40:286–294. doi:10.1016/j.bspc.2017.09.009.

[17] Han, X, Pham, D, Tosun, D, Rettmann, M, Xu, C, Prince, J. Cruise: cortical reconstruction using implicit surface evolution. NeuroImage 2004;23:997–1012. doi:10.1016/j.neuroimage.2004.06.043.

[18] Yamina, Y, Lila, M, Zouagui, T, Jennane, R. A topology constrained geometric deformable model for medical image segmentation. Biomedical Signal Processing and Control 2020;64:997–1012. doi:10.1016/j.bspc.2020.102299.

[19] Bazin, PL, Pham, DL. Topology correction of segmented medical images using a fast marching algorithm. Computer methods and programs in biomedicine 2007;88(2):182–190.

[20] Han, X, Xu, C, Prince, JL. A topology preserving level set method for geometric deformable models. IEEE Transactions on Pattern Analysis and Machine Intelligence 2003;25(6):755–768.

[21] Ségonne, F. Active contours under topology control—genus preserving level sets. International Journal of Computer Vision 2008;79(2):107–117.

[22] Henschel, L, Conjeti, S, Estrada, S, Diers, K, Fischl, B, Reuter, M. Fastsurfer - a fast and accurate deep learning based neuroimaging pipeline. NeuroImage 2020;219:117012. doi:10.1016/j.neuroimage.2020.117012.

[23] Park, J, Florence, P, Straub, J, Newcombe, R, Lovegrove, S. Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019, p. 165–174. doi:10.1109/CVPR.2019.00025.

[24] Cruz, R, Lebrat, L, Bourgeat, P, Fookes, C, Fripp, J, Salvado, O. Deepcsr: A 3d deep learning approach for cortical surface reconstruction. In: WACV. 2020, p. 755–768.

[25] Ma, Q, Robinson, EC, Kainz, B, Rueckert, D, Alansary, A. Pialnn: A fast deep learning framework for cortical pial surface reconstruction. In: International Workshop on Machine Learning in Clinical Neuroimaging. Springer; 2021, p. 73–81.

[26] Lorensen, W, Cline, H. Marching cubes: A high resolution 3d surface construction algorithm. ACM SIGGRAPH Computer Graphics 1987;21:163–169. doi:10.1145/37401.37422.

[27] Nielson, G. On marching cubes. IEEE Transactions on Visualization and Computer Graphics 2003;9:283–297. doi:10.1109/TVCG.2003.1207437.

[28] Custodio, L, Etiene, T, Pesco, S, Silva, C. Practical considerations on marching cubes 33 topological correctness. Computers and Graphics 2013;37:840–850. doi:10.1016/j.cag.2013.04.004.

[29] Grosso, R. Construction of topologically correct and manifold isosurfaces. Computers and Graphics 2016;35:187–196. doi:10.1111/cgf.12975.

[30] Nielson, G. On marching cubes. Dual Marching Cubes 2004;:489–496doi:10.1109/VISUAL.2004.28.

[31] Grosso, R, Zint, D. Parallel reconstruction of quad only meshes from volume data. In: Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. SciTePress; 2020, p. 102–112. doi:10.5220/0008948701020112.

[32] Liao, Y, Donne, S, Geiger, A. Deep marching cubes: Learning explicit surface representations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018, p. 2916–2925.

[33] Chen, Z, Zhang, H. Neural marching cubes. ACM Transactions on Graphics (TOG) 2021;40(6):1–15.

[34] Lazar, R, Dym, N, Kushinsky, Y, Huang, Z, Ju, T, Lipman, Y. Robust optimization for topological surface reconstruction. ACM Transactions on Graphics (TOG) 2018;37(4):1–10.

[35] Brüel-Gabrielsson, R, Ganapathi-Subramanian, V, Skraba, P, Guibas, LJ. Topology-aware surface reconstruction for point clouds. In: Computer Graphics Forum; vol. 39. Wiley Online Library; 2020, p. 197–207.

[36] Frisken, SF. Surfacenets for multi-label segmentations with preservation of sharp boundaries. Journal of Computer Graphics Techniques Vol 2022;11(1).

[37] The CGAL Project, . CGAL User and Reference Manual. 4.3 ed.; CGAL Editorial Board; 2013. URL: http://doc.cgal.org/4.3/Manual/packages.html.

[38] Alliez, P, Jamin, C, Rineau, L, Tayeb, S, Tournois, J, Yvinec, M. 3D mesh generation. In: CGAL User and Reference Manual; 5.4 ed. CGAL Editorial Board; 2022,URL: https://doc.cgal.org/5.4/Manual/packages.html#PkgMesh3.

