

Article

A Level Set-Based Model for Image Segmentation under Geometric Constraints and Data Approximation

Guzel Khayretdinova ^{1,2,†}, Dominique Apprato ^{3,†} and Christian Gout ^{1,*,†} 

¹ National Institute for Applied Sciences (INSA Rouen), Laboratoire de Mathématiques de l'INSA, 76000 Rouen, France; guzel.khayretdinova@insa-rouen.fr

² Tomsk State University of Control Systems and Radioelectronics (TUSUR), Television and Control, 634050 Tomsk, Russia

³ Université de Pau et des Pays de l'Adour, LMA-UMR 5142, 64000 Pau, France

* Correspondence: christian.gout@insa-rouen.fr

† These authors contributed equally to this work.

Abstract: In this paper, we propose a new model for image segmentation under geometric constraints. We define the geometric constraints and we give a minimization problem leading to a variational equation. This new model based on a minimal surface makes it possible to consider many different applications from image segmentation to data approximation.

Keywords: energy minimization; level set methods; numerical analysis

1. Introduction

Image segmentation has been heavily studied for more than 40 years in image processing, computer science, mathematics, and from a more general framework in image understanding and computer vision. In [1], the authors present the modeling, techniques, and applications of variational image processing from the theory to the implementation, which constitutes an excellent introduction to image segmentation using variational approaches (including a rigorous study of the mathematical formulations). In [2], the authors give a unified approach of basic computational model reviews for image segmentation, including the Mumford–Shah model (see also [3]), region-based variational active contours, edge-based variational snakes, generalized fast marching method (see also [4]), and active contours. In recent years, deep learning (DL) approaches have been developed (see [5] for a precise and recent state of the art): deep learning approaches have shown qualitatively impressive results but their performance is strongly related to labeled data, and this is, of course, a major drawback on many numerical simulations in case of a lack of labeled data (such that in geosciences, or the specific case of medical images, for instance).

In many applications (geosciences, or even specific medical image processing), the availability of ground truth labels is an important limitation of supervised methods in practice. Another limitation also appears from the excessive cost and time taken to annotate images (in [6,7], the authors estimate that this task requires around 1.5 h of annotation per image in the well-known Cityscapes dataset).

To address this constraint, the study of unsupervised domain adaptation procedures applied to semantic segmentation has been recently conducted in the form of self-training [8]. The addition of geometric constraints makes it possible to improve existing models to obtain good results when acquiring training data is complicated or not possible. In [9], the authors underline that the semi-supervised learning technique is a basic principle which constitutes a strong and efficient solution to consider geometrical constraints in learning semantic segmentation. In [7], the authors propose a multi-modal regularization model applied to self-training procedures in an unsupervised domain linked to semantic segmentation; the introduced regularizer significantly improves self-training methods to



Citation: Khayretdinova, G.; Apprato, D.; Gout, C. A Level Set-Based Model for Image Segmentation under Geometric Constraints and Data Approximation. *J. Imaging* **2024**, *10*, 2. <https://doi.org/10.3390/jimaging10010002>

Academic Editor: Luminița Moraru

Received: 17 November 2023

Revised: 17 December 2023

Accepted: 20 December 2023

Published: 22 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

various unsupervised domain adaptation benchmarks for semantic segmentation. In [10], the authors propose an enhanced U-Net model with a novel geometric consistency loss for geometry-informed structural component segmentation of post-earthquake buildings, which is of great interest in many applications. In [11], the authors give a new model including a cost term with geometrical constraints during the 2DCNN processing; this term is constructed on a Dice term linked to intensity pairing, a weighted total variation quantity, a piecewise-constant Mumford–Shah-based quantity (see [3] for more details) enforcing intensity homogeneity, and an area penalization. Adding this new cost term with a standard CNN has greatly improved image segmentation results [11].

In [12], the authors introduce a variational approach, in which they combine the approximation from a set of points (like in the model introduced by Zhao et al. [13]) and the curve evolution approach introduced in Caselles et al. [14] under geometric constraints given by the user (points/curves/patches should be taken into account as geometric conditions, as in [15–18]). This is of great interest in numerical simulations wherein data are incomplete or of insufficient quality. Elsewhere, as stressed in [12], occultation phenomena should appear, making it pertinent to add geometrical constraints in the modeling to guide the image segmentation processing.

In this work, a new model is proposed, improving the segmentation model under geometric constraints, guaranteeing at the same time the convergence (that is not the case in [19]) toward the strong gradients of the image and the approximation of the geometrical data, by giving an energy functional of quadratic type.

In the segmentation process, the geometric constraints (point(s), curve(s), surface patch(es)...) are considered to belong to the searched contour of interest. These conditions are defined manually by the user. The proposed model leads to the minimization of an energy functional, admitting a unique solution, and leading to a variational problem. The introduced model also makes it possible to approximate point cloud datasets, like seafloor or topographic surface approximation (see [20,21] for many applications linked to data approximation). The main focus of this work is about the image segmentation process but we also provide applications to data approximation.

2. Modeling

We propose to segment an image $I : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}$. This image is defined after usual regularization (Gaussian, for example): it is well-known that the Gaussian smoothing operator is useful for noise reduction (see Sonka et al. [22]). We consider such a regularization process since we plan to test our method on (potentially noisy) medical images. In case of a large amount of noise, we refer the readers to [23], where the authors introduce a model suitable for segmenting a range of images that have intensity inhomogeneity, noise, and a combination of both.

We propose to segment I under geometric constraints. These constraints can be a set of points (as considered in this paper), a set of curves, or a set of surface patches. We consider $\Omega \subset \mathbb{R}^3$ to simplify the problem, but other choices can be made.

We denote by D the geometric dataset:

$$D = \{a = (a_1, a_2, a_3) \in \mathbb{R}^3\} \subset \Omega. \quad (1)$$

We denote by d_D the distance function defined by

$$d_D : (x, y, z) \in \Omega \rightarrow d_D(x, y, z) = \inf_{a \in D} (\|(x, y, z) - a\|) \quad (2)$$

corresponding to the Euclidean distance of the point (x, y, z) to the set D .

The problem is then the following: we try to find the surface $S \subset \Omega$, S being located near the points of maximum gradient of I : it corresponds to the points $(x, y, z) \in \Omega$ where

the usual potential $g(\|\nabla I\|) = \frac{1}{1 + \|\nabla I\|^2}$ is minimum and considering that the set D is close to this set of points (meaning that S also approximates the set D).

We now introduce the energy functional $E(S)$:

$$E(S) = \alpha \int_S d_D^2(s) ds + \beta \int_S g^2(\|\nabla I\|)(s) ds \quad (3)$$

where α and β are strictly positive, α permits to control the fidelity criterion to the dataset D , and β controls the attraction force of S linked to the potential $g(\|\nabla I\|)$.

To represent the minimal active contour S , we use a level set approach (see Osher and Sethian [24], or Sethian [25]). Minimizing the energy $E(S)$ defined in Equation (3) can be rewritten using the level set approach with $S = \{(x, y, z) \in \Omega; \Phi(x, y, z) = 0\}$ and Φ the solution of Problem (2):

$$\left| \begin{array}{l} \text{Find } \Phi : \Omega \rightarrow \mathbb{R} \text{ such that} \\ \forall \xi : \Omega \rightarrow \mathbb{R}, F(\Phi) = \min_{\xi} F(\xi) \\ \text{where} \\ F(\xi) = \alpha \int_{\Omega} d_D^2(x, y, z) \delta(\xi(x, y, z)) \|\nabla \xi(x, y, z)\| dx dy dz \\ + \beta \int_{\Omega} g^2(\|\nabla I\|) \delta(\xi(x, y, z)) \|\nabla \xi(x, y, z)\| dx dy dz \end{array} \right. \quad (4)$$

where δ is the Dirac in $\xi(x, y, z)$ (see, for instance, [19], Section 3 for more details).

Problem (4) can be reformulated as

$$\left| \begin{array}{l} \text{Find } \Phi : \Omega \rightarrow \mathbb{R} \text{ such that} \\ \forall \xi : \Omega \rightarrow \mathbb{R}, F(\Phi) = \min_{\xi} F(\xi) \\ \text{where} \\ F(\xi) = \\ \int_{\Omega} (\alpha d_D^2 + \beta g^2(\|\nabla I\|)) \delta(\xi) \|\nabla \xi\| dx dy dz. \end{array} \right. \quad (5)$$

We now suppose that the solution Φ depends on time introducing $t \in]0, T[$, and we consider that $\Phi \in W(]0, T[; V)$ where V is a Sobolev space with $V \hookrightarrow H^2(\Omega)$ to obtain a continuous final contour. The space $W(]0, T[; V)$ is equipped with its usual scalar product

$$(u, v)_{W(]0, T[; V)} = \int_0^T (u(t), v(t))_V dt + \int_0^T \left(\frac{\partial u}{\partial t}(t), \frac{\partial v}{\partial t}(t) \right)_{V'} dt. \quad (6)$$

Considering a level set framework [24], the solution $S(t)$ is the zero level at each instant t

$$S(t) = \{(x, y, z) \in \Omega, \Phi(t, x, y, z) = 0\} \quad (7)$$

where the explicit (“mother”) function Φ is the solution of the following evolution problem:

$$\left| \begin{array}{l} \text{Find } \Phi \in W(]0, T[; V) \text{ such that} \\ J(\Phi) = \min_{\xi \in W(]0, T[; V)} J(\xi) \\ \text{with} \\ J(\xi) = F(\xi) + \frac{1}{2} \frac{\partial}{\partial t} \left(\varepsilon(t) \|\xi(t, \cdot)\|_{L^2(\Omega)}^2 \right), \\ \xi = \xi(\cdot, x, y, z), \text{ where } F(\xi) \text{ is defined in (5) at each} \\ \text{instant } t, \text{ and where} \\ \varepsilon(t) > 0, \Phi(0, \cdot) = \Phi_0 \in L^2(\Omega) \text{ (initial condition).} \end{array} \right. \quad (8)$$

The parameter ε introduced in (8) makes it possible to control the variation on time of the energy in space of the solution Φ of (8).

