Published in final edited form as:

SIAM J Imaging Sci. 2010 April 30; 3(2): 176–198. doi:10.1137/090766401.

DIFFEOMORPHIC ACTIVE CONTOURS *

Felipe Arrate[†], J. Tilak Ratnanather[‡], and Laurent Younes[§]

[†]Department Applied Mathematics and Statistics, Center of Imaging Science, Johns Hopkins University, 307-B Clark Hall, 3400 N-Charles Street, Baltimore, MD 21218, USA, (felipe.arrate@jhu.edu).

[‡]Center for Imaging Science, Institute for Computational Medicine, The Johns Hopkins University, 3400 N-Charles Street, Baltimore, MD 21218, USA, (tilak@cis.jhu.edu).

[§]Department of Applied Mathematics and Statistics, Institute for Computational Medicine, Center for Imaging Science, The Johns Hopkins University, 3400 N-Charles Street, Baltimore, MD 21218, USA, (laurent.younes@jhu.edu).

Abstract

In this study we present a geometric flow approach to the segmentation of three-dimensional medical images obtained from magnetic resonance imaging (MRI) or computed tomography (CT) scan methods, by minimizing a cost function. This energy term is based on the intensity of the original image and its minimum is found following a gradient descent curve in an infinite-dimensional space of diffeomorphisms (*Diff*) to preserve topology. The general framework is reminiscent of variational shape optimization methods, but remains closer to general developments on deformable template theory of geometric flows. In our case, the metric that provides the gradient is defined as a right invariant inner product on the tangent space ($\mathcal{V}$ at the identity of the group of diffeomorphisms, following the general Lie group approach suggested by Arnold [2]. To avoid local solutions of the optimization problem and to mitigate the influence of several sources of noise, a finite set of control points is defined on the boundary of the template binary images, yielding a projected gradient descent on *Diff*.

Keywords

Groups of Diffeomorphisms; Image Segmentation; Shape Analysis; Deformable Templates

1. Introduction

Since the introduction of *active contours* or *snakes* by Kass et al. [23], the use of parametric active contours has become a common tool in computer vision for edge detection, shape modeling, segmentation and visual tracking to name but a few applications. The original model assumes the deformation of an initially parametrized curve by minimizing an energy functional that is composed of a smoothing term, and one that forces the curve towards the boundary where there is a considerable change in the image intensity.

As an important improvement on snakes a *geometric deformable model* was proposed by Caselles et al. [4] and Malladi et al. [27]. In their approach, a completely intrinsic and

*The work of Felipe Arrate was partially supported by NSF DMS-0456253, ONR N000140810606, NIH P41-RR015241 and R24-HL085343.

Correspondence to: Felipe Arrate.

parametrization-free deformable method is presented where, using level sets [34], evolving curves automatically merge or split following a geometric flow (PDE). In a further development, Caselles et al. [5] cast the classical snakes model in a Riemannian framework, changing the boundary detection problem for the search of minimal length curves (geodesics) with a metric derived from the image content. In this context, the Euler-Lagrange equation obtained from the minimization of the Hamiltonian energy describes a geodesic flow that is later embedded into a level set formulation to allow for topological changes. Independently, Kichenassamy et al. [24] and Yezzi et al. [49] unified classical energy methods and curve evolution algorithms with a reformulation of the Euclidean curve shortening evolution, under a level set formulation. Their work was later extended to deal with two-dimensional surfaces [6], where the optimal segmented boundary was defined by the surface with minimal weighted area.

Recently, it has been suggested by Sundaramoorthi et al. [45], based on the work of Michor and Mumford [29] and Yezzi and Mennucci [50], that the assumed H^0 (or L^2) metric used implicitly in many of the previous studies does not create gradient flows with respect to a well behaved Riemannian metric on the infinite-dimensional space of curves, making necessary the inclusion of a strong penalty term on the curve's length to maintain smoothness. To solve this problem, they introduce a Sobolev norm on the set of perturbations of the space of curves. Although going a long way in improving and stabilizing curve evolution with active contours, this approach does not ensure that the evolving contour maintain important properties over time, like being simple, for example.

