Active Subdivision Surfaces for the Semiautomatic Segmentation of Biomedical Volumes

Anaïs Badoual, Lucia Romani, and Michael Unser, *Fellow, IEEE*

Abstract—We present a new family of active surfaces for the semiautomatic segmentation of volumetric objects in 3D biomedical images. We represent our deformable model by a subdivision surface encoded by a small set of control points and generated through a geometric refinement process. The subdivision operator confers important properties to the surface such as smoothness, reproduction of desirable shapes and interpolation of the control points. We deform the subdivision surface through the minimization of suitable gradient-based and region-based energy terms that we have designed for that purpose. In addition, we provide an easy way to combine these energies with convolutional neural networks. Our active subdivision surface satisfies the property of multiresolution, which allows us to adopt a coarse-to-fine optimization strategy. This speeds up the computations and decreases its dependence on initialization compared to singleresolution active surfaces. Performance evaluations on both synthetic and real biomedical data show that our active subdivision surface is robust in the presence of noise and outperforms current state-of-the-art methods. In addition, we provide a software that gives full control over the active subdivision surface via an intuitive manipulation of the control points.

Index Terms—Multiresolution, triangular mesh, deformable model.

I. INTRODUCTION

OUR motivation is to develop a generic and versatile framework for the semiautomatic segmentation of volumetric structures in 3D biomedical data. We want the segmentation to be reasonably fast and robust, and the outcome to be easily editable by the user if needed.

Active surfaces allow for the extraction of volumetric structures [1]–[4]. They consist in flexible surfaces that are deformed from an initial user-provided configuration toward the boundary of a volume to be segmented. They became popular because they allow for user interaction, either to specify the initial configuration or to perform final manual adjustments, if needed. The deformation of the flexible surface is driven by the minimization of suitable energies [5]. Currently, 3D deformable models are described either implicitly (*e.g.*, by level sets [6], [7]), or explicitly, by meshes [2], [8] and parameterizations [3]. Parametric active surfaces have a continuously defined spatial representation through basis functions. It is easy to introduce smoothness and shape constraints [9], [10]. They require few parameters (*i.e.*, control points), which results in faster optimization and better robustness [11]. On one head, two well-known drawbacks of parametric approaches are (i) the restricted nature of the shapes that they can generate; and (ii) their inability to deal with topology changes such as surface merging and splitting. On the other hand, active meshes can handle topology changes. In addition, their discrete nature allows for an easy implementation; they are compatible with open-source libraries for optimization or visualization. However, they rely on a large number of parameters (*i.e.*, the vertices of the mesh), which requires an internal regularization term and makes the optimization more challenging. In this paper, we propose to represent the deformable model with a subdivision surface. This geometric representation combines the advantages of parametric and mesh-based approaches: the continuously defined surface is fully driven by an initial coarse mesh whose few control points are the parameters of the representation.

Subdivision is a powerful scheme to generate surfaces of arbitrary topology starting from an initial set of points (coarse mesh) [12]–[17]. A refinement process is recursively applied to the initial mesh to produce a continuously defined limit surface. The subdivision surface has the desirable property of being smooth while its shape is fully controlled by the initial set of points. Moreover, subdivision leads naturally to multiresolution, which allows for the surface of a shape to be represented at varying resolutions. Subdivision has become one of the basic geometric tools in computer graphics for representation and modeling [18]–[20]. The use of subdivision to construct segmentation models was pioneered in 2D by [21] for the Dyn-Levin-Gregory scheme [22] and by [23] for any convergent subdivision scheme. The extension to 3D models is more challenging. From a computational point of view, the geometry of the surface and the mesh connectivity increase the complexity of the implementation. Shapes are encoded with more control points, with three degrees of freedom for each one, which makes the optimization more complicated and slower. Moreover, it might be necessary to maintain evenly

spaced control points to favor a representative sampling of the surface. In the literature, only few works used subdivision to segment volumes. The authors of [24] presented the modeling of left ventricles using Doo-Sabin surfaces [12], while the segmentation of branch vessel structures was performed in [25] using Loop's subdivision scheme [13].

In this paper, we present the generic construction of active subdivision surfaces in the context of any subdivision scheme that operates on triangular meshes. The main contributions of this work are 1) a new 3D geometrical representation based on subdivision; 2) the derivation of region- and gradient-based energy functions that are guaranteed to have the proper limit proposed in [3]; 3) the presentation and integration of an algorithmic coarse-to-fine optimization strategy. We have implemented the method described in this paper as a user-friendly open-source plugin[1] available for the free open-source image-processing package Icy [26].

In Section II, we introduce the relevant theory related to subdivision surfaces. We particularize the construction by making use of Loop's subdivision scheme. In Section III, we describe the generic construction of active subdivision surfaces on triangular meshes. We also describe the proposed coarse-to-fine optimization strategy. In Section IV, we perform an extensive validation of active subdivision surfaces on both synthetic and real biological images. We also introduce a way to incorporate neural networks into our deformable model. Finally, conclusions are drawn in Section V.

II. SUBDIVISION SURFACES

A. Notation and Terminology

A closed triangular mesh $\mathcal{M}_{(k)}$, at resolution k, is defined by the set $\mathbb{P}_{(k)} = \left\{ \mathbf{p}_{(k)}[m] \in \mathbb{R}^3, \ m \in \{0, \ldots, N_k - 1\} \right\}$ of N_k points. These points are implicitly connected by triangles. The valence n of the vertex $\mathbf{p}_{(k)}[m]$ denotes the number of its adjacent vertices in the mesh $\mathcal{M}_{(k)}$. A vertex is extraordinary if its valence is different from six, otherwise it is regular.

B. Subdivision Schemes

Given the coarse mesh $\mathcal{M}_{(0)}$, we repeatedly apply subdivision rules k times to obtain a finer mesh $\mathcal{M}_{(k)}$, which does not necessarily contain the coarser mesh $\mathcal{M}_{(k-1)}$. The construction of $\mathcal{M}_{(k)}$ amounts to the specification of both its geometry (i.e., its vertices $\mathbb{P}_{(k)}$) and its topology (i.e., the connectivity between them). If $\mathcal{M}_{(k)}$ is suitably constructed for each k, then, when k tends to infinity, the sequence of denser and denser meshes $\{\mathcal{M}_{(k)}, \ k \geq 0\}$ converges to the continuously defined surface $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \sigma_3)$ with $\sigma_1, \sigma_2, \sigma_3 \in \mathcal{C}^0$. The subdivision rule from $(k-1)$ to k is defined by

$$\mathbb{P}_{(k)} = S_{k-1}\mathbb{P}_{(k-1)}, \tag{1}$$

where S_k is the subdivision operator at the kth iteration of the subdivision scheme. We say that a subdivision scheme is stationary if the subdivision rules in S_k are the same at each iteration; otherwise, it is non-stationary. The vertices of the

mesh at the kth iteration ($k \geq 1$) can be directly obtained from the initial set of control points $\mathbb{P}_{(0)}$ by

$$\mathbb{P}_{(k)} = S_{k-1}S_{k-2}\cdots S_0\mathbb{P}_{(0)}. \tag{2}$$

Formulation (2) makes it obvious that $\mathbb{P}_{(k)}$ depends exclusively on the N_0 vertices of $\mathbb{P}_{(0)}$, which we call control points. The N_k vertices of the mesh $\mathcal{M}_{(k)}$ at the kth iteration ($k \geq 1$) are called subdivision points.

C. Properties

In the literature, there is a wide range of subdivision schemes to produce surfaces. They differ in the properties that they confer to the limit surface.

1) Reproduction of Specific Shapes: The ability of a subdivision scheme to reproduce ellipsoids or spheres [17], [27]–[29].

2) Interpolation: The subdivision points obtained at each iteration lie on the limit surface [14], [30]–[33].

3) Smoothness: The subdivision rules given in [12] lead to $\mathcal{C}^1$ limit surfaces, while the ones developed in [13], [34] produce $\mathcal{C}^2$ continuous limit surfaces everywhere, except at extraordinary vertices where they are only $\mathcal{C}^1$ continuous. Non-stationary schemes generalizing the Doo-Sabin and Catmull-Clark schemes were proposed in [35]. Theorems to analyze the smoothness of a non-stationary scheme in regions with regular and extraordinary vertices can be found in [36] and [37], respectively.

4) Affine Invariance: The geometry of the limit surface changes in synchrony with any affine transformation that would be applied to the initial mesh. Conditions on the subdivision operators S_k to ensure this property are given in [28], [38].

The framework that we present in this paper is valid for any subdivision scheme for triangular meshes. However, for consistency and clarity purposes, we now specifically describe the subdivision scheme that we shall use in the experiments of Section IV.

D. Loop's Scheme

Loop's scheme is a widely used subdivision scheme for triangular meshes. It has been developed by Charles Loop in 1987 [13]. It generates a limit surface that is C^2-continuous everywhere, except at extraordinary vertices where the regularity is C^1. This scheme refines each triangle of the current mesh into four subtriangles (Figure 1 (a) and (b)). The splitting of each triangular face is achieved by creating the vertices of the finer mesh via the application of special refinement rules and by connecting them together. Such refinement rules are respectively termed *vertex-point* or *edge-point* rules, depending on whether the new vertex is located in correspondence to a vertex or an edge of the coarser mesh.

1) Vertex-Point Rule: The location of every former vertex $\mathbf{p}_{(k)}[m]$, $m \in \{0, \ldots, N_k - 1\}$, is updated. A new vertex $\mathbf{p}_{(k+1)}[q]$, $q \in \{0, \ldots, N_{k+1} - 1\}$, is obtained by the convex combination

$$\mathbf{p}_{(k+1)}[q] = \alpha \ \mathbf{p}_{(k)}[m] + \beta \sum_{\mathbf{p}_{(k)}[u] \in \mathcal{V}_m} \mathbf{p}_{(k)}[u], \tag{3}$$

---

[1]Demos and code of the plugin are available at bigwww.epfl.ch/demo/subdivision-surfaces/, as of May 2021.
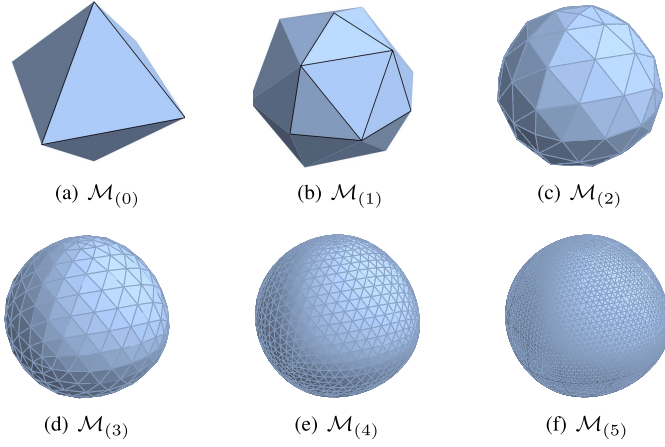
(a) $\mathcal{M}_{(0)}$ (b) $\mathcal{M}_{(1)}$ (c) $\mathcal{M}_{(2)}$

(d) $\mathcal{M}_{(3)}$ (e) $\mathcal{M}_{(4)}$ (f) $\mathcal{M}_{(5)}$

Fig. 1. Generation of an approximation of the surface of a sphere by Loop's scheme, starting from an octahedron.

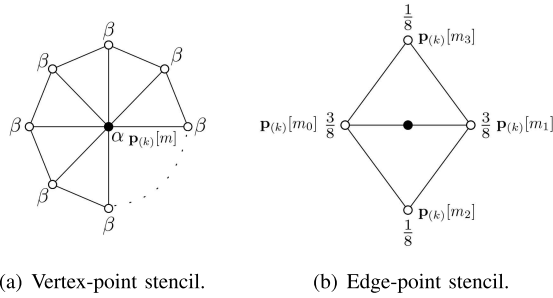(a) Vertex-point stencil.   (b) Edge-point stencil.