Of course, it is necessary to approximate the Dirac δ by a continuous function in the functional F (see Figures 1–3).

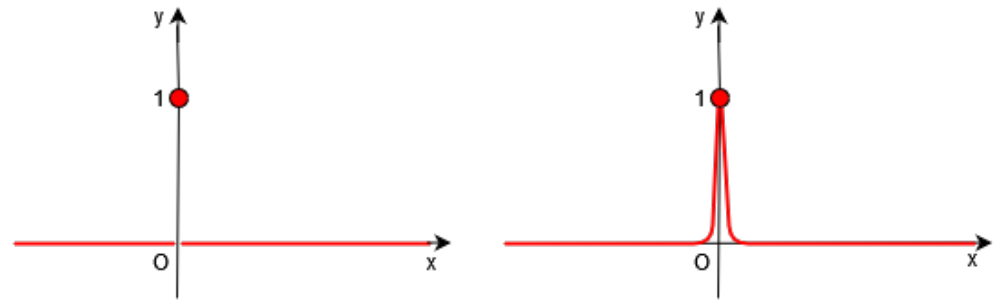


Figure 1. Left: Dirac distribution δ (in red). Right: In Gout et al.'s work [19], the authors introduce a regularized function δ_γ of δ .

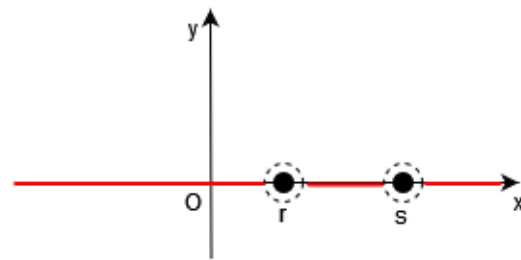


Figure 2. In 1D, in [19], the function $\delta_\gamma(\Phi(t, x, y, z))$ (in red) is equal to 0 outside a neighborhood of (x, y, z) such that $\Phi(t, x, y, z) = 0$, i.e., outside the points $r, s \in S = \{x, \Phi(x) = 0\}$.

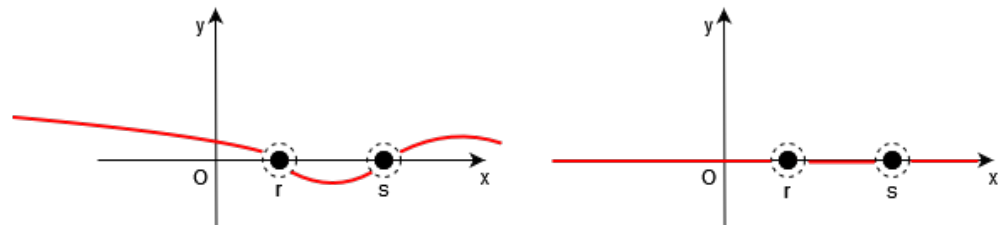


Figure 3. Left: the function $\Phi(t, x, y, z)$ (in red) is close to a constant between the points of S . Right: Thus, we have $\nabla\Phi(t, x, y, z) \cong 0$ (in red) outside the neighborhood of S .

Here, we propose to replace $\delta_\gamma(\xi)$ by $\|\nabla\xi(t, x, y, z)\|$; for any t , it means that, in first approximation, we suppose that for the solution Φ of (8), we consider that $\delta_\gamma(\Phi(t, x, y, z))$ is close to $\|\nabla\Phi(t, x, y, z)\|$ outside a neighborhood of the zero level $S = S(t)$ of $\Phi(t, x, y, z)$.

This choice makes it possible to link the behavior of the solution Φ of (8) with weak variation zones of the image I (zones where the values of the pixels are almost constant, and thus the gradients are close to zero), and this is performed outside the large variation zones of I , that is to say, outside a neighborhood of $S = S(t)$ approximating the set of large variation of I .

We also recall that, like in Gout et al.'s work [19], the term $\frac{1}{2} \frac{\partial}{\partial t} (\varepsilon(t) \|\nabla\xi(x, y, z)\|_{L^2}^2)$ makes it possible to control the variation in time of the energy in space of the solution of Problem (8). We also state that $\varepsilon(t) = \varepsilon > 0$ for any $t \in]0, T[$. This modeling ensures a simultaneous minimization of both d and g .

Moreover, a rescaling can be performed by replacing δ with $\|\nabla\xi(x, y, z)\|$, as performed in Khayretdinova et al.'s work [12]: this rescaling makes it possible to apply the motion to all level sets.

Finally, we obtain the following non-linear energy minimization of a convex functional on the Hilbert space $W(]0, T[; V)$:

$$\begin{array}{|l}
 \text{Find } \Phi \in W([0, T]; V) \text{ such that} \\
 \tilde{J}(\Phi) = \min_{\xi \in W([0, T]; V)} \tilde{J}(\xi) \\
 \text{with} \\
 \tilde{J}(\xi) = \frac{\varepsilon}{2} \frac{\partial}{\partial t} \left(\int_{\Omega} \xi^2(t, x, y, z) dx dy dz \right) \\
 + \int_{\Omega} h(x, y, z) \|\nabla \xi(t, x, y, z)\|^2 dx dy dz \\
 \text{where} \\
 h(x, y, z) = \alpha d_D^2(x, y, z) + \beta g^2(\|\nabla I(x, y, z)\|), \\
 \alpha > 0, \beta > 0, \varepsilon > 0 \text{ and } \Phi(0, \cdot) = \Phi_0 \in L^2(\Omega).
 \end{array} \quad (9)$$

3. Main Results

First, we give the variational formulation of our minimization problem. We then give a result about the minimization problem (the convexity comes from that $\varepsilon > 0$ and $h(x, y, z) > 0$):

Theorem 1. *Problem (9) is a non-linear energy minimization problem of the convex functional $\tilde{J}$ introduced in Equation (9) on the Hilbert space $W([0, T]; V)$ with $V = H^2(\Omega)$.*

The uniqueness of the solution Φ comes from this theorem, and we can obtain the following variational formulation (using differential calculus—Gâteaux derivatives—and functional analysis tools): for any $\xi \in W([0, T]; V)$, $\forall v \in V$, and $\forall t \in]0, T[$

$$\begin{aligned}
 \tilde{J}'(\xi).v = & \varepsilon \int_{\Omega} \frac{\partial \xi}{\partial t} v dx dy dz \\
 & + 2 \int_{\Omega} h v \nabla(\xi(t, x, y, z)) \cdot \nabla(v(t, x, y, z)) dx dy dz
 \end{aligned} \quad (10)$$

where $\nabla(\cdot) \cdot \nabla(\cdot)$ is the Euclidean scalar product in $\mathbb{R}^3$ of two gradients.

If we consider that Φ is the solution of Equation (9), this leads to the following theorem:

Theorem 2. *Problem (9) is equivalent to the following problem*

$$\begin{array}{|l}
 \text{Find } \Phi \in W([0, T]; V) \\
 \text{such that for any } v \in W([0, T]; V), \text{ and } \forall t \in]0, T[\\
 \tilde{J}'(\Phi).v = 0 \\
 \text{with } \Phi(0, \cdot) = \Phi_0 \in L^2(\Omega).
 \end{array} \quad (11)$$

We can rewrite problem (11) as follows.

Problem (9) is equivalent to the following variational problem:

$$\begin{array}{|l}
 \text{Find } \Phi \in W([0, T]; V) \text{ such that for any } v \in W([0, T]; V) \\
 \text{and } \forall t \in]0, T[: \\
 \varepsilon \int_{\Omega} \frac{\partial \Phi}{\partial t} v dx dy dz + a(\Phi, v) = 0 \\
 \text{and } \Phi(0, \cdot) = \Phi_0 \in L^2(\Omega),
 \end{array} \quad (12)$$

where the bilinear form $a(\cdot, \cdot)$ on $V \times V$ is defined as

$$a(u, v) = 2 \int_{\Omega} h(x, y, z) \left[\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} + \frac{\partial u}{\partial z} \frac{\partial v}{\partial z} \right] dx dy dz \quad (13)$$

with $h(x, y, z) = \alpha d_D^2(x, y, z) + \beta g^2(\|\nabla I(x, y, z)\|)$.