In an effort to improve the results of edge-based active contours, region-based methods (Chan and Vese [7], Cremers [11], Grenander et al. [18], Zhu and Yuille [53]) and hybrid combinations of edge and region based active contours (Alexandrov et al. [1], Chen et al. [8], Michailovich [28], Paragios [35], Sundaramoorthi et al. [44]) have been proposed. They mainly combine different cost functions to avoid having an algorithm that is trapped in local minimum, and to increase accuracy. These hybrid models have been extended by several groups to deal with three-dimensional images, with different levels of success [10,20,25,40,42].

The general framework for image segmentation that we will present shares some of the features of the initial snake model and its subsequent improvements, but it is theoretically closer to variational shape optimization methods [12,13,14,15,43] and to general developments on Grenander's deformable template theory of geometric flows [9,17,18,47,51], fitting naturally within the discipline of Computational Anatomy (CA). As it has been described [17,18,19,30], CA introduces the idea of assigning a metric to the space of anatomical structures where shapes are not only considered individually, but as variables belonging to a general space of shapes. In such a space, a representative of the deformable object class (*template*) corresponds to a particular anatomical structure, which, under topologically invariant conditions, is deformed under the action of a group of smooth invertible maps (diffeomorphisms). This setting allows one to focus the modeling effort on the diffeomorphism group rather than on the family of objects being deformed, stimulating a considerable amount of research in the last twenty years [16,32,33] and inspiring an important number of applications in brain studies [22,31,37,38,48] where cortical regions are treated as two-dimensional submanifolds. As our final objective is to study anatomies with known topology, we will follow the viewpoint of CA and consider shapes as points on an infinite-dimensional space. Shape deformation will be defined accordingly as the result of a group action on the space of shape structures, restricting the segmentation to be diffeomorphic to an initial configuration and avoiding the extraction of meaningless morphology resulting from noisy or incomplete data. This will allow us to reproduce large deformations between shapes while preserving their essential non-overlapping constraint.

Furthermore, the segmentation energy that we will define is based on the likelihood of the points in the interior of the template to belong to a target tissue, and its minimization will provide a moving force for the template, making our algorithm more global and closer to region-based procedures, increasing its robustness to noise. The gradient descent procedure that we introduce to minimize this energy has been recently sketched as part of a general review on evolution equations in CA [52], but the main details of the method, its theoretical and computational aspects and its applications to synthetic and medical data are now presented.

The paper is organized as follows: In Section 2, the general mathematical background of Diffeomorphic Gradient Flow is explained. In Section 3, we develop the necessary evolution equations following a *Projected Diffeomorphic Gradient* flow approach. In Section 4, we introduce two possible spatial discretization methods, and discuss general numerical hurdles in the segmentation of biomedical structures obtained from MRI and CT scan. Numerical results are also presented in this Section. Finally in Section 5, remarks about future work and open questions are raised.

2. Diffeomorphic Gradient Flow

In this Section we will detail one general approach to build flows of diffeomorphisms, providing at the same time the necessary notation to understand future sections.

As we have said before, we would like to deform a given ambient space Ω using the action of a group of diffeomorphisms

DEFINITION 2.1. *A diffeomorphism φ of an open subset $\Omega \subset \mathbb{R}^d$ is an invertible differentiable correspondence of Ω onto itself, such that φ^{-1} is continuously differentiable. The set of diffeomorphisms of Ω is denoted by $\text{Diff}(\Omega)$ and it becomes an algebraic group by composition of its elements: if φ and ψ are diffeomorphisms, so is $\varphi \circ \psi^{-1}$. Then, diffeomorphisms act on subsets $\Omega \in \mathbb{R}^3$ by preserving their topology¹.*

In the present paper, we will work with diffeomorphisms that arise as flows of non-autonomous differential equations, following the general framework of [16,30,46,47]. Therefore, in our work, diffeomorphisms inducing arbitrary large deformations will be obtained as solutions of *flow equations* of the form

$$\frac{\partial \varphi(t, \cdot)}{\partial t} = v(t, \varphi(t, \cdot)) \varphi(0, \cdot) = id. \quad (2.1)$$

To ensure that this ODE is well-defined and generates diffeomorphisms, the time dependent vector field $v(t, \cdot)$ has to be smooth enough, as is formally discussed next. In the following we will denote the unique solution at time t of equation (2.1), as $\varphi^v(t, \cdot)$.