Fig. 2. Stencils for vertex-point (a) and edge-point (b) rules of Loop's scheme.

where $\mathcal{V}_m$ is the set of the n adjacent vertices of $\mathbf{p}_{(k)}[m]$ in $\mathcal{M}_{(k)}$, $\alpha = \frac{3}{8} + \left(\frac{3}{8} + \frac{1}{4}\cos(\frac{2\pi}{n}) \right)^2$, and $\beta = \frac{1-\alpha}{n}$. The rule is illustrated in Figure 2 (a).

2) Edge-Point Rule: For every edge in the previous mesh, a new vertex is inserted. Let us consider the two old adjacent triangles made of the vertices $\mathbf{p}_{(k)}[m_0]$, $\mathbf{p}_{(k)}[m_1]$, $\mathbf{p}_{(k)}[m_2]$, and $\mathbf{p}_{(k)}[m_3]$, with $m_0, m_1, m_2, m_3 \in \{0, \ldots, N_k - 1\}$, such that $\mathbf{p}_{(k)}[m_0]\mathbf{p}_{(k)}[m_1]$ is the common edge (Figure 2 (b)). The subdivision rule yields that

$$\mathbf{p}_{(k+1)}[q] = \frac{3}{8}\left(\mathbf{p}_{(k)}[m_0] + \mathbf{p}_{(k)}[m_1]\right) \\ + \frac{1}{8}\left(\mathbf{p}_{(k)}[m_2] + \mathbf{p}_{(k)}[m_3]\right), \quad (4)$$

where $q \in \{0, \ldots, N_{k+1} - 1\}$.

This scheme is stationary, easy to implement, and has the property to be affine invariant. In Figure 1, we illustrate the sphere approximated by Loop's scheme using an octahedron as initial mesh. Extensions of Loop's scheme to get additional properties like exponential polynomial reproduction and optimal shrinkage are considered in [17] and [39], respectively.

III. ACTIVE SUBDIVISION SURFACES

An active surface is composed of a geometric representation of the surface and an energy term. This energy measures the deformation and attraction of the surface to the boundary of

an object of interest in an image. In this section, we describe the construction of an active subdivision surface on triangular meshes and propose an integrated coarse-to-fine optimization strategy to minimize the energy term. The flowchart of the proposed framework is depicted in Figure 3. From now on, we consider orientable closed surfaces (*i.e.*, compact and without boundary) since we want our active surface to segment blob-like objects within 3D images.

A. Geometric Representation

We implicitly represent the surface of the deformable model by the continuously defined, orientable, closed, limit surface σ obtained by means of the selected convergent subdivision scheme, which is informally denoted by

$$\sigma = \lim_{k \to \infty} \mathcal{M}_{(k)}, \quad (5)$$

where $\mathcal{M}_{(k)}$ is the triangular mesh at resolution k defined by connecting the vertices in (2). This implies that the properties of the model depend on the choice of the subdivision scheme. A mandatory requirement is affine invariance; it ensures that the surface is described independently from its location and orientation. Moreover, the quality of the segmentation can be influenced as follows: first, the regularity of the surface defines the smoothness of the segmentation result; second, the geometric reproduction properties have to match the shape of interest. The number N_0 of control points determines the number of degrees of freedom of the model. A small N_0 leads to simple and constrained shapes, while an increase in N_0 brings additional flexibility to approximate arbitrary surfaces.

B. Energy Terms

The second important component is the energy functional that influences the quality of the segmentation. Traditionally, the energy is divided into an internal energy, which drives the smoothness of the surface, and an external energy, which is purely data-driven. In the present case, the smoothness of the surface is driven by the choice of the subdivision scheme that produces at least $\mathcal{C}^1$ surfaces. Therefore, we reduce the energy E of the active subdivision surface to the external term

$$E(f, \mathbb{P}_{(k)}) = b E_{\text{grad}}(f, \mathbb{P}_{(k)}) + (1 - b) E_{\text{region}}(f, \mathbb{P}_{(k)}), \quad (6)$$

where E_{grad} is a surface-based term that uses gradient information to detect boundaries, and E_{region} is a region-based term that uses statistical information to establish a homogeneity-based partition of an image. The contribution of those two energies is balanced by $b \in [0, 1]$, whose value depends on the application. The volumetric image f is the image to be segmented or a filtered version of it, while $\mathbb{P}_{(k)} = \mathbb{P}_{(k)}(\mathbb{P}_{(0)})$ describes the surface. The optimization is performed in terms of the control points $\mathbb{P}_{(0)}$, with

$$\mathbb{P}_{(0)\text{opt}} = \underset{\mathbb{P}_{(0)}}{\arg \min} \; E\left(f, \mathbb{P}_{(k)}(\mathbb{P}_{(0)})\right). \quad (7)$$

The gradient energy that we propose is expressed as

$$E_{\text{grad}}(f, \mathbb{P}_{(k)}) = -\frac{1}{2^{2k}} \sum_{m=0}^{N_k - 1} \nabla f(\mathbf{p}_{(k)}[m]) \cdot \mathbf{n}(\mathbf{p}_{(k)}[m]), \quad (8)$$
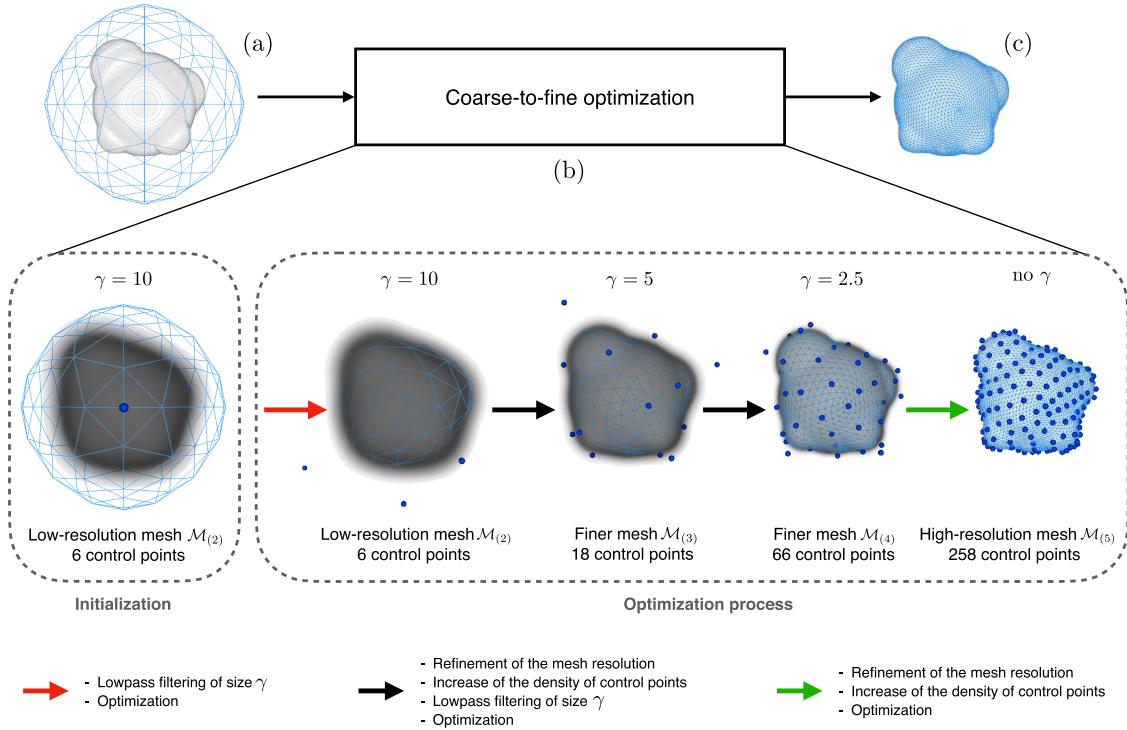
Fig. 3. Flowchart of the proposed active subdivision surface. Blue dots: control points; Blue mesh: subdivision surface. (a) Segmentation inputs: the volume f to segment and a subdivision surface represented by the triangular low-resolution mesh $\mathcal{M}_{(k)}$ and encoded by N_0 control points (Section II). For the illustration, we choose $k = 2$ and $N_0 = 6$. (b) Coarse-to-fine optimization (Section III-C): first, we apply a lowpass filter to f to obtain a volume where the object of interest contains few details only. We fit the coarse mesh $\mathcal{M}_{(2)}$ on this smoothed volume through the minimization of an energy functional that consists of a combination of gradient-based and region-based terms (Section III-B). Then, we use the outcome of this first optimization as the initialization at the next resolution level. We refine the mesh and the number of control points, optimizing it on a smoothed version of f where the object of interest has more details than at the previous step. We continue until we reach the finest resolution level that corresponds to the original volume f. (c) Segmentation outcome.

where $\nabla f(\mathbf{p}_{(k)}[m])$ and $\mathbf{n}(\mathbf{p}_{(k)}[m])$ are the gradient of f and the normal vector, respectively, at the vertex $\mathbf{p}_{(k)}[m]$. The expression of $\mathbf{n}(\mathbf{p}_{(k)}[m])$ is given by

$$\mathbf{n}(\mathbf{p}_{(k)}[m]) = \frac{\sum_{T \in \mathcal{O}_m} \mathbf{n}_t(T)}{||\sum_{T \in \mathcal{O}_m} \mathbf{n}_t(T)||}, \qquad (9)$$

where $\mathcal{O}_m$ is the set of all the triangles to which $\mathbf{p}_{(k)}[m]$ belongs and $\mathbf{n}_t(T)$ is the normal of the triangle T.

Proposition 1: As $k \to \infty$, the energy given by (8) *converges to the standard energy given in [3] as*

$$E_{\mathrm{grad}}(f, \mathbb{P}_{(k)}) \xrightarrow[k \to \infty]{} - \oiint_{\mathcal{S}} \nabla f(\boldsymbol{\sigma}) \cdot \mathrm{d}\boldsymbol{\sigma}, \qquad (10)$$

where $\mathcal{S}$ is the surface described by $\boldsymbol{\sigma}$ and $\mathrm{d}\boldsymbol{\sigma}$ represents the vector differential area.

The proof of Proposition 1 is given in Appendix B. For the region energy, we adopt the strategy proposed in [3] and [40]. We build an enclosing surface $\boldsymbol{\sigma}_\lambda$ around $\boldsymbol{\sigma}$. The volumes enclosed by $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}_\lambda$, denoted by $\mathcal{V}$ and $\mathcal{V}_\lambda$, respectively, are such that $\mathcal{V} \subset \mathcal{V}_\lambda$, while $\mathcal{V}$ and $\mathcal{V}_\lambda \setminus \mathcal{V}$ have the same volume. Then, E_{region} discriminates an object from its background by maximizing the signed contrast between the intensity of the image f averaged within $\mathcal{V}$, and the intensity of f averaged over the volume of the shell $\mathcal{V}_\lambda \setminus \mathcal{V}$. For the

expression of E_{region}, we choose

$$E_{\mathrm{region}}(f, \mathbb{P}_{(k)}) = \frac{1}{\left| \mathrm{V}(\mathbb{P}_{(k)}) \right| 2^{2k}}$$

$$\times \left(2 \sum_{m=0}^{N_k - 1} F(\mathbf{p}_{(k)}[m]) n_1(\mathbf{p}_{(k)}[m]) \right.$$

$$\left. - \sum_{m=0}^{N_k - 1} F(\mathbf{p}_{\lambda(k)}[m]) n_1(\mathbf{p}_{\lambda(k)}[m]) \right), \quad (11)$$

where $\left| \mathrm{V}(\mathbb{P}_{(k)}) \right|$ is the volume of $\mathcal{M}_{(k)}$, n_1 is the first coordinate of the normal vector given by (9), and $\mathbb{P}_{\lambda(k)}$ is the set of subdivision points that defines the surface $\boldsymbol{\sigma}_\lambda$. The volume F is the pre-integrated volumetric image along the first dimension defined by $F(p_1, p_2, p_3) = \int_{-\infty}^{p_1} f(\tau, p_2, p_3) \mathrm{d}\tau$. Note that the volume F is precomputed and stored in a lookup table, which speeds up the computation of the algorithm.