Let us note that the bilinear form defined in Equation (13) is symmetric, continuous on $V \times V$ since $h(x, y, z)$ is positive, and superiorly bounded by $(\alpha \times d_H(\Omega, D) + \beta)$ where d_H represents the Hausdorff distance.

4. Numerical Examples

4.1. Discretization of the Variational Problem

The discretization is performed using finite differences in time and finite elements in space. We have chosen the C^1 Bogner Fox Schmidt rectangle as generic finite elements (see Ciarlet [26]). We approximate $a(\cdot, \cdot)$ using a quadrature formula (using the nodes of a voxel grid of image I).

4.1.1. Discretization on Time

We divide $]0, T[$ into m subintervals of equal lengths δt : we introduce the steps $t_m, \forall m \in \left\{1, 2, \dots, \frac{T}{\delta t}\right\}$, and we have

$$t_m = m\delta t.$$

We then use a classical finite difference scheme to approximate $\frac{\partial \Phi}{\partial t}(t, x, y, z)$:

$$\begin{cases} \forall m \in \left\{1, 2, \dots, \frac{T}{\delta t}\right\}, \\ \frac{\partial \Phi}{\partial t}(t_m, x, y, z) \simeq \frac{\Phi(t_m, x, y, z) - \Phi(t_{m-1}, x, y, z)}{\delta t}. \end{cases} \quad (14)$$

4.1.2. Discretization of the Bilinear Form

The main difficulty in this part is the discretization of the function h . This function uses the computation of the distance d_D and the computation of the potential g .

We propose to use a quadrature formula: for any measurable function f on Ω

$$\int_{\Omega} f(x, y, z) dx dy dz \simeq \sum_{i=1}^N \lambda_i f(x_i, y_i, z_i) \quad (15)$$

where $(x_i, y_i, z_i)_i$ are the nodes of the quadrature formula and $(\lambda_i)_i$ the corresponding weights.

Considering that the function h is applied on the values of the image I (via the function g), we choose to take for the nodes $(x_i, y_i, z_i)_i$ the centroids of the voxels of the image I . This makes it possible to compute

$$g^2(\|\nabla I(x_i, y_i, z_i)\|) = \frac{1}{\left(1 + \|\nabla I(x_i, y_i, z_i)\|^2\right)^2}$$

after having discretized the term $(\|\nabla I\|)$ using finite differences.

Moreover, the choice of the weights $(\lambda_i)_i$ should be made such that high degrees polynomials satisfy the quadrature Formula (15), to have a numerical integration error which is consistent with the approximation error of the space $V = H^2(\Omega)$ by the finite element space V_h we will introduce in the following subsection.

Therefore, at this stage, we replace the bilinear form $a(\cdot, \cdot)$ by

$$\begin{cases} \forall u, v \in V, \\ \tilde{a}(u, v) = 2 \sum_{i=1}^N \lambda_i h(x_i, y_i, z_i) \left(\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} + \frac{\partial u}{\partial z} \frac{\partial v}{\partial z} \right) (x_i, y_i, z_i) . \end{cases} \quad (16)$$

4.1.3. Discretization on Time and Space

Following the discretization in time and the discretization of $a(\cdot, \cdot)$, the Problem (12) is approximated by the following: for any $m \in \left\{1, 2, \dots, \frac{T}{\delta t}\right\}$, noting $\Phi^m = \Phi(t_m)$,

$$\begin{cases} \text{Find } \Phi \in W([0, T]; V) \text{ such that, } \forall m \in \left\{1, 2, \dots, \frac{T}{\delta t}\right\}, \forall v \in V, \\ \varepsilon \int_{\Omega} \Phi^m(x, y, z) v(x, y, z) dx dy dz + \delta t \tilde{a}(\Phi^m, v) \\ = \varepsilon \int_{\Omega} \Phi^{m-1}(x, y, z) v(x, y, z) dx dy dz \end{cases} \quad (17)$$

where $\tilde{a}(\cdot, \cdot)$ is defined in (16), with $\Phi^0 = \Phi_0 \in L^2(\Omega)$ (initial condition).

Now, we have to discretize Problem (17) on space; thus, we have to approximate $V = H^2(\Omega)$ by a finite element space $V_h \subset V$.

We consider a meshing (Bogner Fox Schmit, for example, see [26] for more details) corresponding to the voxel grid of the image I .

We then consider a finite element space V_h : V_h is of finite dimension ($\dim V_h = M(h) = M$), and $V_h \subset V$. Let $(\varphi_j)_{j=1, \dots, M}$ be a basis of V_h .

We state

$$\begin{cases} \forall m = 1, \dots, \frac{T}{\delta t}, \\ \Phi^m = \sum_{j=1}^M \alpha_j^m \varphi_j \\ \text{and } \alpha^m = (\alpha_j^m)_{j=1, \dots, M}. \end{cases} \quad (18)$$

As usual, we then take $v = \varphi_l \in V_h \subset V, l = 1, \dots, M$, and using (18), we then deduce that Problem (17) is approximated by

$$\begin{cases} \forall m \in \left\{1, 2, \dots, \frac{T}{\delta t}\right\}, \text{ find } \alpha^m = (\alpha_j^m)_{j=1, \dots, M} \text{ such that, } \forall l = 1, \dots, M, \\ \varepsilon \sum_{j=1}^M \alpha_j^m \int_{\Omega} \varphi_j(x, y, z) \varphi_l(x, y, z) dx dy dz \\ + \delta t \sum_{j=1}^M \alpha_j^m \tilde{a}(\varphi_j, \varphi_l) \\ = \varepsilon \int_{\Omega} \Phi^{m-1}(x, y, z) \varphi_l(x, y, z) dx dy dz \\ \text{with } \Phi^0 = \Phi_0 \in L^2(\Omega) \text{ (initial condition).} \end{cases} \quad (19)$$

Let us note that the initial guess $\Phi^0 = \Phi_0$ is chosen in order to obtain

$$\Phi_0 \in C^k(\Omega) \subset V \subset L^2(\Omega), k \in \mathbb{N}^*,$$

so, this initial condition is regular enough to define its interpolation $\Phi_0^h \in V_h$ such that

$$\Phi_0 \simeq \Phi_0^h = \sum_{j=1}^M \alpha_j^0 \varphi_j \in V_h$$

where $\alpha^0 = (\alpha_j^0)_{j=1, \dots, M}$ are the degrees of freedom of Φ_0 in V_h .

We now write

$$\left\{ \begin{array}{l} \forall m \in \left\{ 1, 2, \dots, \frac{T}{\delta t} \right\}, \text{ find } \alpha^m = (\alpha_j^m)_{j=1, \dots, M} \text{ such that, } \forall l = 1, \dots, M, \\ \varepsilon \sum_{j=1}^M \alpha_j^m \int_{\Omega} \varphi_j(x, y, z) \varphi_l(x, y, z) dx dy dz \\ + \delta t \sum_{j=1}^M \alpha_j^m \tilde{a}(\varphi_j, \varphi_l) \\ = \varepsilon \sum_{j=1}^M \alpha_j^{m-1} \int_{\Omega} \varphi_j(x, y, z) \varphi_l(x, y, z) dx dy dz \\ \text{with } \alpha^0 = (\alpha_j^0)_{j=1, \dots, M} \in \mathbb{R}^M. \end{array} \right. \quad (20)$$

Problem (18) can be written as a linear system, and we first write

$$\left\{ \begin{array}{l} R = (R_{jl}) \in \mathcal{M}_{M,M}(\mathbb{R}) \text{ with} \\ \forall j, l = 1, \dots, M, \\ R_{jl} = \varepsilon \int_{\Omega} \varphi_j(x, y, z) \varphi_l(x, y, z) dx dy dz + \delta t \tilde{a}(\varphi_j, \varphi_l) \\ \text{and} \\ L^{m-1} = \varepsilon (L_l^{m-1}) \in \mathcal{M}_{M,1}(\mathbb{R}) \text{ with } \forall l = 1, \dots, M, \\ L_l^{m-1} = \varepsilon \sum_{j=1}^M \alpha_j^{m-1} \int_{\Omega} \varphi_j(x, y, z) \varphi_l(x, y, z) dx dy dz. \end{array} \right. \quad (21)$$

and then, we obtain the following linear system with (20) and (21):

$$\left\{ \begin{array}{l} \forall m \in \left\{ 1, 2, \dots, \frac{T}{\delta t} \right\}, \\ \text{find } \alpha^m \in \mathbb{R}^M, \text{ such that} \\ R \alpha^m = L^{m-1} \\ \text{with } \alpha^0 \in \mathbb{R}^M. \end{array} \right. \quad (22)$$

4.2. Image Segmentation

To illustrate our approach, we choose to consider the BraTS Dataset [27]. We take an analogous process as we did in [12]: we take 274 MR scans, each with four MIR sequences. The training data have the size $240 \times 240 \times 155$ pixels, and we obtain the (manual) segmentation labels for different brain tumors. We trained the deep network using 79 training data, and we set the initial learning rate as 10^{-4} and multiplied this by 0.5 after every 20 epochs.