2.1. Admissible Vector Spaces

Let Ω be a smooth domain of $\mathbb{R}^d$. Let $C_0^1(\Omega, \mathbb{R}^d)$ denote the set of continuously differentiable vector fields on Ω that vanish, with their first derivatives, at the boundary of Ω and at infinity. We will equip this set with the usual supremum norm, denoted $\|\cdot\|_{1,\infty}$, which

¹In general, diffeomorphisms can act not only on the space Ω , but on various structures supported by Ω . For example if $I: \Omega \rightarrow \alpha$ defines an image on Ω , with $I(x)$ representing a gray level function at x , the associated deformation of the ambient space Ω creates a new image $\tilde{I}$, by setting $\tilde{I}(y) = I(\varphi^{-1}(y))$.

controls the maximum absolute values of the vector field coordinates and their first derivatives.

DEFINITION 2.2. A Banach space $\mathcal{V}$ is called admissible if it is canonically embedded in $C_0^1(\Omega, \mathbb{R}^d)$, so that there exists a constant c such that, for any $w \in \mathcal{V}$

$$\|w\|_{1,\infty} \leq c\|w\|_{\mathcal{V}}.$$

We will denote by $\mathcal{X}_{\mathcal{V}}^1(T, \Omega)$ the set of time-dependent vector fields such that for all fixed $t \in [0, T]$, $v(t, \cdot)$ is a vector field that belongs to $\mathcal{V}$ and

$$\|v\|_{\mathcal{X}_{\mathcal{V}}^1} := \int_0^T \|v(t, \cdot)\|_{\mathcal{V}} dt < \infty. \quad (2.2)$$

The following result extends the classical theory of existence of solutions of differential equations to vector fields in $\mathcal{X}_{\mathcal{V}}^1(T, \Omega)$

THEOREM 2.3 (Dupuis et al. [16], Trouné [46]). If $\mathcal{V}$ is admissible and $t \mapsto v(t, \cdot)$ is in $\mathcal{X}_{\mathcal{V}}^1(T, \Omega)$, the flow equation (2.1) can be uniquely solved, generating a trajectory $\phi^v(t, \cdot)$ in $\text{Diff}(\Omega)$. In addition to Theorem 2.3, one can show that the following group

$$\mathcal{G}_{\mathcal{V}} = \{\phi^v(t, \cdot), t \in [0, T], v \in \mathcal{X}_{\mathcal{V}}^1(T, \Omega)\} = \{\phi^v(1, \cdot), v \in \mathcal{X}_{\mathcal{V}}^1(1, \Omega)\}$$

is a subgroup of the groups of diffeomorphisms. (The definition of $\mathcal{G}_{\mathcal{V}}$ does not depend on T because of the identity $\phi^v(T, \cdot) = \phi^{Tv}(1, \cdot)$.)

For computational purposes, the most interesting admissible spaces are those that are also Hilbert, the norm being associated with an inner product. Moreover, the admissibility condition implies that these spaces have a so-called reproducing kernel which will be important in our computations. In detail, take any point x in Ω , $a \in \mathbb{R}^d$ and a Hilbert admissible space $\mathcal{V}$ and define the continuous linear form $a \otimes \delta_x \in \mathcal{V}'$ (the topological dual space of $\mathcal{V}$ by

$$(a \otimes \delta_x | v) = a^T v(x) \in \mathbb{R}.$$

Then, by the Riesz representation theorem there exists a unique element $K_x^a(\cdot) \in \mathcal{V}$ such that for any $v \in \mathcal{V}$

$$(a \otimes \delta_x | v) = \langle K_x^a, v \rangle_{\mathcal{V}}.$$

For any $y \in \Omega$, the association $a \mapsto K_x^a(y)$ defines a linear map from $\mathbb{R}^d$ to $\mathbb{R}^d$ that we will denote by the matrix $K(y, x)$, so $K_x^a(y) = K(y, x)a$. We note that $\langle K_x^a, K_y^b \rangle = a^T K_y^b(x) = a^T K(x, y)b$ which is equal to $b^T K(x, y)^T a$, implying the relationship $K(x, y)^T = K(y, x)$. In the one-dimensional case this means that $K(x, y)$ is symmetric.