Proposition 2: As $k \to \infty$, the proposed energy (11) *converges to the standard energy given in [3] as*

$$E_{\mathrm{region}}(f, \mathbb{P}_{(k)}) \xrightarrow[k \to \infty]{} \frac{1}{|\mathrm{V}|} \left(\iiint_{\mathcal{V}} f \mathrm{dV} - \iiint_{\mathcal{V}_\lambda \setminus \mathcal{V}} f \mathrm{dV} \right),$$

$$(12)$$

where V is the volume of $\mathcal{V}$.

The proof of Proposition 2 is given in Appendix C.

C. A Coarse-to-Fine Optimization Strategy

The advantage of a multiresolution strategy has been shown in [23], for 2D subdivision models, to accelerate the segmentation and to make it more robust to the initialization. The use of such algorithms is even more relevant in 3D as the optimization is more difficult. There, we propose an integrated coarse-to-fine-optimization strategy that combines the refinement of 1) the mesh resolution, to make the computation faster and less sensitive to the initialization; 2) the number of control points, to maintain a favorable sampling of the mesh throughout the process.

1) Coarse-to-Fine Mesh Resolution: The energy terms (8) and (11) depend on the subdivision points $\mathbb{P}_{(k)}$, which is the source of the main computational bottleneck. The accuracy of the energy and, therefore, of the segmentation, increases with the resolution of the mesh. However, the choice of a large k also considerably slows down the computation. Moreover, active surfaces tend to be sensitive to the initialization, especially when using surface-based energy. To address those issues, we optimize the active surface in a coarse-to-fine fashion that is inherent to the iterative process of subdivision. Our algorithm exploits the following properties: 1) a smoothed volume contains fewer details and less noise than the original one; 2) the resolution of the mesh (*i.e.*, the number of subdivision points) can be adapted to the resolution of the object to be segmented (*i.e.*, the level of details in the volume).

Algorithm: We apply K successive lowpass filters G_k to the original volume f to obtain K smoothed volumes f_k, with the width of G_k being higher than that of G_{k+1}. The active subdivision surface is first optimized on the coarsest volume f_1, where the object of interest only contains few details. The initial mesh on f_1 can be coarse as well since the shape of the underlying object tends to get simplified. The optimization on f_1 is fast and the outcome is then used as initialization at the next resolution level on f_2. We refine the mesh and optimize it on f_2. The process continues until the optimization reaches the finest resolution level that corresponds to the original volume f.

2) Coarse-to-Fine Density of the Control Points: The segmentation of intricate shapes requires a large number of control points in order to catch all the details. However, the segmentation becomes less robust when N_0 increases since the optimization iterative process is more likely to be stuck in local minima. Moreover, the deformation of the surface can lead to an undesirable distribution of the control points along the surface, which results in an unfavorable sampling of the mesh. We want to avoid it during the optimization. Popular methods for keeping the vertices well distributed (*e.g.*, Laplacian smoothing) have a high computational cost [8], [41]. Instead, we propose the following strategy: We start with few control points and then progressively increase N_0. This is made possible by the fact that the mesh $\mathcal{M}_{(k)}$ is entirely defined by any coarser mesh $\mathcal{M}_{(q)}$, $0 \leq q \leq (k-1)$. In this way, our initial segmentation is coarse at first, with a poor flexibility of the active subdivision surface. The upside is a good distribution of the control points and, thus, of the subdivision points over the surface. We then progressively

---

Algorithm 1 Coarse-to-Fine Optimization Strategy

Input: original volume f, initial control points $\mathbb{P}_{(0)}$, level k_f for the final control points $\mathbb{P}_{(k_f)}$

Initialization: $\mathbb{P}_{(1)} = S_0 \mathbb{P}_{(0)}$ (low-resolution mesh)
　　　　　　　$\mathbb{C} = \mathbb{P}_{(0)}$ 　　　(set containing the control points)
　　　　　　　$k_0 = 0$ 　　　　(current level of the control points)

For K iterations over $k \geq k_0 + 1$:
　　compute: $f_k = f * G_k$
　　optimize: $\mathbb{C}_{\text{opt},k} = \underset{\mathbb{C}_{\text{opt},k-1}}{\arg\min} E(f_k, \mathbb{P}_{(k)})$
　　　　　　$\mathbb{P}_{(k)_{\text{opt}}} = S_{k-1} S_{k-2} \cdots S_{k_0} \mathbb{C}_{\text{opt},k}$
　　refine the resolution of the mesh: $\mathbb{P}_{(k+1)} = S_k \mathbb{P}_{(k)_{\text{opt}}}$
　　refine the density of the control points:
　　　　If $(k_0 < k_f)$ $\mathbb{C}_{\text{opt},k} = S_{k_0} \mathbb{C}_{\text{opt},k}$ and $k_0 \leftarrow k_0 + 1$

Until: high-resolution segmentation on the original volume f

---

refine local details by increasing N_0. The final number of control points depends on the intricacy of the shape to be segmented.

The pseudocode in Algorithm 1 describes the entire coarse-to-fine optimization strategy. It combines the refinement of the resolution of the mesh with that of the density of the control points. In this code, the set $\mathbb{C}$ contains the control points. Note that they change after each round of optimization. We thus denote by $\mathbb{C}_{\text{opt},k}$ the set of the optimized control points at iteration k. The volumes f_k and their pre-integrated versions are precomputed, which accelerates the segmentation process.

IV. EXPERIMENTS AND VALIDATION

We proceed in four steps to evaluate the performance of the proposed active subdivision surface. We first test its robustness to noise and, in a second step, its sensitivity to the initialization. Then, we investigate its accuracy in the context of the segmentation of an intricate shape, when a lot of flexibility is required from the active surface. Finally, we illustrate applications on real biomedical data where the ground truth is not available.

For each experiment, we use Loop's scheme to represent our active subdivision surface and we carry out its optimization by a Powell-like line-search method [42]. The experiments are performed on a 1.7 GHZ processor with 8 GB RAM.
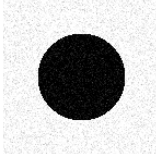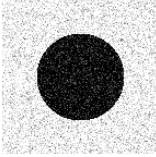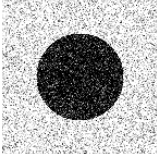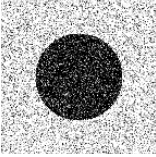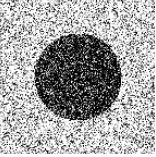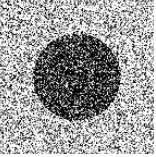
We use the Jaccard index J to measure the overlap between a segmentation result $\mathcal{V}$ and the corresponding ground truth $\mathcal{V}_{\text{GT}}$. It is defined as

$$0 \leq J = \frac{|\mathcal{V} \cap \mathcal{V}_{\text{GT}}|}{|\mathcal{V} \cup \mathcal{V}_{\text{GT}}|} \leq 1. \tag{13}$$

A. Robustness to Noise

We investigate the robustness to noise of the active subdivision surface as a function of the number N_0 of control points. The test images consist in a binary sphere of radius 40 voxel units on a 3D array of size ($256 \times 256 \times 256$) voxels. We corrupted the test image by six levels of additive

TABLE I

JACCARD INDICES FOR THE SEGMENTATION OF NOISY 3D DATA

N_0	SNR $= \infty$	SNR $= 12.75$	SNR $= 5.1$	SNR $= 3.18$	SNR $= 2.55$	SNR $= 1.7$	SNR $= 1.275$
6	0.96	$0.96 \pm 7 \times 10^{-4}$	$0.95 \pm 2 \times 10^{-3}$	0.94 ± 0.06	0.94 ± 0.12	0.79 ± 0.16	0.59 ± 0.14
18	0.98	0.98 ± 10^{-3}	0.97 ± 0.11	0.95 ± 0.06	0.94 ± 0.089	0.80 ± 0.16	0.60 ± 0.23
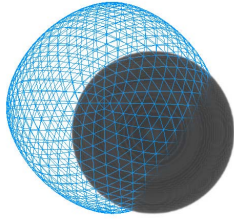
Fig. 4. Initialization on the volumetric image, $J = 0.39$.

TABLE II

PARAMETERS OF THE COARSE-TO-FINE OPTIMIZATION STRATEGY (SECTION IV-B)

Iteration	Control Points		Mesh		Filter Size γ
1	$\mathbb{P}_{(0)}$	6 pts	$\mathcal{M}_{(2)}$	66 pts	7
2	$\mathbb{P}_{(1)}$	18 pts	$\mathcal{M}_{(3)}$	258 pts	2
3	$\mathbb{P}_{(1)}$	18 pts	$\mathcal{M}_{(4)}$	1026 pts	none

white Gaussian noise (50 realizations per level of noise). Slices of the resulting 3D volumes are illustrated in Table I. We initialized the active subdivision surface with the roughly spherical surface described by the high-resolution mesh $\mathcal{M}_{(4)}$ given in Figure 1 (e). Its overlap with the ground truth corresponds to $J = 0.39$ (Figure 4). We ran the optimization process for 3,500 iterations using only the region-based energy and the strategy described in Section III-C. The signal-to-noise ratio (SNR) corresponding to a given noise level and the median Jaccard index were computed. The SNR that we use is the ratio of the mean value of the signal and the standard deviation of the noise. The results are summarized in Table I, where N_0 is the final number of control points. We observe that the active subdivision surface is robust to noise since it is able to segment satisfactorily, even for low SNRs. This can be explained by the fact that the region energy (11) estimates the mean intensity over regions, while the Gaussian noise has zero mean.

B. Robustness to the Initialization

To study the sensitivity of the active subdivision surface to its initialization, we compared our model in terms of accuracy and speed against other segmentation methods such as the active parametric surface described in [3] and the 3D active mesh of [2]. Note that these two methods are also robust to noise. As far as we know, there are no other available software for 3D data where the segmentation is editable. The implementation of the two methods was taken from the package Icy [26].

The test image is the binary sphere of Section IV-A. For each method, the initialization is (essentially) a sphere of radius r voxel units centered in the image. The goal is to segment the binary sphere from several initializations by varying the value of r.
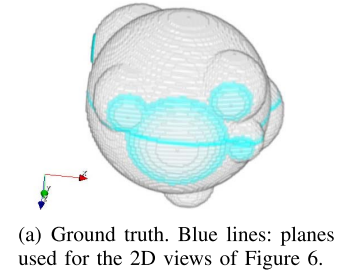
We initialized our active subdivision surface with the low-resolution mesh $\mathcal{M}_{(2)}$ illustrated in Figure 1 (c), the control points being the 6 vertices that make up the octahedron (Figure 1 (a)). We optimized the model using the surface-based energy and the coarse-to-fine strategy with the parameters given in Table II. In this table, γ denotes the standard deviation of a Gaussian filter. For the active mesh, we set the mesh resolution to 12, the time-evolution step to 0.1, the window size to 100, the contour smoothness to 0.04, and we evolved the mesh using the gradient energy with weight (-0.1). For the active parametric surface, we set the number of control points to 12 as it favors ellipsoid-like shapes during the segmentation process. We deformed the parametric surface using a surface-based energy. For this method only, we ran the optimization on both the original image and on a smoothed version of the image filtered with a Gaussian kernel with $\gamma = 7$. We computed the Jaccard index of the segmentation outcome over all initializations and methods. The results as well as the segmentation time (without preprocessing) are given in Table III. In this table, "failed" means that the active mesh did not detect the sphere and vanished, or that the parametric surface self-intersected. Active subdivision surfaces and active parametric surfaces have a similar performance in terms of accuracy and speed as soon as the initialization is close enough to the object to segment. Otherwise, the basin of attraction of the active parametric surface is too narrow for the surface-based energy. We observe that the standard deviation $\gamma = 7$ is large enough to attract the active parametric surface for almost every initial configuration. However, in each case the segmentation is less accurate as the boundary of the sphere to segment is smoothed. The active mesh performs well, provided that the initialization includes the sphere to segment.