To define the geometric constraints, we choose two to three points given by the user and located near the boundary of the part we want to segment. Let us note that the choice of the geometric constraints (two or three points in all our tests) modifies the dice score by less than 6% on the 150 different tests we have conducted on the examples of the second and fourth column of Figure 4, but it reaches 20% on several cases, for example, in the first and third columns of Figure 4. An explanation for this is that, in these images, the contours are rather blurred/noisy. Regarding the sensitivity of the model, we can say that the choice of geometric conditions logically impacts the final result, especially in very noisy areas.

We recall that a high Dice value or a small Hd value represent a high-quality segmentation result. In Table 1, we notice that the computational time is acceptable with our method (Chan–Vese being the fastest). The quality of the segmentation result is analogous: our algorithm and the one of [12] are slightly better. Of course, we have to take into account that our approach is not user-free: the user has to define the geometric conditions (with the mouse), and several parameters have to be defined (time step, space step, α , β , and the initial condition for the level set approach which is a cone in all cases.) Our approach has the benefits of topological independence given by the level set method [24]. Our method

is fast and very useful when not working with many labeled datasets, or when having blurred data or missing data on the image.

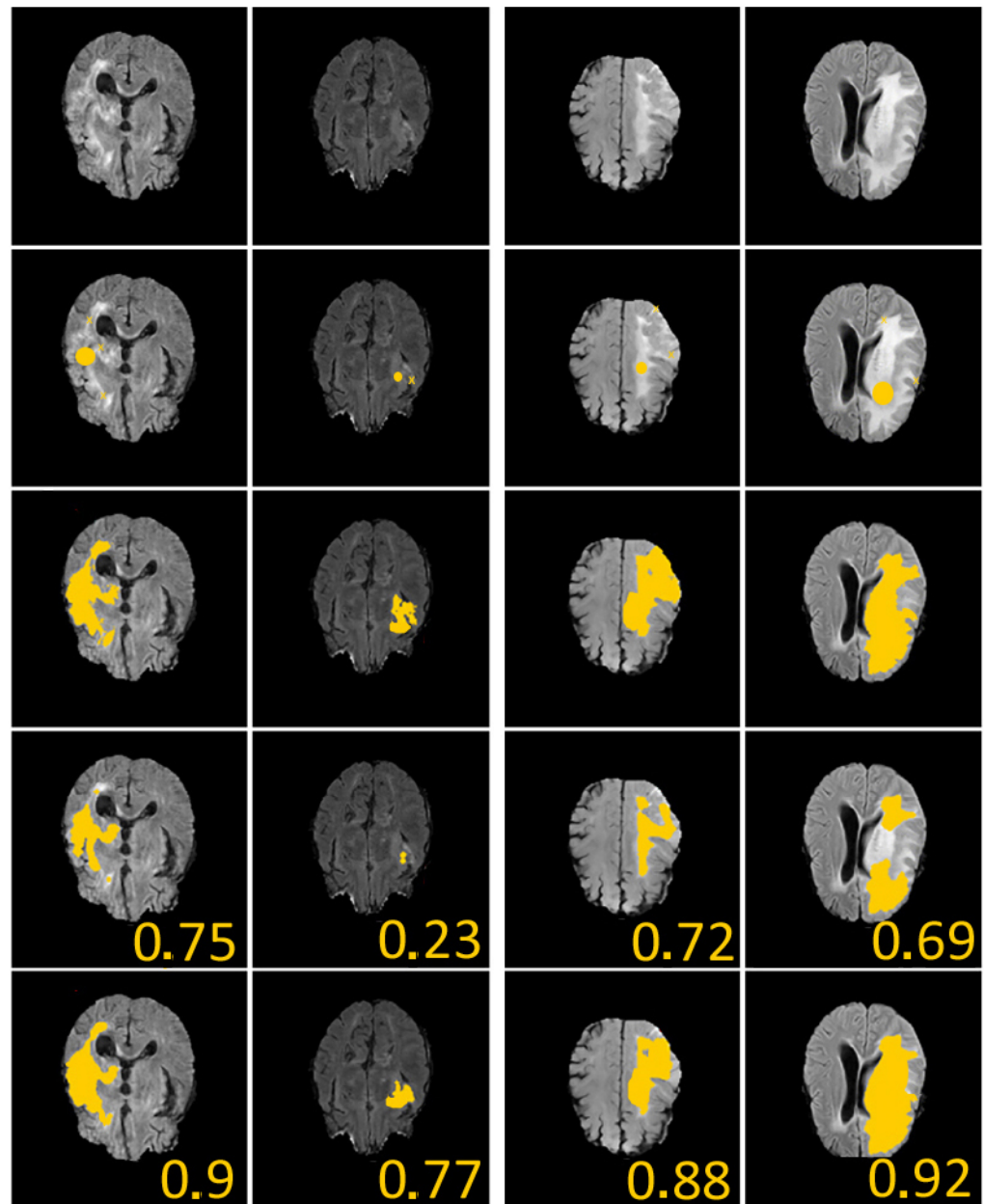


Figure 4. We give 4 examples from the Brats dataset [27] with comparisons between our method and U-Nets. First line: considered images. Second line: initial guess (yellow crosses represent geometrical conditions (set of point(s)), and yellow disc is zero level of initial condition). Third line: ground-truth labels. Fourth line: segmentation obtained using supervised U-Nets [28]. Fifth line: segmentation obtained with our algorithm. In all examples, we considered $\alpha = 1$, $\beta = 1$, $\delta t = 0.3$, and $\delta x = 0.1$. The given values represent the Dice score. These results illustrate the efficiency of our proposed approach.

Table 1. Precision of segmentations models on 1/3 of labeled data from BraTS dataset [27]. The computations were conducted on a Nvidia GeForce RTX 2080 (GPU memory: 11 GB). We compare our method with U-Net, with the image segmentation approach under geometric constraints of [12] (taking equivalent points as initial condition as we did with our method) and with the Chan–Vese segmentation method ([29]) (the initial guess here corresponds to a closed contour located inside the region of interest). We give the results of classical metrics in image segmentation: mean Intersection over Union (mIoU), Dice, Hausdorff distance (Hd), and GPU time for these 4 segmentation methods.

Method	mIoU	Dice	Hd	GPU Time
U-Net [28]	78.3	87.7	43.5	4.45
Chan–Vese [29]	77.6	88.1	41.5	2.02
Khayretdinova et al. [12]	79.6	89.1	39.5	2.12
Our method	79.4	89.1	39.5	2.72

Remark 1. *The main difference between the model presented in this paper and the model given in the work of Gout and Le Guyader [16] is the assurance that we (here) minimize both d and g simultaneously, while the energy to be minimized in [16,19] does not guarantee this point (where the minimization concerns the product $d \times g$). In [12], the advantage is the initial guess (generated from the geometric conditions), but the approximation method is less efficient than the one presented in this paper.*

To test the robustness of our algorithm, we now present several comparisons on noisy images. In Figure 5, we show the considered image (courtesy of CHU Bordeaux, this image represents a slice set perpendicularly to the main pulmonary artery axis). We then artificially add noise to this image, and we compare our algorithm with the one of [12] and the classic geodesic active contour without geometric conditions (in Figures 6 and 7). We can see that our method gives the best results, although it remains sensitive to noise. It is, of course, possible to improve the modeling in the case of applications to image segmentation of noisy images by adding, for instance, the gradient vector flow in the modeling, as stressed in [30].

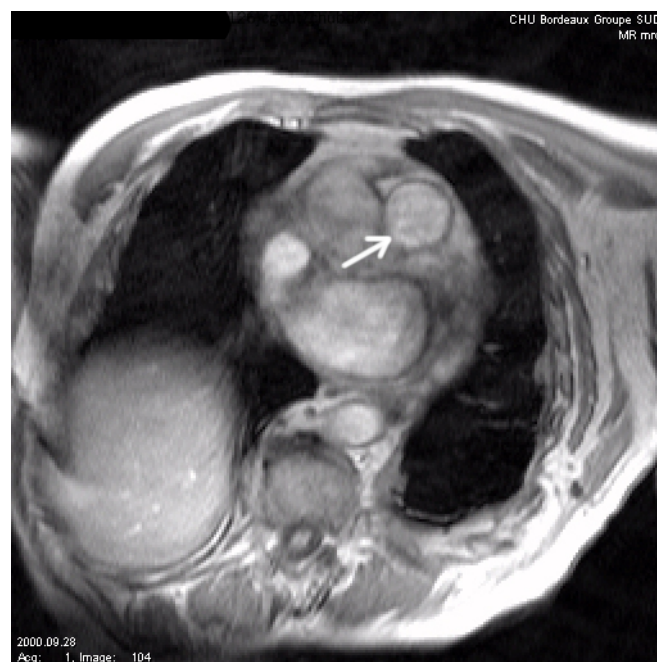


Figure 5. Initial image. The arrow shows the vessel to be segmented (main pulmonary artery).