[39] Lewiner, T, Lopes, H, Vieira, AW, Geovan, T. Efficient implementation of marching cubes' cases with topological guarantees. Journal of graphics tools 2003;8(2):1–15.

[40] Paiva, A, Lopes, H, Lewiner, T, de Figueiredo, LH. Robust adaptive meshes for implicit surfaces. 19th Brazilian symposium on computer graphics and image processing 2006;:205–212.

[41] De Araújo, BR, Lopes, DS, Jepp, P, Jorge, JA, Wyvill, B. A survey on implicit surface polygonization. ACM Computing Surveys (CSUR) 2015;.

[42] Kong, T, Rosenfeld, A. Digital topology: Introduction and survey. Computer Vision, Graphics, and Image Processing 1989;48(3):357–393. URL: https://doi.org/10.1016/0734-189X(89)90147-3.

[43] Damiand, G, Lienhardt, P. Combinatorial Maps: Efficient Data Structures for Computer Graphics and Image Processing. A K Peters/CRC Press; 2014.

[44] Baumgart, B. A polyhedron representation for computer vision. In: AFIPS Nat. Conf. Proc. 44, AFIPS Press, Alrington, Va. 1975, p. 589–596. URL: https://doi.org/10.1145/1499949.1500071.

[45] Weiler, K. The radial-edge data structure: a topological representation for non-manifold geometry boundary modeling. In: Proc. IFIP WG 5.2 Working Conference, Rensselaerville, USA. 1986, p. 3–36.

[46] Lienhardt, P. N-dimensional generalized combinatorial maps and cellular quasi-manifolds. International Journal on Computational Geometry and Applications 1994;4:275–324. doi:10.1142/S0218195994000173.

[47] Ballester, MG, Zisserman, A, Michael, B. Estimation of the partial volume effect in mri. Medical Image Analysis 2002;6:389—-405. doi:10.1016/S1361-8415(02)00061-0.

[48] Schroeder, W, Maynard, R, Geveci, B. Flying edges: A high-performance scalable isocontouring algorithm. 5th IEEE Symposium on Large Data Analysis and Visualization 2015;:33–40doi:10.13140/RG.2.1.3415.9609.

[49] Chernyaev, E. Marching cubes 33: Construction of topologically correct isosurfaces. No CERN-CN-95-17 1995;.

[50] Belhaouari, H, Horna, S. Reconstruction of volumes from soup of faces with a formal topological approach. Computer-Aided Design and Applications 2018;16. doi:10.14733/cadaps.2019.972-984.

[51] Smith, S. Fast robust automated brain extraction. Human Brain Mapping 2002;17:143–155. doi:10.1002/hbm.10062.

[52] Y. Zhang, MB, Smith, S. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. IEEE Trans Med Imag 2001;20:45–57. doi:10.1109/42.906424.

[53] Vidil, F, Damiand, G, Dexet-Guiard, M, Guiard, N, Ledoux, F, Fousse, A, et al. Moka: 3d topological modeler. 2002. https://projet.liris.cnrs.fr/moka/index.php.

[54] Fedorov, A, Beichel, R, Kalpathy-Cramer, J, Finet, J, Fillion-Robin, JC, Pujol, S, et al. 3d slicer as an image computing platform for the quantitative imaging network. Magn Reson Imaging 2012;30:1323–1341. doi:10.1016/j.mri.2012.05.001.

[55] Heller, N, Sathianathen, N, Kalapara, A, Walczak, E, Moore, K, Kaluzniak, H, et al. The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. arXiv preprint arXiv:190400445 2019;.

[56] Heller, N, Isensee, F, Maier-Hein, KH, Hou, X, Xie, C, Li, F, et al. The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge. Medical Image Analysis 2020;:101821.

[57] Grossner, E, Bernier, R, Brenner, E, Chiou, K, Hillary, F. Prefrontal gray matter volume predicts metacognitive accuracy following traumatic brain injury. Neuropsychology 2018;32:484–494. doi:10.1037/neu0000446.