TABLE III
ACCURACY AND EFFICIENCY WITH RESPECT TO THE INITIALIZATION

Method Radius	Active Subdivision Surface		Active Parametric Surface		Active Parametric Surface, $\gamma = 7$		Active Mesh	
	J	Time [s]	J	Time [s]	J	Time [s]	J	Time [s]
20	**0.99**	5.74	0.98	17.56	0.91	5.40	failed	-
25	**0.99**	5.71	0.16	5.87	0.91	7.27	failed	-
30	**0.99**	5.73	0.33	6.36	0.91	5.35	failed	-
35	**0.99**	5.72	0.99	6.15	0.91	5.32	failed	-
45	**0.99**	5.82	0.99	6.01	0.91	5.40	0.96	7.61
50	**0.99**	5.85	0.49	5.84	0.91	5.73	0.95	12.38
55	**0.99**	5.81	0.37	5.63	0.91	5.83	0.95	18.66
60	**0.99**	5.83	0.30	5.82	0.91	5.42	0.95	32.58
65	**0.99**	5.84	0.23	6.10	0.91	5.81	0.95	41.92
70	**0.99**	5.80	0.17	6.40	failed	-	0.96	60.70

TABLE IV
PARAMETERS OF THE COARSE-TO-FINE OPTIMIZATION
STRATEGY (SECTION IV-C)

Iteration	Control Points		Mesh		Filter Size γ
1	$\mathbb{P}_{(0)}$	6 pts	$\mathcal{M}_{(2)}$	66 pts	10
2	$\mathbb{P}_{(1)}$	18 pts	$\mathcal{M}_{(3)}$	258 pts	5
3	$\mathbb{P}_{(2)}$	66 pts	$\mathcal{M}_{(4)}$	1026 pts	2.5
4	$\mathbb{P}_{(3)}$	258 pts	$\mathcal{M}_{(5)}$	4098 pts	none

Otherwise, it systematically fails. The active subdivision surface led to accurate segmentation even for initializations far from the object to segment. This result is explained by the use of the coarse-to-fine optimization strategy since the model is initialized on the coarsest image with reduced details and a large basin of attraction. In addition, the proposed method is also the fastest.

C. Segmentation Accuracy

In this section, we generated a synthetic blebbing cell [43] located at the center of a volumetric image of size ($256 \times 256 \times 256$) voxels (Figure 5 (a)). The presence of blebs on the surface of the cell makes the segmentation challenging and requires a lot of flexibility from the segmentation model. We then compared our proposed method in terms of accuracy to the two segmentation methods mentioned in Section IV-B.

For each method, the initialization is (essentially) a centered sphere of radius 50 voxel units. Its overlap with the blebbing cell corresponds to $J = 0.56$. We initialized the active subdivision surface with the low-resolution mesh $\mathcal{M}_{(2)}$ illustrated in Figure 1 (c), encoded by the 6 control points that form the octahedron of Figure 1 (a). For the energy (6), we set the tradeoff parameter to $b = 0.8$. We evolved the active subdivision surface using the coarse-to-fine optimization strategy with the parameters given in Table IV. For the active mesh, we set the mesh resolution to 5, the time-evolution step to 0.1, the window size to 100, the contour smoothness to 0.05, and we optimized the mesh using the region energy with weight 1. For the active parametric surface, we also used a mix of surface-based and region-based energies with $b = 0.8$.

For this last segmentation method, we repeated the experiment for different numbers of control points. They are listed in Table V, where we also show a comparison of the final

(a) Ground truth. Blue lines: planes used for the 2D views of Figure 6.

(b) Active subdivision surface.

(c) Active mesh.

(d) Active parametric surface (12 control points).

(e) Active parametric surface (36 control points).

(f) Active parametric surface (62 control points).

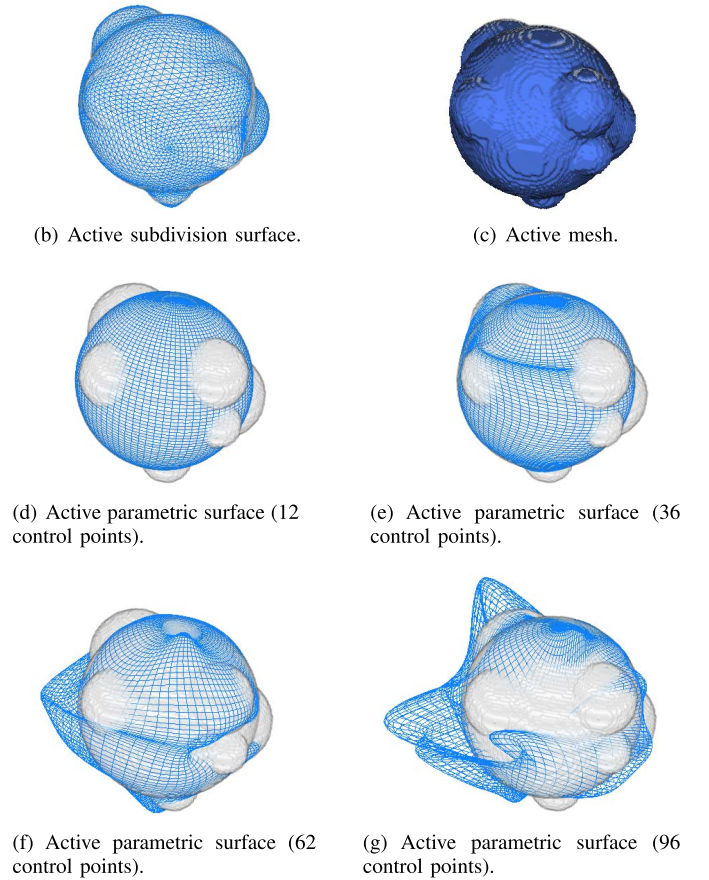(g) Active parametric surface (96 control points).

Fig. 5. 3D views of the segmentation outcomes.

Jaccard index.[2] 3D views as well as 2D orthogonal views of the segmentation outcomes are illustrated in Figures 5 and 6, respectively. We observe that both our proposed method and

[2]The ideal segmentation of just the main sphere of the blebbing cell would correspond to $J = 0.88$.

TABLE V
Jaccard Indices for the Segmentation of a Blebbing Cell

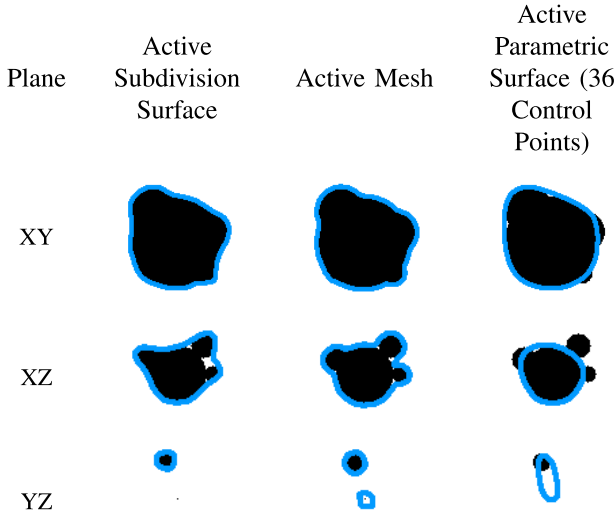| Method	J
Active Subdivision Surface	**0.98**
Active Mesh	0.96
Active Parametric Surface (12 Control Points)	0.87
Active Parametric Surface (36 Control Points)	0.93
Active Parametric Surface (62 Control Points)	0.87
Active Parametric Surface (96 Control Points)	0.69

Fig. 7.   UNet architecture.

Plane	Active Subdivision Surface	Active Mesh	Active Parametric Surface (36 Control Points)
XY			
XZ			
YZ			

Fig. 6.   2D orthogonal views of the segmentation outcomes.

(a) Synthetic volume.   (b) Probability map.

Fig. 8.   Probability map of 3D HL60 cell nuclei in low SNR.

the active mesh accurately segment the blebbing cell and give smooth meshes (Figure 5 (b) and (c)). The active mesh tends to extend outside of the boundary of the object compared to the active subdivision surface (Figure 6, YZ plane). However, it segments better two close blebs (Figure 6, XZ plane). Clearly, the active parametric surface has the worst performance. The segmentation with 12 control points leads to a very smooth surface. However, only the main sphere of the blebbing sphere is segmented, as the flexibility afforded by 12 control points is low. With the additional flexibility afforded by 36 control points it is able to segment some blebs. However, irregularities in the surface start to appear. When we further increase the flexibility of the active parametric surface, the distribution of the control points misbehaves; moreover, irregularities and twists of the surface grow too large.

D. Combination With Convolutional Neural Networks

Structures cannot always be fully characterized from their internal distribution of pixel values. Our energy terms (8) and (11), based on intensity information only, thus have their limitations when applied on the original image. Sometimes, it is best to exploit the power of deep learning to automatically learn and detect the features of the object of interest [44], [45]. In this context, we present an easy way to combine our method with convolutional neural networks (CNN) such as UNet [46]. UNet is the workhorse of deep segmentation methods and achieves state-of-the-art results in many image-segmentation
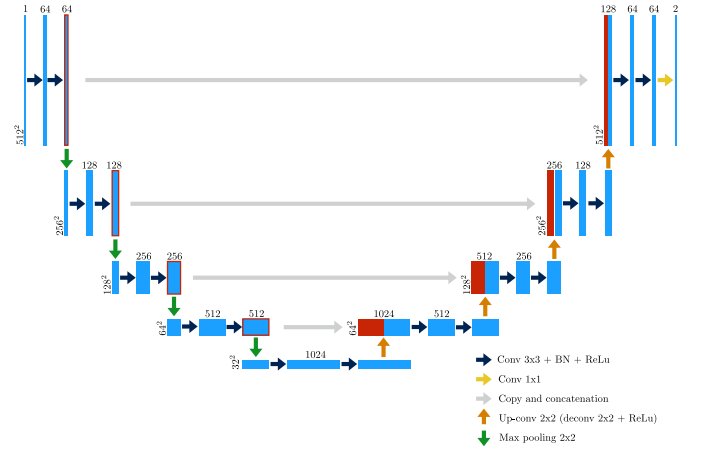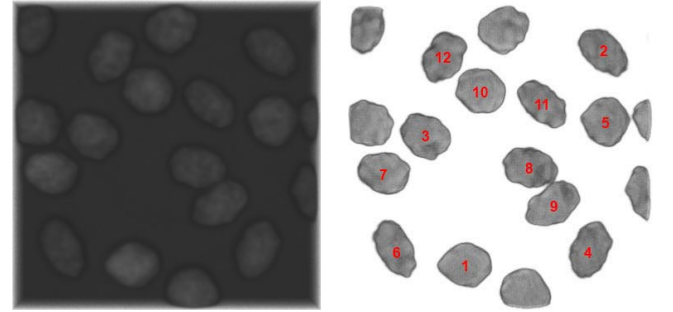
applications. Let f in (6) be the probability map of a pre-trained UNet. The idea then is to optimize our active subdivision surface on f using our energy terms (8) and (11).