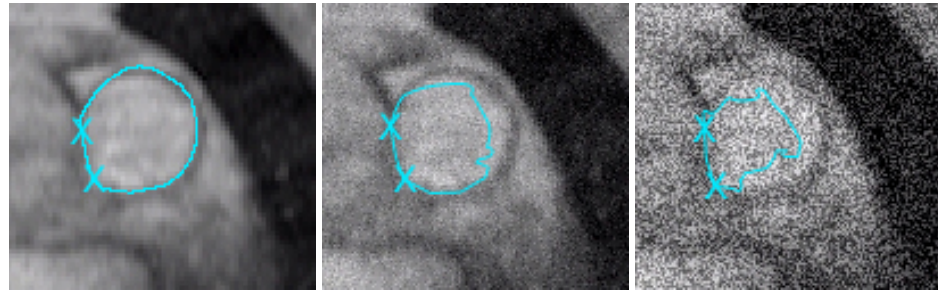


Figure 6. Studied zone around the main pulmonary artery. We use the model proposed in this paper. **Left:** the MPA is perfectly segmented on the initial image (we have considered 2 points as geometric conditions). We obtain equivalent results until adding 40% of noise. **Middle:** after adding 50% of noise on the initial image, the geometric conditions are efficient, but in the right part of the artery, the segmentation contour is (logically) distorted by the noise. **Right:** after adding 200% of noise, the result is of course worse (except near the geometric conditions).

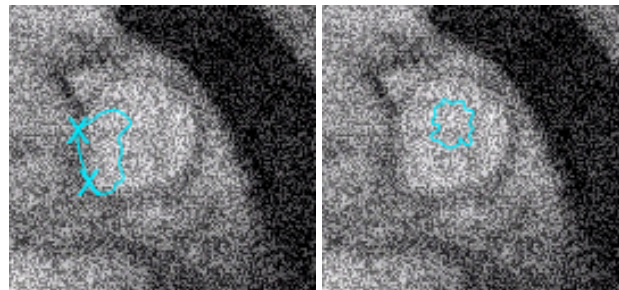


Figure 7. **Left:** with the same geometric conditions as in Figure 6 and on the image with the addition of 200% of noise, we test the algorithm given in [12]. We can see that the result is equivalent to the one of our approach near the 2 points to be interpolated but worse than with our algorithm in other zones. **Right:** the geodesic active contours (without interpolation conditions) do not give a good result (it is well-known that they are sensitive to noise).

4.3. Data Approximation

The main objective of the proposed modeling is image segmentation under geometric constraints, but it is also of interest to underline that our proposed modeling is efficient for data approximation from a large amount of data. Data approximation remains an important research field. For example, ocean mapping to obtain a complete map of the Earth's seabed is a main objective of the next years: this is a crucial point to better understand many environmental challenges from ocean circulation and climate models to tsunami forecasting, cable routing, sediment transportation, renewable energy production, rising of a submarine volcano becoming a new island on a hot spot (like in Hawai'i), etc.

Projects like TOPEX (<https://topex.ucsd.edu/> (accessed on 21 December 2023)) (Scripps Institution of Oceanography, UC San Diego) or more recently GEBCO (https://www.gebco.net/data_and_products (accessed on 21 December 2023)) (International Hydrographic Organization and the Intergovernmental Oceanographic Commission of UNESCO) focus on Earth mapping (especially seafloor surfaces). Another current project is Seabed 2030 (of the Nippon Foundation and the General Bathymetric Chart of the Oceans nonprofit organization), the goal being to map the entire seafloor by 2030 (we are currently at 21%, we were at only 6% in 2017 [31]). Elsewhere, it is of course of interest to propose a surface approximation method from topographic data (from Earth, or other planets like Mars) or lidar/bathymetry data to obtain the value of a surface on every point of a studied domain. Of course, several well-known approaches have been introduced like spline approximation or spline under tension (see [32] for more details), D^m spline [20,21,33,34], or kriging [35]. All these methods have drawbacks like presenting oscillations in the case of rapidly varying data (spline functions), or lack of regularity of the obtained approximant (kriging),

or difficulties in managing significant amounts of data (approximation using polynomial approximation with a significant CPU cost...).

Here, we consider the energy functional $E(S)$ of Equation (3) with $\alpha > 0$ and $\beta = 0$. The initial condition has no impact on the quality of the result (but it has a small impact on the CPU time). The considered dataset was obtained from the TOPEX project. The dataset is constituted of 8736 points giving the seafloor surface and topography of Maui and the Big Island, Hawai'i (Figure 8). This zone (around Mauna Kea) is of interest since it is the largest "mountain" on Earth from base to top (from -7000 m to $+4207$ m, so around $11,207$ m in total) and this zone is in permanent evolution (Hawaiian hot point, with active volcanoes like Kilauea, Mauna Loa, Mauna Kea, or the youngest volcano Kama'ehuakanaloa). We give the approximation in Figure 9.

To compute the error rate, we recall the quadratic error formula:

$$Quad_Err = \left(\frac{\sum_{i=1}^{1500} (\tilde{z}_i - z_i)^2}{\sum_{i=1}^{1500} z_i^2} \right)^{1/2}, \quad (23)$$

where z_i is the value of the z -data, and $\tilde{z}_i$ is the value of the z -approximation for the (same) point (x_i, y_i) .

To compute quadratic errors (Table 2), we only consider 7236 points of the dataset (out of 8736) and we compute the error of Equation (23) on the 1500 randomly deleted points.

Table 2. Error tables: we give the quadratic error (23) between the obtained approximation and the dataset. We give the results for 2 different finite element meshes.

Method	Mesh 20 × 20	Mesh 10 × 10
Spline [21]	0.0000045	0.00026
Kriging [36]	0.0074	0.0074
Our method	0.000068	0.00092

The best error is obtained by the D^m -spline operator of [21], and our proposed method here is (rather) equivalent in terms of error, but faster than the D^m -spline (see Table 3). Kriging is the fastest but the approximation quality is inferior. A drawback of our approach is that it requires an initial condition while it is not necessary when using spline approximation/kriging. The CPU time with our proposed approach can be improved: instead of using finite elements (that have the advantage of guaranteeing a C^1 regularity of the final surface), it is possible to use the fast sweeping scheme (Gauss–Seidel iterations with alternating orderings) to solve the Eikonal equation and the Euler–Lagrange equations can be computed by gradient descent algorithm, and with finite differences in the discretization.

Table 3. Tests are carried out on a 2.7 GHz laptop with an Intel Core i7-7500U CPU @2.70 GHz, 2901 MHz. We give a comparison of the CPU time between the different methods we have tested. We give the results (in seconds) for 2 different finite element meshes. Of course, there is no mesh needed for kriging.

Method	Mesh 20 × 20	Mesh 10 × 10
Spline (Fortran) [21]	26 s	11
Kriging (C++) [36]	4	4
Our method (C++)	16	9