[58] Grossner, E, Brenner, E, Bernier, R, Rabinowitz, A, Hillary, F. Metacognition, aging, and tbi metacognition in individuals aging with

traumatic brain injury; 2021. doi:10.31234/osf.io/967zj; (preprint).
[59] Fischl, B, van der Kouwe, A, Destrieux, C, Halgren, E, Ségonne, F, Salat, DH, et al. Automatically Parcellating the Human Cerebral Cortex. Cerebral Cortex 2004;14(1):11–22. URL: https://doi.org/10.1093/cercor/bhg087. doi:10.1093/cercor/bhg087.
[60] Jamin, C, Alliez, P, Yvinec, M, Boissonnat, JD. Cgalmesh: a generic framework for delaunay mesh generation. ACM Transactions on Mathematical Software (TOMS) 2015;41(4):1–24.
[61] Rineau, L, Yvinec, M. CGAL 3d surface mesh generation user manual. https://doc.cgal.org/latest/Surface_mesher/index.html#Chapter_3D_Surface_Mesh_Generation; ???? Accessed: 2023-10-05.
[62] Fang, X, Desbrun, M, BAO, H. Topocut: fast and robust planar cutting of arbitrary domains. ACM Transactions on Graphics 2022;41(4):1–15.
[63] Trettner, P, Nehring-Wirxel, J, Kobbelt, L. Ember: exact mesh booleans via efficient & robust local arrangements. ACM Transactions on Graphics (TOG) 2022;41(4):1–15.
[64] Quadrelli, S, Mountford, C, Ramadan, S. Hitchhiker's guide to voxel segmentation for partial volume correction of in vivo magnetic resonance spectroscopy. Magnetic Resonance Insights 2016;9.
[65] Damiand, G, Lienhardt, P. Removal and contraction for n-dimensional generalized maps. In: Procedings of 11th Discrete Geometry for Computer Imagery; vol. 2886 of *LNCS*. 2003, p. 408–419.

Appendix

Appendix A. Notations and additional notions

Appendix A.1. Generalized maps

Generalized maps define the topological structure of geometric objects. They are based on a single type of basic elements (called *darts*) and involutions (called α) defined on these darts represent object cells and adjacency/incidence relationships.

Each involution α_i, with $0 \leq i \leq n$ (n being the dimension of the considered space), represents the adjacency relationships between *i-dimensional* cells (called *i*-cells). α_0 represents link between two vertices (0-cells), α_1 links two edges (1-cells), α_2 links two faces (2-cells) and α_3 links two volumes (3-cells) (cf. FigureA.18).

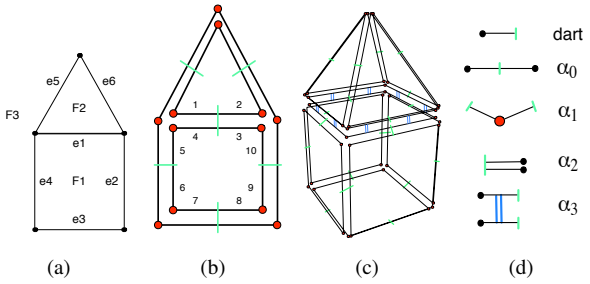


Fig. A.18. n-Gmaps representation: (a) 1 2D object containing 3 faces, 6 edges and 5 vertices; (b) Corresponding generalized map: the set of darts {1,2,3,4} represents edge e_1, the set of darts {3,4,5,6,7,8,9,10} represents face F_1; (c) 3D object composed of 2 volumes represented with 3-Gmap, volumes are linked by α_3; (d) Symbolic representation of involutions. In most cases, α_0 links are not represented for the sake of readability.

Definition 1 (Generalized map [46]). *A generalized map in dimension $n \geq 0$ (or n-Gmap) is an algebra $G = (D, \alpha_0, ..., \alpha_n)$, where:*

- *D is a finite set of darts;*
- *$\alpha_0, ..., \alpha_n$ are involutions (An involution f is a one to one mapping such that: $f^2 = Id$)*
- *$\alpha_i\alpha_j$ is an involution $\forall i, j$ such that $0 \leq i < i + 2 \leq j \leq n$.*

Generalized maps are defined homogeneously in any dimension, so that operations implemented in 2D can be naturally extended in 3D or in higher dimension. In addition, all descriptions made in dimension 2 are extensible to dimension 3.

Two darts d and d' are *i-sewed* if $\alpha_i(d) = d'$ with $d \neq d'$, and d is *i-free* if $\alpha_i(d) = d$. A dart d is *i-free* when it belongs to a boundary of an *i*-cell, i.e. d belongs to an *i*-cell which has no other *i*-cell adjacent along this dart. A n-Gmap is *with boundary* (or *open*) if it contains some free darts; it is *closed* otherwise.

The *orbit* associated with a given dart d and a set of involutions is $\langle f_1, ..., f_k \rangle(d)$, the set of all the darts obtained starting from d, and using any composition of the given involutions. Any *i*-cell is a particular orbit composed with all involutions except α_i. Two cells are incident when their intersection is non empty, i.e. when there is at least one dart shared by the two cells. Two *i*-cells c_1 and c_2 are adjacent if there is one $(i-1)$-cell incident to both c_1 and c_2. In Figure A.19, the different cells which compose a 3-Gmap are represented.