The image set BBBC024vl [47] from the Broad Bioimage Benchmark Collection[3] contains 60 labeled volumes $(807 \times 565 \times 129)$ of 3D HL60 cell nuclei in low SNR. We trained a 2D UNet on $1,548$ 2D images corresponding to the slices of 12 volumes of this database (6 for the training set and 6 for the test set). The UNet architecture is given in Figure 7. We applied the trained UNet on each slice of a new volume of the BBBC024vl data set (see Figure 8 (a)) and we combined the obtained 2D probability maps to construct the final volume f. The resulting 3D probability map f is given in Figure 8 (b). We then applied our active subdivision surface on f to segment twelve nuclei. They are numbered in Figure 8 (b). We used a mix of surface-based and region-based energies with $b = 0.5$. We compared this method in terms of accuracy against our active subdivision surface optimized on the original volume and UNet with 0.5 as threshold value to decide whether to classify a pixel as 0 or 1. The results are given in Table VI and illustrated in Figure 9. We observe that, for this application, the incorporation of deep learning into our deformable model slightly improves the results. The good

[3]The image set was taken from https://data.broadinstitute.org/, as of May 2019.

(a) Nuclei No 6 and No 1.

(b) Nuclei No 8 and No 9.
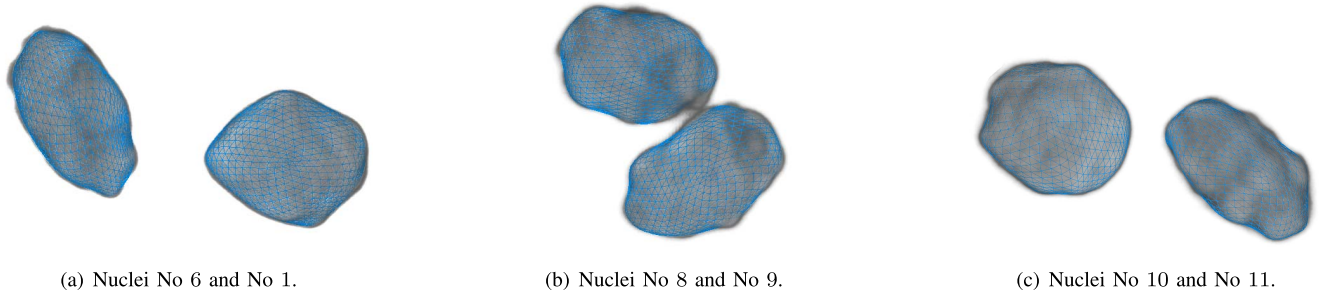
(c) Nuclei No 10 and No 11.

Fig. 9. Segmentation of 3D HL60 cell nuclei using the active subdivision surface on the probability map obtained with UNet. For numbering, refer to Figure 8.

TABLE VI
JACCARD INDICES FOR THE SEGMENTATION OF 3D HL60 CELL NUCLEI IN LOW SNR

Method Nucleus	Active Subdivision Surface on the Original Volume	Active Subdivision Surface on the 3D Probability Map	UNet
$n^o 1$	0.89	0.91	**0.92**
$n^o 2$	**0.89**	0.87	0.75
$n^o 3$	0.87	**0.88**	0.86
$n^o 4$	**0.89**	0.88	0.80
$n^o 5$	0.88	**0.92**	0.87
$n^o 6$	0.86	**0.88**	0.77
$n^o 7$	0.89	**0.92**	0.90
$n^o 8$	**0.88**	0.87	0.79
$n^o 9$	0.87	**0.91**	0.87
$n^o 10$	0.89	0.89	**0.90**
$n^o 11$	0.86	**0.88**	0.77
$n^o 12$	**0.87**	0.87	0.76
Average	0.88 ± 0.01	$\mathbf{0.89 \pm 0.02}$	0.83 ± 0.06

performance of the active subdivision surface optimized on the original volume can be explained by two reasons. First, our method is robust to noise (see Section IV-A). Second, although the internal intensity distribution of nuclei is heterogeneous, nuclei are well contrasted with the background image; this renders our region-based energy (11) suitable for this application. By additionally taking nuclei texture into account through the use of the probability map f, we further improve the segmentation accuracy.

Compared to UNet alone, the combined method has two main advantages: it provides a continuous limit surface of the structure of interest, and leads to better segmentation results, at least for this application. The weaker performance of UNet is mainly due to the threshold that is applied to the probability map to obtain the final binary segmentation. This threshold introduces holes in the nuclei and irregularities at the boundaries (see Figure 10 (a_2) and (b_2)). The continuity and smoothness of our active subdivision surface prevent these issues (see Figure 10 (a_3) and (b_3)). In addition, while the thresholding decision process uses only local features, our method infers the boundaries of the regions of highest probability by taking into account both local and global information (through the energies (8) and (11), respectively). This prevents our surface to be attracted by outlier voxels with false high probabilities or to integrate in the segmentation outcome voxels with false low probabilities (see Figure 10 (a_1) and (b_1)).
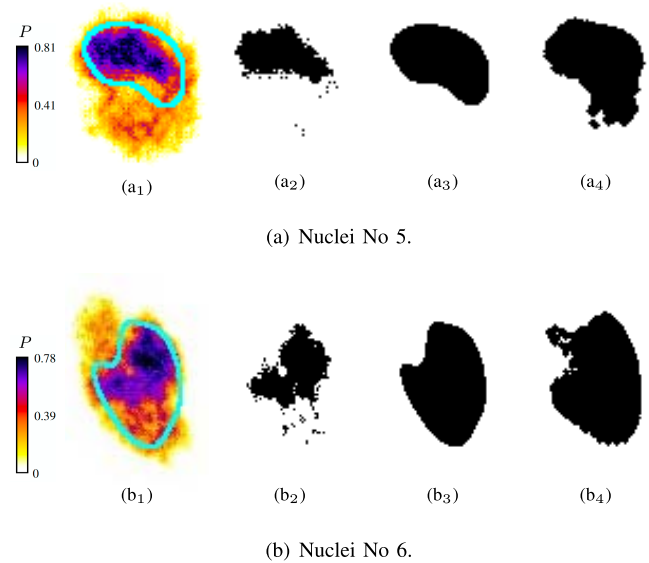

(a) Nuclei No 5.

(b) Nuclei No 6.

Fig. 10. 2D XY orthogonal views of two nuclei segmentation outcomes obtained with the UNet ((a_2) and (b_2)) and with our active subdivision surface optimized on the 3D probability map f ((a_3) and (b_3)). Slices of the corresponding volumes f are displayed in (a_1) and (b_1), where we indicate the segmentation outcomes obtained with our method (cyan contour). The ground-truths are given in (a_4) and (b_4).

In a second experiment, we investigated the segmentation of brain tumors on MRI images. Tumors have low contrast with the surrounding healthy tissue and a high spatial

TABLE VII

JACCARD INDICES FOR THE SEGMENTATION OF 3D BRAIN
TUMORS ON MRI IMAGES

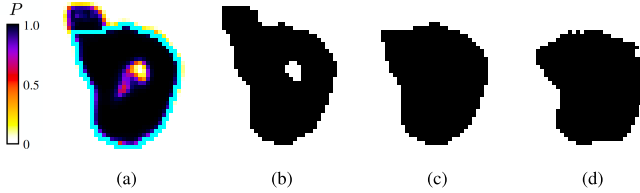| Method	J
Active Subdivision Surface on the Original Volume	0.38 ± 0.19
Active Subdivision Surface on the 3D Probability Map	0.72 ± 0.15
UNet	$\mathbf{0.75 \pm 0.13}$

Fig. 11. 2D XY orthogonal views of the 3D probability map f obtained with the UNet (a), the segmentation outcome obtained with UNet for a decision threshold of 0.5 (b), the one obtained with our active subdivision surface optimized on f (c), and the ground-truth (d).

heterogeneity. Neural networks are widely used in medical and cancer applications thanks to their ability to learn the optimal discriminative features [48], [49]. We trained a 3D UNet on the Brain Tumors Segmentation (BraTS) dataset from the publicly-available Medical Segmentation Decathlon Challenge[4] [50], [51]. This dataset contains 484 multimodal (FLAIR, T1w, T1gd, T2w) MRI volumes ($240 \times 240 \times 155$) depicting high- or low-grade gliomas. For this experiment, we used only the FLAIR channel and we focused on the segmentation of the whole tumor. We used 411 images for the training set, 36 images for the validation set, and 37 for the test set. For each test image, we optimized our active subdivision surface on the 3D probability map f provided by the 3D trained UNet by using a mix of our intensity-based energies (8) and (11). The average Jaccard index over the 37 test images is given in Table VII. We also reported the average Jaccard index obtained with UNet for a decision threshold of 0.5, and the one obtained with our active subdivision surface optimized on the original volumes.

As expected, our proposed active subdivision surface fails to extract gliomas from the original volumes. However, the incorporation of deep learning into our method considerably improves its performance, as it almost reaches the accuracy level obtained with UNet. The aforementioned benefits of finding the regions of highest probability using the active subdivision surface over the threshold are still observed in this application (see Figure 11). However, some tumors are fragmented in the test set (see Figure 12); this does not adapt well to our method, which was developed for the segmentation of a single object in the volume. This limitation of our method is discussed in Section IV-F1. It explains the reduced performance of our active subdivision surface combined to deep learning compared to UNet. As user-interaction is one
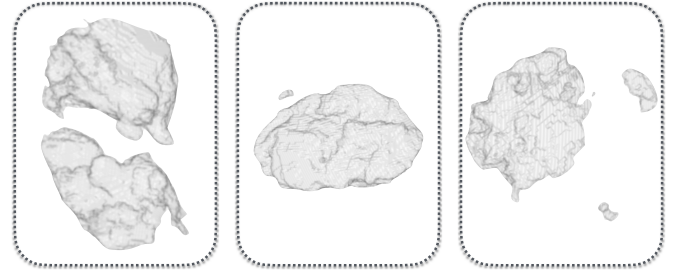
[4]The BraTS dataset was taken from http://medicaldecathlon.com/, as of January 2021.

Fig. 12. Illustration of three fragmented brain tumors from 3D MRI images.

Ground-truth.   Outcome.   Manual edit.
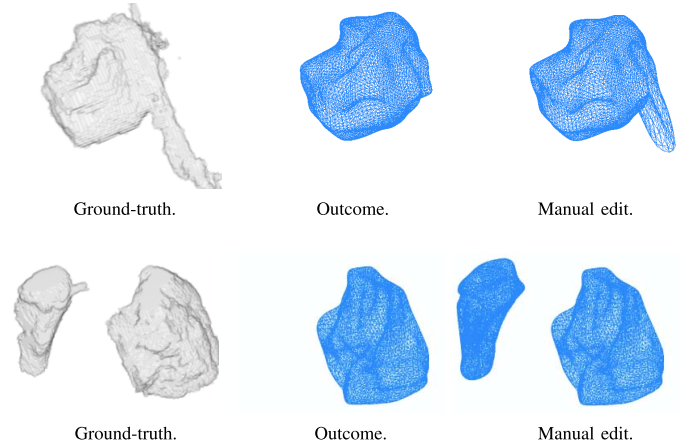
Ground-truth.   Outcome.   Manual edit.

Fig. 13. Illustrations of the user-interactivity of our method. We manually edit the segmentation outcome of brain tumors obtained with our active subdivision surface that was optimized on the probability map produced by a UNet.

of the main assets of our framework, we used it either to manually initialize additional subdivision surfaces in volumes containing fragmented tumors, or to refine segmentation outcome locally by manually moving one or several control points (see Figure 13). Note that the Jaccard indices given in Table VII were computed on the non-edited segmentation outcome.

We will not go any further into deep-learning experiments. The goal here was to highlight and illustrate that our method can be easily combined with CNN if needed.