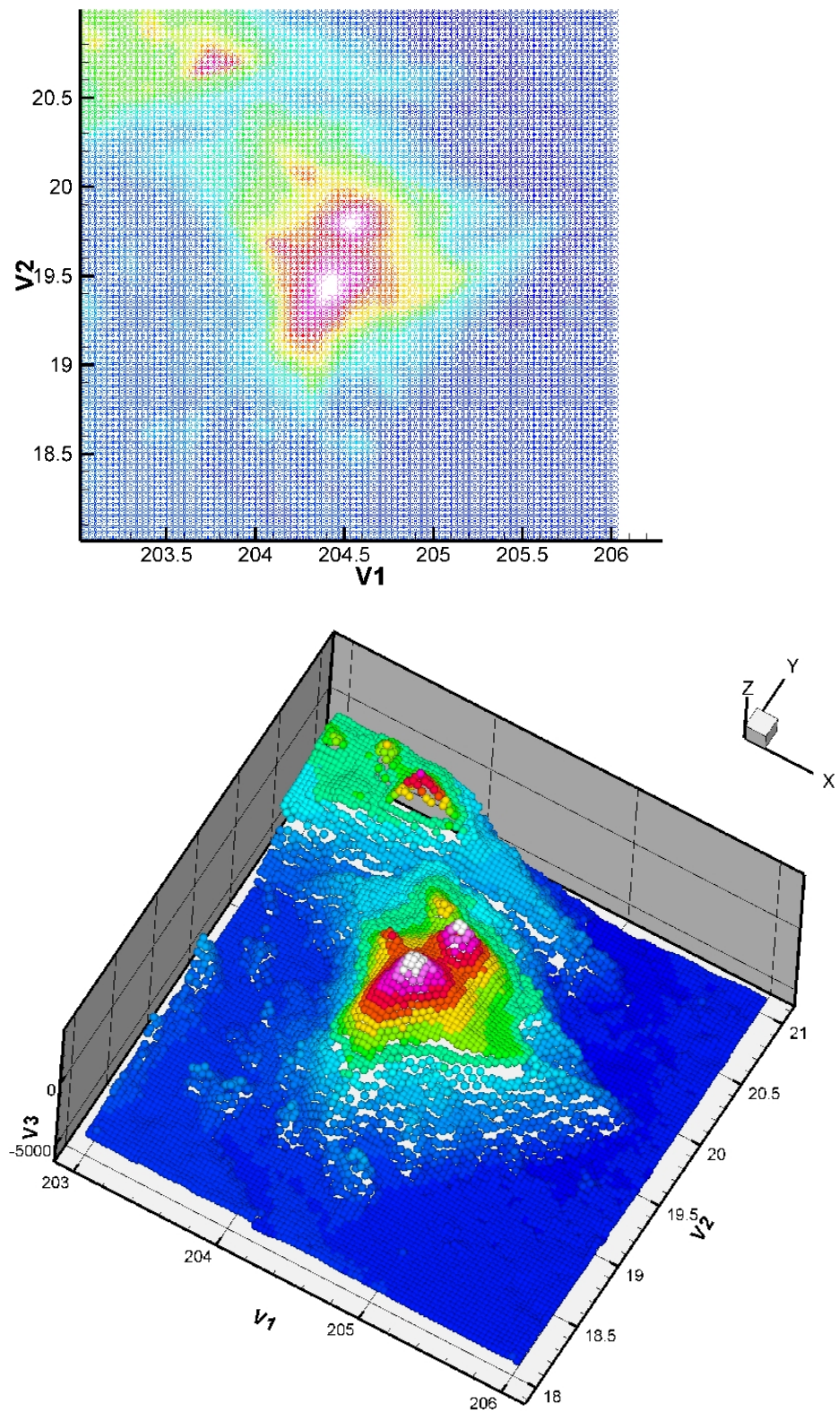


Figure 8. Two-dimensional and three-dimensional views of the dataset: Big Island (Hawaii) zone, 8736 points.

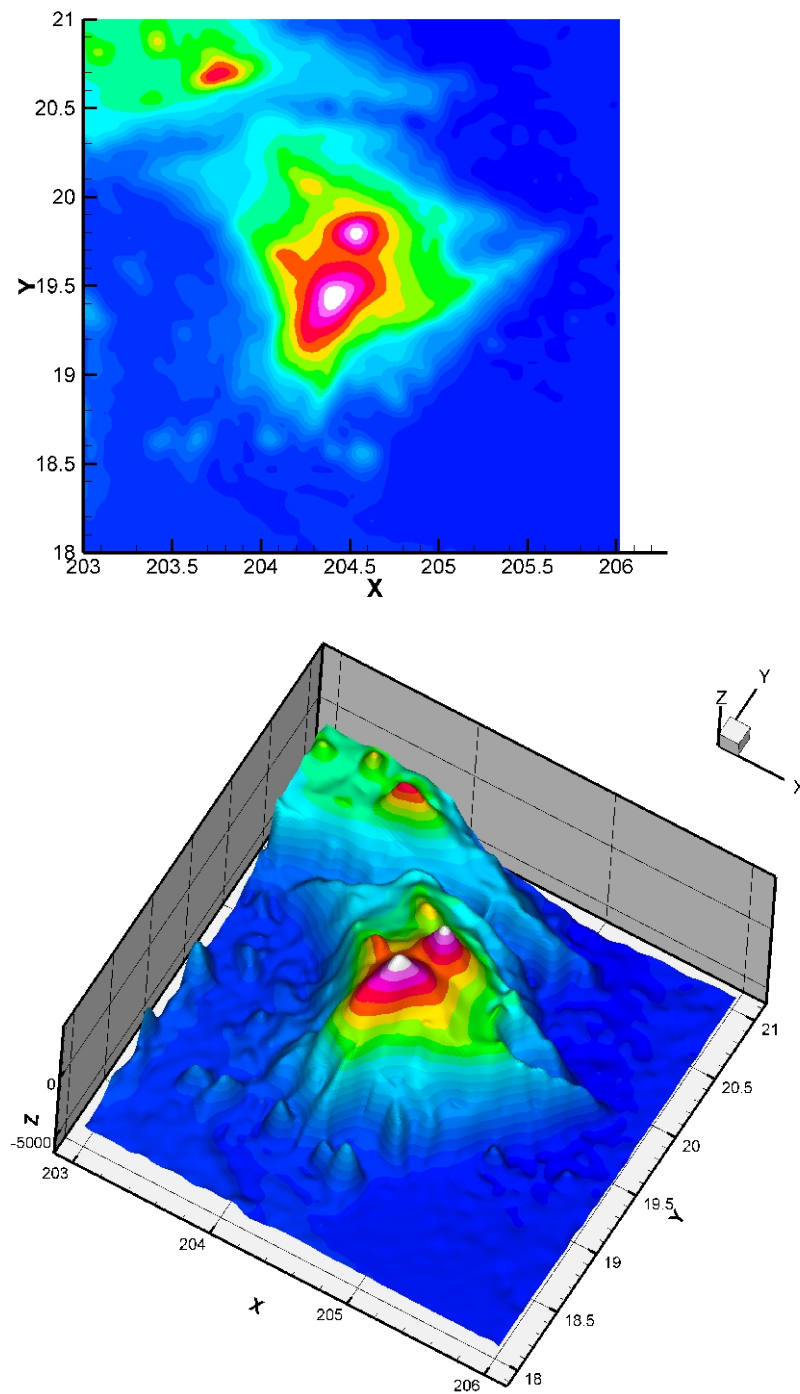


Figure 9. Obtained approximation of the Big Island (Hawaii'i) zone using a finite element grid of 400 Bogner Fox Schmidt rectangles (of class C^1 , see [26] for more details). The step δt is equal to 0.3. The quadratic error (23) is equal to 6.8×10^{-5} . Such quadratic error values are very good in the surface approximation framework, and show that our approach is efficient, even in the case of this rather complex dataset (having large variations). In the global dataset, the maximum error measured is 6%, corresponding to a maximal error of 42 m (the location of this maximum error is logically near the steep valleys).

5. Conclusions

We have proposed efficient modeling for both image segmentation under geometric constraints and data approximation.

For image segmentation, the considered constraints are a set of points to interpolate, but other choices can be made (set of curves, surface patches...). The user defines these points with the mouse.

The role of these geometric constraints is multiple: they can contribute to the acceleration of the convergence of the algorithm by having a similar role as the inflation force of the Balloons model [37], or they can be imposed in the model as wells data in geophysics [17]. These geometric conditions are also useful when image data are blurred or are missing (to help the segmentation process).

We insist on the fact that the goal of our variational approach is not to challenge DL approaches when having a significantly large set of labeled datasets, the goal is mainly to use it on specific cases (when not having a labeled dataset, which is the case in several medical applications and segmentation in geophysics).

In all our segmentation examples, we have kept the same values of the coefficients that modulate the relative weight of the data fidelity term (β) and the one associated with the gradient modulus measure (α) can be optimized to improve the results (but it makes the method less user-free, which is why we have chosen $\alpha = \beta = 1$).

The first numerical results we obtained are very promising. Of course, other tests and comparisons (geophysical datasets like seismic datasets, and 3D datasets in medicine or geosciences...) have to be conducted (work in progress). In Figure 10, we show a 3D geophysical dataset: from this dataset, the goal is to be able to give a visualization of layers and faults like on Figure 11 using a segmentation process; the geometric conditions correspond in this case to wells data, helping the segmentation process to obtain the right layer.

This is a work in progress: a current difficulty we face consists in having a sufficiently good visualization to choose the points (geometric conditions given by the user) inside the 3D bloc; this is unsolved for now (for 3D datasets, we have unsuccessfully tried to link our segmentation process with Paraview [38] and Tecplot [39] so far).

Regarding the parameters, the spatial step is (in general) chosen so that a finite element of the mesh comprises approximately 25 to 100 pixels. Let us note that this leads to more relevant results if we choose a smaller space step (which involves solving a larger linear system). The time step is related to the accuracy of the calculation: less errors will be made between two iterations if we choose a small time step. Values of the order of 10^{-3} seem to be suitable in many tests we have conducted and a smaller number of iterations can be given as a stop criterion.

Moreover, in the energy defined in Equation (3), it is possible to consider the following cases to obtain many different applications:

- $\alpha > 0$ and $\beta > 0$: it corresponds to our proposed segmentation model under geometric conditions.
- $\alpha = 0$ and $\beta > 0$: it corresponds to a basic segmentation model without geometric conditions.
- $\alpha > 0$ and $\beta = 0$: it corresponds to data approximation from a finite set of data with potential applications to seafloor surfaces approximation from various kinds of data (ship tracks data in bathymetry, lidar measurements...) or to shape optimization.

Another work in progress concerns the data approximation of coastal cliffs in Normandy (France) from topographic datasets and from infrared datasets (see Figures 12 and 13).