Definition 2 (*i*-cell). *Let $d \in D$, $N = \{0, \ldots, n\}$ and let $i \in N$. The i-cell containing dart d is the orbit $\langle\rangle_{N-\{i\}}(d) = \langle \alpha_0, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n \rangle(d)$.*

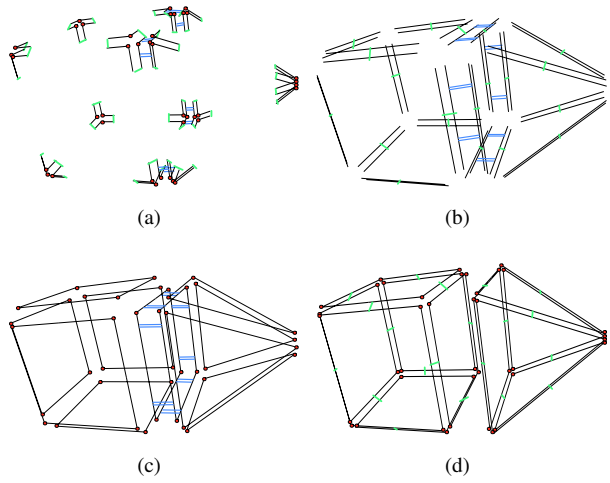


Fig. A.19. Each cell can be obtained from a dart and an orbit: $\langle \alpha_1, \alpha_2, \alpha_3 \rangle$ for vertices, $\langle \alpha_0, \alpha_1, \alpha_3 \rangle$ for edges, $\langle \alpha_0, \alpha_1, \alpha_3 \rangle$ for faces and $\langle \alpha_0, \alpha_1, \alpha_2 \rangle$ for volumes.

Appendix A.2. Removal for n-Gmaps

In a general way for an n-dimensional space, the removal of an $i-$cell consists in removing this cell and in merging its two incidents $(i+1)-$cells: so removal can be defined for $0 \ldots (n-1)-$cells (vertices, edges and faces in 3D).

It is not always possible to remove a given *i*-cell C. Indeed, if there are more than two $(i+1)$-cells incident to C, C can not be removed because there is no rule to re-attach these cells after the removal. This condition is formalized in def 3 by the removable condition.

Definition 3 (Removable cell [65]). *An i-cell C in an n-Gmap $G = (D, \alpha_0, ..., \alpha_n)$ is* removable *if:*

- *$i = n - 1$;*
- *or $0 \leq i < n-1$, and, $\forall d \in C$, $\alpha_{i+1} \circ \alpha_{i+2}(d) = \alpha_{i+2} \circ \alpha_{i+1}(d)$.*

Def 4 gives the remove operation of a removable *i*-cell. Intuitively, this operation consists in modifying the α_i links of the darts which were previously *i*-sewn with the removed cell. Figure A.20 illustrates an example of a $0-$removal and a $1-$removal in a 2-Gmap.

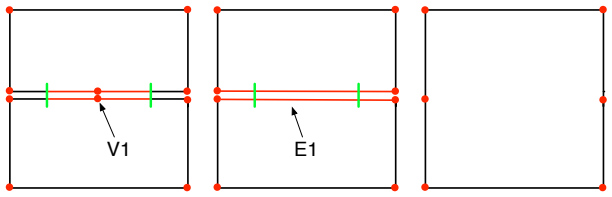


Fig. A.20. Example of removal operation: (a) a 2D objet; (b) the result of the removal of the vertex V_1; (c) the result of the removal of the edge E_1.

Definition 4 (Removal operation [65]). *Let* $G = (D, \alpha_0, ..., \alpha_n)$ *be an n-Gmap and C be a removable i-cell. Let* $D^s = \alpha_I(C)$ *C be the set of darts i-linked with C which do not be belong to C. the n-Gmap resulting for i-*removal *of C in G is* $G_{R_i}(C) = (D', \alpha'_0, ..., \alpha'_n)$ *defined by:*

- $D' = D/C$
- $\forall j \in \{0, \ldots, n\}, j \neq i, \alpha'_j = \alpha_{j/D'}$
- $\forall d \in D'\ D^S, \alpha'_i(d) = \alpha_i(d)$
- $\forall d \in D^S, \alpha'_i(d) = (\alpha_i \circ \alpha_{(i+1)})^k \circ \alpha_i(d)$,
  k being the smaller positive integer s.t. $(\alpha_i \circ \alpha_{(i+1)})^k \circ \alpha_i(d) \in D^S$