E. Segmentation on Real Biomedical Images

We illustrate the behavior of the proposed active subdivision surface on two real biomedical images with unknown ground truth. We rely on qualitative assessments to validate the accuracy of the segmentation. The volumetric data of Figure 14 represent the optical density of the neuron of a rat in a 3D microscopic image [52]. The volumetric data of Figure 15 results from the MRI acquisition of a human brain, with the purpose of measuring its total intracranial volume (TIV). TIV is used in medicine to detect morphological changes related to the evolution of neurological diseases [53]. This particular segmentation task is challenging because of the numerous concavities that compose the brain, such as the convoluted areas formed around the temporal lobe and the cerebellum. As a point of a comparison, we provide in Figure 16 the TIV

(a) Initialization.   (b) Intermediate stage.   (c) Segmentation outcome.
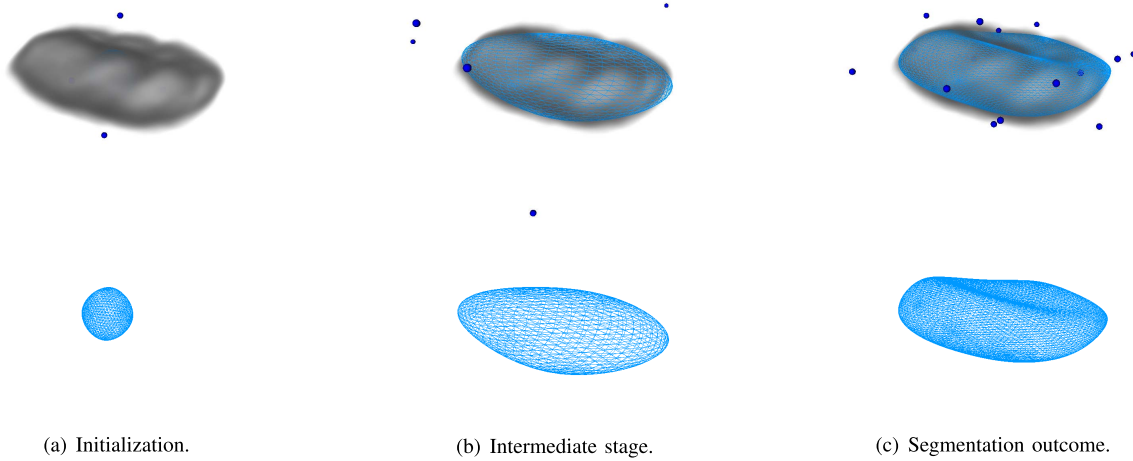
Fig. 14.   Segmentation of the nucleus of the neuron of a rat in a 3D microscopic volume with the active subdivision surface. Blue dots: control points.

(a) Initialization.   (b) Intermediate stages.   (c) Segmentation outcome.
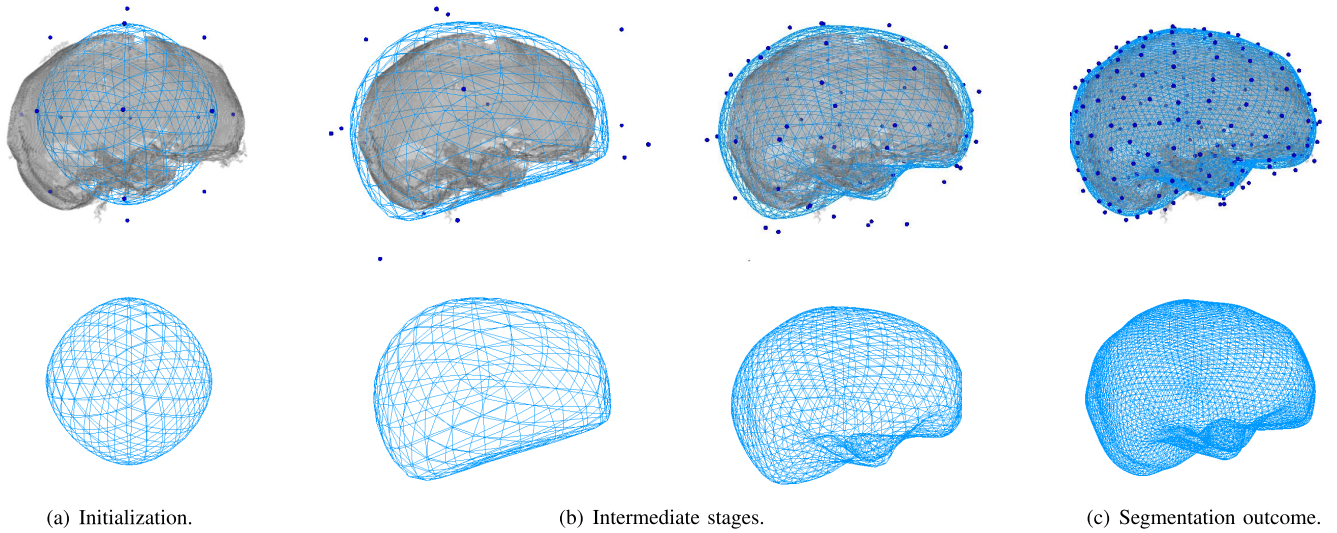
Fig. 15.   TIV segmentation of a 3D MRI scan with the active subdivision surface. Blue dots: control points.

segmentation outcomes obtained with the 3D active mesh and the active parametric surface. Besides being the most accurate, our method is also the fastest: it took minutes while the active mesh took hours.

F. Limitations

In this section, we identify some limitations of our method and suggest options to address them.

1) Initialization: Like all deformable models, our active subdivision surface requires the specification of an initial configuration. In the present implementation, it is placed manually by the user through a graphical interface. However, this approach is appropriate neither for automatization nor for applications with numerous objects to segment in a single image. In this case, the initial position ought to be provided by some auxiliary rough detection algorithms (*e.g.*, DoG filtering followed by a local maxima detection or atlas-based segmentation methods) applied either to the original volume or to the probability map produced by a CNN. Moreover, as the energy E_{region} in (11) is sensitive to the image contrast between the core and the shell of the active subdivision surface, the initial configuration ought to be such that the core intersects the object of interest and the shell intersects the background.

2) Energy Functional: Since our energy terms (8) and (11) are based on intensity information only, they do not perform well when the contrast between the object of interest and the background is low. In this context, an efficient approach is to find discriminative features, by making use of CNN for instance, as illustrated in Section IV-D.

3) Segmentation of Touching Objects: With our method, the segmentation of touching objects may result in overlapping segmentations as two active subdivision surfaces are optimized independently. A solution is to include in our framework a collision detection between two meshes, as the ones described in [8] and [26]. Another solution is to design a new energy term to repulse meshes that would be optimized jointly.
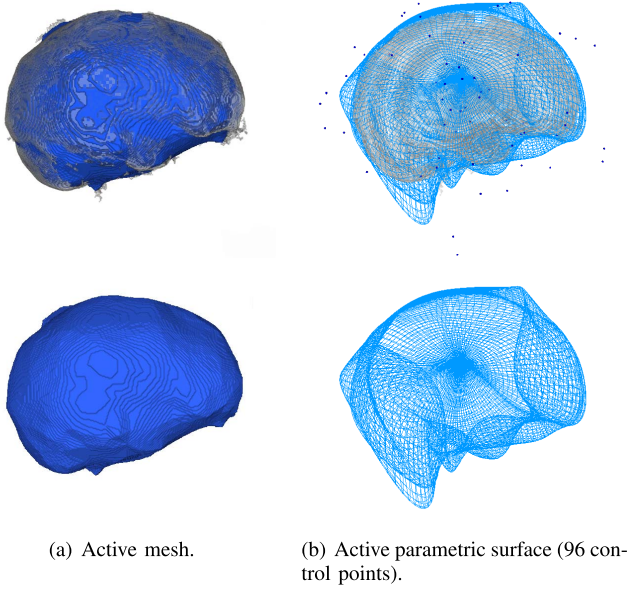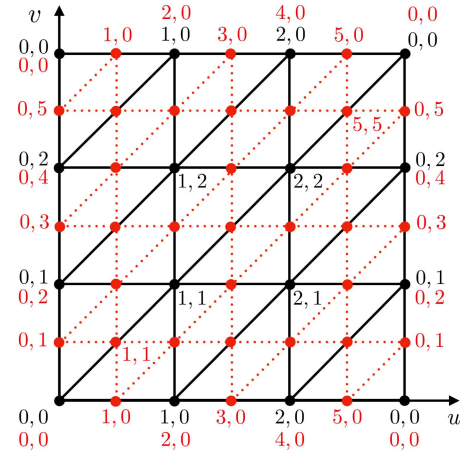
Fig. 17. Flat image of a regular mesh that forms a closed surface. We show the mesh for two subdivision steps of a face-splitting subdivision scheme. The initial coarse mesh $\mathcal{M}_{(0)}$ (solid black lines) is made of $N_0 = 9$ control points that are on a rectangular grid of size $(N_{0,u} \times N_{0,v})$, with $N_{0,u} = N_{0,v} = 3$. Note that some of the control points appear several times in this figure as it is a flat representation of a closed surface. At the first subdivision step, we obtain a regular mesh $\mathcal{M}_{(1)}$ (dotted red lines) made of $2^2 N_0 = 36$ vertices (black and red disks) that are on a rectangular grid of size $(2 N_{0,u} \times 2 N_{0,v})$. The indices correspond to the pair (p, q) of $\mathbf{p}_{(k)}[p, q], k = 0, 1$.

(a) Active mesh.   (b) Active parametric surface (96 control points).

Fig. 16. 3D views of TIV segmentation outcomes. Blue dots: control points.

V. CONCLUSION

Our main contribution in this paper is a generic framework that specifies active surfaces based on subdivision for the semiautomatic segmentation of volumetric biomedical structures. Our active surfaces are characterized by a small set of control points, which eases the interactions with the user. Such deformable models can approximate closed surfaces with arbitrary precision by iteratively refining the underlying mesh. We have provided a specific example of our construction using a widely used subdivision scheme: Loop's scheme. We have designed a surface-based energy as well as a robust region-based energy. We have also proposed an integrated coarse-to-fine optimization strategy to adapt the resolution of the mesh to the level of details in the volume. It results in speedup of the optimization and better robustness. Moreover, this multiresolution strategy allows us to maintain a favorable sampling of the mesh by gradually increasing the flexibility of the model during the optimization. We have applied our proposed method to a variety of problems that involve synthetic data and real biomedical images. We have compared our framework with several segmentation methods and shown that our model is robust with respect to noise and initial conditions.

APPENDIX

A. Definition, Notation, and Properties of Regular Meshes

A triangular mesh without extraordinary vertices (see Section II-A) is called regular. In this section, we present properties of regular meshes that will be used in the proofs of Propositions 1 and 2.

For subdivision schemes applied to regular meshes, the vertices of a mesh $\mathcal{M}_{(k)}$, $k \geq 0$, are on a rectangular grid (u, v) and

$$N_k = 2^{2k} N_0. \tag{14}$$

Let the N_0 control points be on a rectangular grid of size $(N_{0,u} \times N_{0,v})$, such that $N_{0,u}, N_{0,v} \in \mathbb{N}$, and $N_{0,u} N_{0,v} = N_0$. We re-express the set $\mathbb{P}_{(k)}$ as $\mathbb{P}_{(k)} = \{\mathbf{p}_{(k)}[p, q] \in \mathbb{R}^3, \ p \in \{0, \ldots, 2^k N_{0,u} - 1\}, q \in \{0, \ldots, 2^k N_{0,v} - 1\}\}$ (Figure 17).

For convergent subdivision schemes applied to regular meshes, we have the property that

$$\lim_{k \to \infty} \mathbf{p}_{(k+l)}[2^k p, 2^k q] = \boldsymbol{\sigma}(u, v)\big|_{(u,v)=\left(\frac{p}{2^l}, \frac{q}{2^l}\right)}, \tag{15}$$

where $l \in \mathbb{N} \cup \{0\}$, $0 \leq p \leq 2^l N_{0,u}$ and $0 \leq q \leq 2^l N_{0,v}$ [54].

B. Proof of Proposition 1

We first recall a classical result.