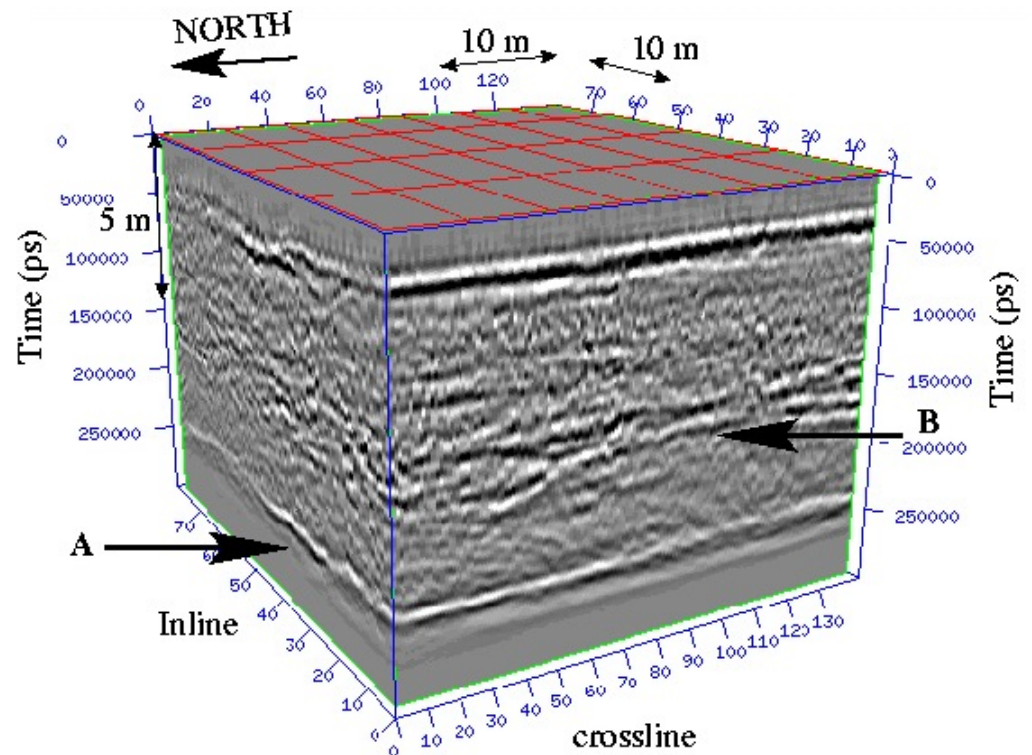


Figure 10. Example of a 3D seismic dataset wherein two continuous reflectors (layer A and layer B) appear.

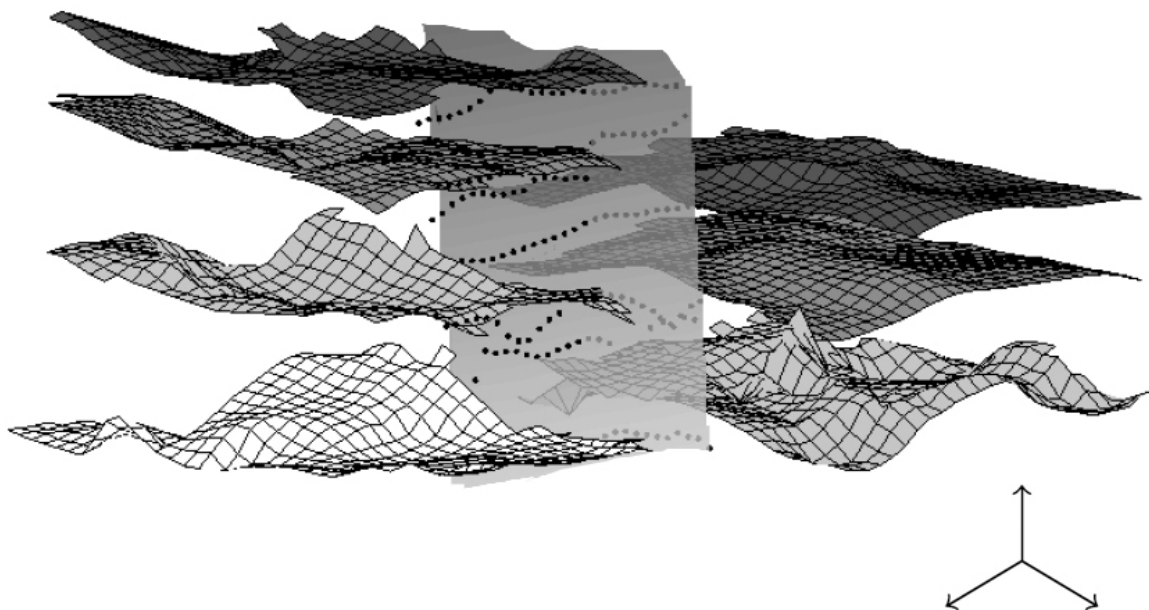


Figure 11. An example of layers and a vertical fault extracted from the complex 3D dataset of Figure 10. Obtaining such visualization requires for a geologist to directly work on the 3D bloc (almost pixel after pixel); we propose to use a segmentation process with geometric constraints to segment one layer after another.

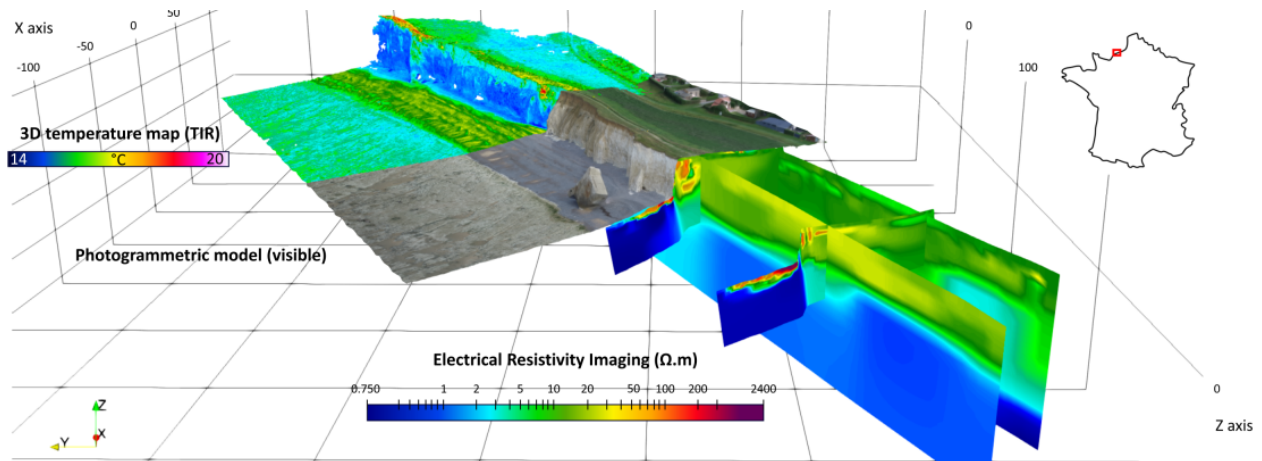


Figure 12. An example of a studied zone in Normandy (Sainte Marguerite cliffs). From different datasets (including acquisition using drones carrying infrared cameras and photogrammetry). The goal is to precisely reconstruct the topography (credits: Defhy3geo project, with Cerema Normandie).

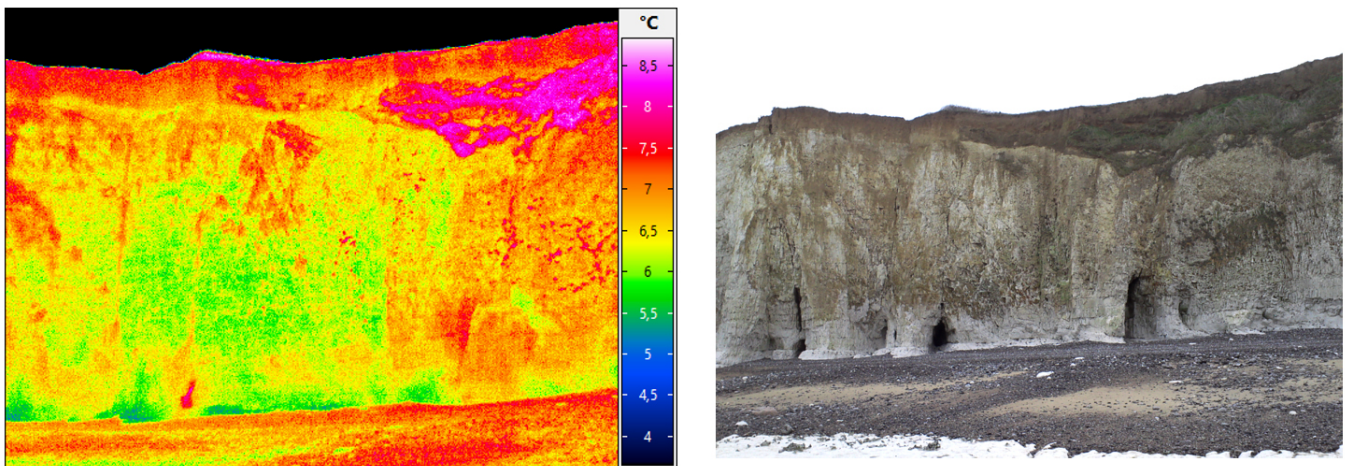


Figure 13. A studied zone with infrared datasets (Vaches noires cliffs, credits: Defhy3geo project, with Cerema Normandie). Approximation of coastal zones is required for many applications like security concerns (cliffs collapsing), or to study the impact of topography on velocity wind fields (Intertwind project).

This general framework is promising, and the model Equation (3) can be developed, adding new kinds of geometric datasets, such as surface patches, Hermite datasets with tangent planes to given 3D datasets, wind velocity fields approximation from lidar datasets, in applications both linked to image segmentation and data approximation.

Author Contributions: Conceptualization, G.K., D.A. and C.G.; methodology, G.K., D.A. and C.G.; software, G.K., D.A. and C.G.; validation, G.K. and C.G.; formal analysis, G.K., D.A. and C.G.; investigation, G.K., D.A. and C.G.; resources, C.G.; writing—original draft preparation, G.K.; writing—review and editing, G.K., D.A. and C.G.; visualization, G.K. and C.G.; supervision, C.G. and D.A.; project administration, C.G.; funding acquisition, C.G. All authors have read and agreed to the published version of the manuscript.

Funding: G. Khayretdinova and C. Gout thank Agence Nationale de la Recherche (ANR MEDISEG, ref. ANR-21-CE23-0013), France Relance and ANR (INTERTWIND project with Siemens Gamesa Renewable Energy), and Région Normandie and ERDF (DEFHY3GEO project, ref. 00130306–21E05300) for supporting this work. The authors also thank Labex AMIES (“Projet Exploratoire” PEPS M2Si2—under convention Labex ANR-10-LABX-002-01) for supporting in part this work.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We acknowledge the Centre Régional Informatique et d'Applications Numériques de Normandie (CRIANN) for supplying computing resources.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Le Guyader, C.; Vese, L. *Variational Methods in Image Processing*, Chapman & Hall/CRC Mathematical and Computational Imaging Sciences Series; Chapman and Hall/CRC: New York, NY, USA, 2015.
2. Antonelli, L.; Simone, V.D.; di Serafino, D. A view of computational models for image segmentation. *Ann. Univ. Ferrara* **2022**, *68*, 277–294. [\[CrossRef\]](#)
3. Mumford, D.; Shah, J. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* **1989**, *42*, 577–685. [\[CrossRef\]](#)
4. Forcadel, N.; Le Guyader, C.; Gout, C. Generalized fast marching method: Applications to image segmentation. *Numer. Algorithms* **2008**, *48*, 189–211. [\[CrossRef\]](#)
5. Minaee, S.; Boykov, Y.; Porikli, F.; Plaza, A.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 3523–3542. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
7. Das, R.; Francis, J.; Mehta, S.V.; Oh, J.; Strubell, E.; Moura, J. Regularizing Self-training for Unsupervised Domain Adaptation via Structural Constraints. *arXiv* **2023**, arXiv:2305.00131.
8. Grandvalet, Y.; Bengio, Y. Semi-supervised learning by entropy minimization. *Adv. Neural Inf. Process. Syst.* **2004**, 1–8.
9. Stekovic, S.; Fraundorfer, F.; Lepetit, V. Casting Geometric Constraints in Semantic Segmentation as Semi-Supervised Learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 1843–1852.
10. Wang, Y.; Jing, X.; Chen, W.; Li, H.; Xu, Y.; Zhang, Q. Geometry-informed deep learning-based structural component segmentation of post-earthquake buildings. *Mech. Syst. Signal Process.* **2023**, *188*, 110028. [\[CrossRef\]](#)
11. Lambert, Z.; Le Guyader, C.; Petitjean, C. A geometrically-constrained deep network for CT image segmentation. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021.
12. Khayretdinova, G.; Chaumont-Frelet, T.; Gout, C.; Kuksenkov, S. Image segmentation with a priori conditions: Applications to medical and geophysical imaging. *Math. Comput. Appl.* **2022**, *27*, 26. [\[CrossRef\]](#)
13. Zhao, H.K.; Chan, T.; Merriman, B.; Osher, S. A variational level set approach to multiphase motion. *J. Comput. Phys.* **1996**, *127*, 179–195. [\[CrossRef\]](#)
14. Caselles, V.; Kimmel, R.; Sapiro, G. Geodesic Active Contours. *Int. J. Comput. Vis.* **1997**, *22*, 61–87. [\[CrossRef\]](#)
15. Ali, H.; Faisal, S.; Chen, K.; Rada, L. Image-selective segmentation model for multi-regions within the object of interest with application to medical disease. *Vis Comput.* **2021**, *37*, 939–955. [\[CrossRef\]](#)
16. Gout, C.; Le Guyader, C. Geodesic active contour under geometrical conditions: Theory and 3D applications. *Numer. Algorithms* **2008**, *48*, 189–211.
17. Le Guyader, C.; Apprato, D.; Gout, C. Using a level set approach for image segmentation under interpolation conditions. *Numer. Algorithms* **2005**, *39*, 221–235. [\[CrossRef\]](#)
18. Rada, L.; Chen, K. A new variational model with dual level set functions for selective segmentation. *Commun. Comput. Phys.* **2012**, *12*, 261–283. [\[CrossRef\]](#)
19. Gout, C.; Le Guyader, C.; Vese, L. Segmentation under geometrical conditions using geodesic active contours and interpolation using level set methods. *Numer. Algorithms* **2005**, *39*, 155–173. [\[CrossRef\]](#)
20. Arcangéli, R.; de Silanes, M.C.L.; Torrens, J.J. *Multidimensional Minimizing Splines: Theory And Applications*; UGA Press: Athens, GA, USA, 2004; ISBN 9781402077869.
21. Gout, C.; Lambert, Z.; Apprato, D. *Data Approximation: Mathematical Modelling and Numerical Simulations*; INSA Rouen Normandie; EDP Sciences: Les Ulis, France, 2019; 168p, ISBN 9978-2-7598-2367-3.
22. Sonka, M.; Hlavac, V.; Boyle, R. *Image Processing, Analysis and Machine Vision*; Springer: New York, NY, USA, 1993.
23. Ali, H.; Rada, L.; Badshah, N. Image Segmentation for Intensity Inhomogeneity in Presence of High Noise. *IEEE Trans. Image Process.* **2018**, *27*, 3729–3738. [\[CrossRef\]](#)
24. Osher, S.; Sethian, J.A. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.* **1988**, *79*, 12–49. [\[CrossRef\]](#)
25. Sethian, J.A. *Level Set Methods and Fast Marching Methods: Evolving interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Material Science*, 2nd ed.; Cambridge University Press: Cambridge, UK, 1999.
26. Ciarlet, P.G. *The Finite Element Methods for Elliptic Problems*; North Holland: Amsterdam, The Netherlands, 1978.

27. Menze, B.H.; Jakab, A.; Bauer, S.; Kalpathy-Cramer, J.; Farahani, K.; Kirby, J.; Burren, Y.; Porz, N.; Slotboom, J.; Wiest, R.; et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Trans. Med. Imaging* **2014**, *34*, 1993–2024. [[CrossRef](#)]
28. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; pp. 234–241.
29. Chan, T.F.; Vese, L. Active contours without edges. *IEEE Trans. Image Process.* **2001**, *10*, 266–277. [[CrossRef](#)] [[PubMed](#)]
30. Xu, H.; Lin, Z.; Guo, Y.; Wang, M. Image Segmentation using GAC Model Combining with GVF and Balloon Force. *J. Algorithms Comput. Technol.* **2015**, *9*, 177–198. [[CrossRef](#)]
31. Fischetti, M. High-Tech Seafloor Mapping Is Finding Surprising Structures Everywhere. *Sci. Am.* **2022**, *327*.
32. Laurent, P.J.; Mehauté, A.L.; Schumaker, L.L. *Curves and Surfaces*; Academic Press Inc.: Cambridge, MA, USA, 1991.
33. Gout, C. C^k surface approximation from surface patches. *Comput. Math. Appl.* **2002**, *44*, 389–406. [[CrossRef](#)]
34. Gout, C.; Komatitsch, D. Surface fitting of rapidly varying data using rank coding: Application to geophysical surfaces. *Math. Geol.* **2000**, *32*, 873–888. [[CrossRef](#)]
35. Cressie, N. The origins of kriging. *Math. Geol.* **1990**, *22*, 239–252. [[CrossRef](#)]
36. Cressie, N.; Johansson, G. Fixed rank kriging for very large spatial data sets. *J. R. Statist. Soc. B* **2008**, *70*, 209–226. [[CrossRef](#)]
37. Liu, M.; Liu, P.; Song, B.; Zhang, Y.; Zhang, L. Active Contour Building Segmentation Model based on Convolution Neural Network. *Iop Conf. Ser. Earth Environ. Sci.* **2022**, *1004*, 012015. [[CrossRef](#)]
38. Paraview. Open Source Post-Processing Visualization Engine. Available online: <https://www.paraview.org> (accessed on 24 August 2023).
39. Tecplot. Tecplot Visualization & Analysis. Available online: <https://www.tecplot.com> (accessed on 24 August 2023).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.