Theorem of the Double Riemann Sum: Let $g : [a, b] \times [c, d] \to \mathbb{R}^2$ be a real continuous function that is Riemann-integrable on $[a, b] \times [c, d]$. The Riemann sum $R_{n,m}$ defined by

$$R_{n,m} = \frac{b-a}{n} \frac{d-c}{m} \sum_{p=0}^{n-1} \sum_{q=0}^{m-1} g(p\frac{b-a}{n}, q\frac{d-c}{m}) \tag{16}$$

satisfies

$$\lim_{n \to +\infty} \lim_{m \to +\infty} R_{n,m} = \int_a^b \int_c^d g(u, v) \mathrm{d}u \mathrm{d}v. \tag{17}$$

We first prove Proposition 1 for regular meshes that are topologically equivalent to a torus. Surfaces that are topologically equivalent to a torus are periodic along u, v and are parameterized by

$$\sigma(u, v) = \sum_{p=0}^{N_{0,u}-1} \sum_{q=0}^{N_{0,v}-1} \mathbf{c}[p, q] \varphi_{N_{0,u}, N_{0,v}}(u - p, v - q), \tag{18}$$

where $(u, v) \in [0, N_{0,u}] \times [0, N_{0,v}]$, the $\mathbf{c}[p, q]$ are the control points, and

$$\varphi_{N_{0,u}, N_{0,v}}(u, v) = \sum_{p \in \mathbb{Z}} \sum_{q \in \mathbb{Z}} \varphi(\text{u-p}N_{0,u}, \text{v-q}N_{0,v}), \quad (19)$$

with $\varphi : \mathbb{R}^2 \to \mathbb{R}$ a suitable basis function.

Using the notation for regular meshes described in Section A, we rewrite E_{grad} as

$$E_{\text{grad}}(f, \mathbb{P}_{(k)}) = -\frac{1}{2^{2k}}$$
$$\times \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} \nabla f(\mathbf{p}_{(k)}[p, q]) \cdot \mathbf{n}(\mathbf{p}_{(k)}[p, q]). \quad (20)$$

Combining (15) and (20), we have that

$$\lim_{k \to \infty} E_{\text{grad}}(f, \mathbb{P}_{(k)}) = -\underbrace{\lim_{k \to \infty} \frac{1}{2^{2k}} \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} g\left(\frac{p}{2^k}, \frac{q}{2^k}\right)}_{E},$$
$$(21)$$

where $g(u, v) = \nabla f(\sigma(u, v)) \cdot \mathbf{n}(\sigma(u, v))$ is Riemann-integrable on $[0, N_{0,u}] \times [0, N_{0,v}]$ because $f, \sigma_1, \sigma_2, \sigma_3 \in \mathcal{C}^1$. We use the theorem of the double Riemann sum with $a = c = 0$, $b = N_{0,u}$, $d = N_{0,v}$, $n = 2^k N_{0,u}$, and $m = 2^k N_{0,v}$ to obtain that

$$E = -\int_0^{N_{0,u}} \int_0^{N_{0,v}} \nabla f(\sigma(u, v)) \cdot \mathbf{n}(\sigma(u, v)) \mathrm{d}u \mathrm{d}v. \quad (22)$$

We have that

$$\mathrm{d}\sigma = (\sigma_u \wedge \sigma_v) \mathrm{d}u \mathrm{d}v$$
$$= \mathbf{n}(\sigma(u, v)) \mathrm{d}u \mathrm{d}v. \quad (23)$$

So, using (22) and (23), we finally obtain that

$$E = -\oiint_{\mathcal{S}} \nabla f(\sigma) \cdot \mathrm{d}\sigma. \quad (24)$$

This concludes the proof for regular meshes with a topology equivalent to a torus. For regular meshes that form closed surfaces with a different topology, the proof is the same. The parameterization (18) still holds for $(u, v) \in [0, N_{0,u}] \times [0, N_{0,v}]$ but the bounds of the sum over q depend on the support of φ; additional conditions over the control points may be necessary.

For meshes with extraordinary vertices, we give a gist of the proof, as a detailed one may imply the introduction of too many notations and notions. Note that, in addition, we validated our energy for meshes with extraordinary vertices in our experiments. The gist of the proof is based on the following mathematical pipeline:

- we consider a portion of the initial mesh that contains one extraordinary vertex only, and we locally express the limit surface σ as the union between the limit point of the extraordinary vertex and the sequence of rings $\{\mathbf{r}_{(k)}\}_{k \geq 0}$ defined by the regular vertices around it [37], [55];

- for each ring $\mathbf{r}_{(k)}$, we apply the reasoning that we used for regular meshes where the Riemann integral is instead defined on a local part of the surface σ;
- finally, doing this for each ring and taking the union over k, we conclude the proof.

C. Proof of Proposition 2

We first recall Gauss' theorem.

Gauss' Theorem: Let $\mathcal{V}$ be a subset of $\mathbb{R}^n$ that is compact and has a piecewise-smooth boundary $\mathcal{S}$. If $\mathbf{G}$ is a continuously differentiable vector field defined on a neighborhood of $\mathcal{V}$, then we have that

$$\oiint_{\mathcal{S}} \mathbf{G} \cdot \mathrm{d}\mathbf{S} = \iiint_{\mathcal{V}} \mathrm{div}(\mathbf{G}) \mathrm{d}V. \quad (25)$$

For this proof, we follow the same procedure as described in the proof of Proposition 1. The gist of the proof for meshes with extraordinary vertices is the same as the one presented in Appendix B. Hereafter, we thus only detail the proof for regular meshes with a topology equivalent to that of a torus.

Using the notation for regular meshes introduced in Section A, we rewrite E_{region} as

$$E_{\text{region}}(f, \mathbb{P}_{(k)}) = \frac{1}{|V(\mathbb{P}_{(k)})| 2^{2k}}$$
$$\times \left(2 \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} F(\mathbf{p}_{(k)}[p, q]) n_1(\mathbf{p}_{(k)}[p, q]) \right.$$
$$\left. - \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} F(\mathbf{p}_{\lambda(k)}[p, q]) n_1(\mathbf{p}_{\lambda(k)}[p, q]) \right). \quad (26)$$

As σ is the limit surface of the subdivision scheme (see Equation (5)), we have that

$$\lim_{k \to \infty} |V(\mathbb{P}_{(k)})| = |V|, \quad (27)$$

where $V(\mathbb{P}_{(k)})$ and V are the volume enclosed by $\mathcal{M}_{(k)}$ and σ, respectively. Combining (15), (26) and (27), we obtain that

$$\lim_{k \to \infty} E_{\text{region}}(f, \mathbb{P}_{(k)}) = \frac{1}{|V|} \lim_{k \to \infty} \frac{1}{2^{2k}}$$
$$\times \left(2 \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} g\left(\frac{p}{2^k}, \frac{q}{2^k}\right) \right.$$
$$\left. - \sum_{p=0}^{2^k N_{0,u}-1} \sum_{q=0}^{2^k N_{0,v}-1} g_\lambda\left(\frac{p}{2^k}, \frac{q}{2^k}\right) \right)$$
$$= E, \quad (28)$$

where the functions $g(u, v) = F(\sigma(u, v)) n_1(\sigma(u, v))$ and $g_\lambda(u, v) = F(\sigma_\lambda(u, v)) n_1(\sigma_\lambda(u, v))$ are Riemann-integrable on $[0, N_{0,u}] \times [0, N_{0,v}]$ because $f, \sigma_1, \sigma_2, \sigma_3, \sigma_{\lambda,1}, \sigma_{\lambda,2}, \sigma_{\lambda,3} \in \mathcal{C}^1$. We use the theorem of the double Riemann sum with $a = c = 0$, $b = N_{0,u}$, $d = N_{0,v}$, $n = 2^k N_{0,u}$, and

$m = 2^k N_{0,v}$ to obtain that

$$
\begin{aligned}
E &= \frac{1}{|V|}\left(2\int_0^{N_{0,u}}\int_0^{N_{0,v}} F(\boldsymbol{\sigma}(u,v))n_1(\boldsymbol{\sigma}(u,v))\mathrm{d}u\mathrm{d}v \right. \\
&\quad \left. - \int_0^{N_{0,u}}\int_0^{N_{0,v}} F(\boldsymbol{\sigma}_\lambda(u,v))n_1(\boldsymbol{\sigma}_\lambda(u,v))\mathrm{d}u\mathrm{d}v \right) \\
&= \frac{1}{|V|}\left(2\int_0^{N_{0,u}}\int_0^{N_{0,v}} \mathbf{G}(\boldsymbol{\sigma}(u,v))\cdot\mathbf{n}(\boldsymbol{\sigma}(u,v))\mathrm{d}u\mathrm{d}v \right. \\
&\quad \left. - \int_0^{N_{0,u}}\int_0^{N_{0,v}} \mathbf{G}(\boldsymbol{\sigma}_\lambda(u,v))\cdot\mathbf{n}(\boldsymbol{\sigma}_\lambda(u,v))\mathrm{d}u\mathrm{d}v \right), \quad (29)
\end{aligned}
$$

where $\mathbf{G}(x_1,x_2,x_3) = (F(x_1,x_2,x_3),0,0)$. Thus, we have that

$$
E = \frac{1}{|V|}\left(2\oiint_S \mathbf{G}\cdot\mathrm{d}\boldsymbol{\sigma} - \oiint_{S_\lambda} \mathbf{G}\cdot\mathrm{d}\boldsymbol{\sigma} \right). \quad (30)
$$

The use of Gauss' theorem then yields that

$$
\begin{aligned}
E &= \frac{1}{|V|}\left(2\iiint_{\mathcal{V}} \mathrm{div}(\mathbf{G})\mathrm{d}V - \iiint_{\mathcal{V}_\lambda} \mathrm{div}(\mathbf{G})\mathrm{d}V \right) \\
&= \frac{1}{|V|}\left(2\iiint_{\mathcal{V}} f\,\mathrm{d}V - \iiint_{\mathcal{V}_\lambda} f\,\mathrm{d}V \right) \\
&= \frac{1}{|V|}\left(\iiint_{\mathcal{V}} f\,\mathrm{d}V - \iiint_{\mathcal{V}_\lambda\setminus\mathcal{V}} f\,\mathrm{d}V \right), \quad (31)
\end{aligned}
$$

which concludes the proof in the case of regular meshes.

REFERENCES

[1] B. De Leener, S. Kadoury, and J. Cohen-Adad, "Robust, accurate and fast automatic segmentation of the spinal cord," *NeuroImage*, vol. 98, pp. 528–536, Sep. 2014.

[2] A. Dufour, R. Thibeaux, E. Labruyère, N. Guillen, and J.-C. Olivo-Marin, "3-D active meshes: Fast discrete deformable models for cell tracking in 3-D time-lapse microscopy," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1925–1937, Jul. 2011.

[3] R. Delgado-Gonzalo, N. Chenouard, and M. Unser, "Spline-based deforming ellipsoids for interactive 3D bioimage segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3926–3940, Oct. 2013.

[4] J. Montagnat, H. Delingette, and N. Ayache, "A review of deformable surfaces: Topology, geometry and deformation," *Image Vis. Comput.*, vol. 19, no. 14, pp. 1023–1040, Dec. 2001.

[5] M. Jacob, T. Blu, and M. Unser, "Efficient energies and algorithms for parametric snakes," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1231–1244, Sep. 2004.

[6] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[7] Y.-T. Chen, "A novel approach to segmentation and measurement of medical image using level set methods," *Magn. Reson. Imag.*, vol. 39, pp. 175–193, Jun. 2017.

[8] Y. Duan and H. Qin, "Intelligent balloon: A subdivision-based deformable model for surface reconstruction of arbitrary topology," in *Proc. 6th ACM Symp. Solid Modeling Appl.*, Ann Arbor, MI, USA, 2001, pp. 47–58.

[9] R. Delgado-Gonzalo, D. Schmitter, V. Uhlmann, and M. Unser, "Efficient shape priors for spline-based snakes," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3915–3926, Nov. 2015.

[10] V. Uhlmann, R. Delgado-Gonzalo, C. Conti, L. Romani, and M. Unser, "Exponential Hermite splines for the analysis of biomedical images," in *Proc. 39th IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 1650–1653.

[11] R. Delgado-Gonzalo, V. Uhlmann, D. Schmitter, and M. Unser, "Snakes on a plane: A perfect snap for bioimage analysis," *IEEE Signal Process. Mag.*, vol. 32, no. 1, pp. 41–48, Jan. 2015.

[12] D. Doo and M. Sabin, "Behaviour of recursive division surfaces near extraordinary points," *Comput.-Aided Des.*, vol. 10, no. 6, pp. 356–360, Nov. 1978.

[13] C. Loop, "Smooth subdivision surfaces based on triangles," M.S. thesis, Dept. Math., Univ. Utah, Salt Lake City, UT, USA, Aug. 1987.

[14] N. Dyn, D. Levine, and J. A. Gregory, "A butterfly subdivision scheme for surface interpolation with tension control," *ACM Trans. Graph.*, vol. 9, no. 2, pp. 160–169, Apr. 1990.

[15] K. Karčiauskas and J. Peters, "Bicubic polar subdivision," *ACM Trans. Graph.*, vol. 26, no. 4, p. 14, Oct. 2007.

[16] A. Myles, K. Karčiauskas, and J. Peters, "Extending catmull-clark subdivision and PCCM with polar structures," in *Proc. 15th IEEE Pacific Conf. Comput. Graph. Appl. (PG)*, Maui, HI, USA, Oct. 2007, pp. 313–320.

[17] A. Badoual, P. Novara, L. Romani, D. Schmitter, and M. Unser, "A non-stationary subdivision scheme for the construction of deformable models with sphere-like topology," *Graph. Models*, vol. 94, pp. 38–51, Nov. 2017.

[18] T. DeRose, M. Kass, and T. Truong, "Subdivision surfaces in character animation," in *Proc. 25th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, Orlando FL, USA, Jul. 1998, pp. 85–94.

[19] A. Lee, H. Moreton, and H. Hoppe, "Displaced subdivision surfaces," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, New Orleans, LA, USA, Jul. 2000, pp. 85–94.

[20] H.-T. D. Liu, V. G. Kim, S. Chaudhuri, N. Aigerman, and A. Jacobson, "Neural subdivision," *ACM Trans. Graph.*, vol. 39, no. 4, p. 124, Jul. 2020.

[21] J. Hug, C. Brechbühler, and G. Székely, "Tamed snake: A particle system for robust semi-automatic segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, vol. 1679, Cambridge, U.K., Sep. 1999, pp. 106–116.

[22] N. Dyn, D. Levin, and J. A. Gregory, "A 4-point interpolatory subdivision scheme for curve design," *Comput. Aided Geometric Des.*, vol. 4, no. 4, pp. 257–268, Dec. 1987.

[23] A. Badoual, D. Schmitter, V. Uhlmann, and M. Unser, "Multiresolution subdivision snakes," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1188–1201, Mar. 2017.

[24] F. Orderud and S. I. Rabben, "Real-time 3D segmentation of the left ventricle using deformable subdivision surfaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Anchorage, AK, USA, Jun. 2008, pp. 1–8.

[25] P. H. Kitslaar, R. van't Klooster, M. Staring, B. P. F. Lelieveldt, and R. J. van der Geest, "Segmentation of branching vascular structures using adaptive subdivision surface fitting," *Proc. SPIE*, vol. 9413, Feb. 2015, Art. no. 94133Z.

[26] F. de Chaumont *et al.*, "Icy: An open bioimage informatics platform for extended reproducible research," *Nature Methods*, vol. 9, no. 7, pp. 690–696, Jul. 2012.

[27] L.-E. Andersson and N. F. Stewart, *Introduction to the Mathematics of Subdivision Surfaces*, vol. 120. Philadelphia, PA, USA: SIAM, 2010.

[28] M. Charina, C. Conti, and L. Romani, "Reproduction of exponential polynomials by multivariate non-stationary subdivision schemes with a general dilation matrix," *Numerische Math.*, vol. 127, no. 2, pp. 223–254, Jun. 2014.

[29] C. Conti and L. Romani, "Algebraic conditions on non-stationary subdivision symbols for exponential polynomial reproduction," *J. Comput. Appl. Math.*, vol. 236, no. 4, pp. 543–556, Sep. 2011.

[30] D. Zorin, P. Schröder, and W. Sweldens, "Interpolating subdivision for meshes with arbitrary topology," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, New Orleans, LA, USA, Aug. 1996, pp. 189–192.

[31] L. Kobbelt, "Interpolatory subdivision on open quadrilateral nets with arbitrary topology," in *Proc. Comput. Graph. Forum (EUROGRAPHICS)*, Poitiers, France, Aug. 1996, vol. 15, no. 3, pp. 409–420.

[32] C. Deng and W. Ma, "A unified interpolatory subdivision scheme for quadrilateral meshes," *ACM Trans. Graph.*, vol. 32, no. 3, p. 23, Jun. 2013.

[33] P. Novara and L. Romani, "On extraordinary rules of quad-based interpolatory subdivision schemes," *Comput. Aided Geometric Des.*, vols. 35–36, pp. 225–242, May 2015.

[34] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Comput.-Aided Des.*, vol. 10, no. 6, pp. 350–355, Nov. 1978.

[35] M.-E. Fang, W. Ma, and G. Wang, "A generalized surface subdivision scheme of arbitrary order with a tension parameter," *Comput.-Aided Des.*, vol. 49, pp. 8–17, Apr. 2014.

[36] M. Charina, C. Conti, N. Guglielmi, and V. Protasov, "Regularity of non-stationary subdivision: A matrix approach," *Numerische Math.*, vol. 135, no. 3, pp. 639–678, Mar. 2017.

[37] C. Conti, M. Donatelli, L. Romani, and P. Novara, "Convergence and normal continuity analysis of nonstationary subdivision schemes near extraordinary vertices and faces," *Construct. Approximation*, vol. 50, no. 3, pp. 457–496, Jul. 2019.

[38] J. Peters and U. Reif, *Subdivision Surfaces*, vol. 3. Berlin, Germany: Springer, 2008.

[39] M. Donatelli, P. Novara, L. Romani, S. Serra-Capizzano, and D. Sesana, "A merged tuning of binary and ternary Loop's subdivision," *Comput. Aided Geometric Des.*, vol. 69, pp. 27–44, Feb. 2019.

[40] P. Thevénaz, R. Delgado-Gonzalo, and M. Unser, "The ovuscule," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 382–393, Feb. 2011.

[41] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel, "Laplacian surface editing," in *Proc. Eurograph./ACM SIGGRAPH Symp. Geometry Process.*, Nice, France, Jul. 2004, pp. 175–184.

[42] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. Cambridge, U.K.: Cambridge Univ. Press, 1986.

[43] M. L. Coleman, E. A. Sahai, M. Yeo, M. Bosch, A. Dewar, and M. F. Olson, "Membrane blebbing during apoptosis results from caspase-mediated activation of ROCK I," *Nature Cell Biol.*, vol. 3, no. 4, pp. 339–345, Apr. 2001.

[44] M. Michalkiewicz, J. K. Pontes, D. Jack, M. Baktashmotlagh, and A. Eriksson, "Implicit surface representations as layers in neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Seoul, South Korea, Oct. 2019, pp. 4743–4752.

[45] Y. Shen, Z. Fang, Y. Gao, N. Xiong, C. Zhong, and X. Tang, "Coronary arteries segmentation based on 3D FCN with attention gate and level set function," *IEEE Access*, vol. 7, pp. 42826–42835, Mar. 2019.

[46] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, vol. 9351, Munich, Germany, Oct. 2015, pp. 234–241.

[47] D. Svoboda, M. Kozubek, and S. Stejskal, "Generation of digital phantoms of cell nuclei and simulation of image formation in 3D image cytometry," *Cytometry A, J. Int. Soc. Adv. Cytometry*, vol. 75, no. 6, pp. 494–509, Jun. 2009.

[48] D. Zhang, G. Huang, Q. Zhang, J. Han, J. Han, and Y. Yu, "Cross-modality deep feature learning for brain tumor segmentation," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107562.

[49] D. Zhang, J. Zhang, Q. Zhang, J. Han, S. Zhang, and J. Han, "Automatic pancreas segmentation based on lightweight DCNN modules and spatial prior propagation," *Pattern Recognit.*, vol. 114, Jun. 2021, Art. no. 107762.

[50] A. L. Simpson *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," Feb. 2019, *arXiv:1902.09063*. [Online]. Available: http://arxiv.org/abs/1902.09063

[51] B. H. Menze *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[52] S. A. D. Silveira *et al.*, "Phosphorylation does not prompt, nor prevent, the formation of α-synuclein toxic species in a rat model of Parkinson's disease," *Human Mol. Genet.*, vol. 18, no. 5, pp. 872–887, Dec. 2008.

[53] I. B. Malone *et al.*, "Accurate automatic estimation of total intracranial volume: A nuisance variable with less nuisance," *NeuroImage*, vol. 104, no. 1, pp. 366–372, Jan. 2015.

[54] L. Romani, V. H. Mederos, and J. E. Sarlabous, "Exact evaluation of a class of nonstationary approximating subdivision algorithms and related applications," *IMA J. Numer. Anal.*, vol. 36, no. 1, pp. 380–399, Mar. 2016.

[55] G. Umlauf, "Analyzing the characteristic map of triangular subdivision schemes," *Construct. Approximation*, vol. 16, no. 1, pp. 145–155, Jan. 2000.

Anaïs Badoual received the M.Sc. degree in information technology and communication for health from Télécom Physique Strasbourg, France, in 2014, the M.Sc. degree in image processing from the University of Strasbourg in 2014, and the Ph.D. degree from the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, Switzerland, in 2019, under the supervision of Michael Unser. She is currently a Postdoctoral Researcher with the Serpico Team, Inria, Rennes, France. Her research interests include subdivision theory, deep learning, and segmentation problems.

Lucia Romani received the master's degree in mathematics from the University of Bologna, Italy, in 2000, and the Ph.D. degree in computational mathematics from the University of Padua, Italy, in 2004. She is currently an Associate Professor of numerical analysis with the University of Bologna. She is also the Program Director of the SIAM activity group on Geometric Design. Her main research interests include numerical methods for computer aided geometric design, approximation theory and algorithms, and multiresolution mesh-based methods and meshless methods. She serves as the Co-Editor-in-Chief for the *Journal of Computational and Applied Mathematics* (Elsevier) and an Associate Editor for the *Journal Computer Aided Geometric Design* (Elsevier).

Michael Unser (Fellow, IEEE) was with the Biomedical Engineering and Instrumentation Program, National Institutes of Health, Bethesda, USA, conducting research on bioimaging, from 1985 to 1997. He is a Professor and the Director of the EPFL's Biomedical Imaging Group, Lausanne, Switzerland. He is the coauthor, along with P. Tafti, of the book *An Introduction to Sparse Stochastic Processes* (Cambridge University Press, 2014). His primary area of investigation is biomedical image processing. He is internationally recognized for his research contributions to sampling theory, wavelets, the use of splines for image processing, stochastic processes, and computational bioimaging. He has published over 350 journal articles on those topics.

Prof. Unser is an EURASIP Fellow in 2009 and a member of the Swiss Academy of Engineering Sciences. He was a recipient of several international prizes, including five IEEE-SPS Best Paper Awards, two Technical Achievement Awards from the IEEE 2008 SPS and IEEE EMBS 2010, the Technical Achievement Award from EURASIP in 2018, and a recent Career Achievement Award from IEEE EMBS in 2020. He is the Founding Chair of the Technical Committee on Bio Imaging and Signal Processing (BISP) of the IEEE Signal Processing Society. He has served on the Editorial Board of most of the primary journals in his field, including the IEEE TRANSACTIONS ON MEDICAL IMAGING as an Associate Editor-in-Chief from 2003 to 2005, IEEE TRANSACTIONS ON IMAGE PROCESSING, PROCEEDINGS OF THE IEEE, and *SIAM Journal on Imaging Sciences*.