In what follows, we will refer to $K(x, y)$ as the kernel of the space $\mathcal{V}$. Such a kernel is positive by construction, i.e., it is such that, for every family $x_1, \dots, x_N$ of distinct points in Ω

and any sequence $\alpha_1, \dots, \alpha_N \in \mathbb{R}^d$, the scalar $\sum_{i,j} \alpha_i^T K(x_i, x_j) \alpha_j$ is nonnegative and vanishes if and only if $\alpha_i = 0$, for all i .

For future use, we will define $\hat{K}$ as the duality map between $\mathcal{V}$ and $\mathcal{V}'$ so, for example

$$\widehat{K}(a \otimes \delta_x) = K(\cdot, x)a \in \mathcal{V}'. \quad (2.3)$$

An important property of reproducing kernel Hilbert spaces is that they are characterized by their kernel. In fact, as proved by Aronszajn [3], any symmetric, positive definite kernel uniquely determines a Hilbert space, as the abstract completion of the function space provided by all finite linear combinations

$$x \mapsto \sum_i K(x, x_i) \alpha_i.$$

This Hilbert space is admissible in the sense of Definition 2.2 as soon as the kernel $K(x, y)$

vanishes at infinity and $\frac{\partial K}{\partial x}$, $\frac{\partial K}{\partial y}$ and $\frac{\partial^2 K}{\partial x \partial y}$ also exist and vanish at infinity. More generally, for the associated space to contain C^p functions, all partial derivatives of total order $2p$, and of order p with respect to x and y must exist.

In this paper, we will restrict to the use of radial kernels which are invariant to translation and rotations, so that $K(x, y) = \gamma(|x - y|)Id_d$. In particular, for any dimension d we will choose γ to be

$$\gamma(t) = \int_0^{+\infty} e^{-t^2 u} d\mu(u) \quad (2.4)$$

for a given positive measure μ on $(0, +\infty)$. This will constrain the scalar kernel $K(x, y)$ to be positive definite. (This is known as Schoenberg's theorem.)

A special and widely used case is with $\mu = \delta_\sigma^{-2}$; the Dirac measure at σ^{-2} , which yields

$\gamma(t) = e^{-\frac{t^2}{\sigma^2}}$, with an associated *Gaussian kernel*

$$K(x, y) = g_{\sigma^2}(x - y) Id_d \quad \text{where } g_{\sigma^2}(u) = e^{-\frac{\|u\|^2}{\sigma^2}}. \quad (2.5)$$

2.2. Variational Problems over Diffeomorphisms and Shapes

The segmentation energy to be introduced later will be defined as a function $\Omega \mapsto J(\Omega)$ over measurable subsets $\Omega \subset \mathbb{R}^d$. In order to minimize this energy, we will need to compute its first variation under some specific perturbation of the domain Ω . Such a concept is usually defined in shape optimization theory [12,14,15] as the *shape derivative* of $J(\Omega)$. One of its possible constructions uses what is known as the *speed method*, in which the derivative $dJ(\Omega)$ is defined as a linear operator over vector fields $u(\cdot)$ on $\mathbb{R}^d$,

$$dJ(\Omega)(u) = \frac{d}{d\varepsilon} J(\chi(\varepsilon, \cdot)(\Omega))$$

where $\varepsilon \mapsto \chi(\varepsilon, \cdot)$ is a time dependent diffeomorphism, such that

$$\frac{d\chi(\varepsilon, \cdot)}{d\varepsilon} = v(\varepsilon, \chi(\varepsilon, \cdot)) \quad \text{and} \quad \chi(0, \cdot) = id \quad (2.6)$$

in a neighborhood of $\varepsilon = 0$. Here, $v(\varepsilon, \cdot)$ is a time-dependent vector field on $\mathbb{R}^d$ such that $v(0, \cdot) = u(\cdot)$, which is assumed to belong to some admissible family; for example, v can be smooth and compactly supported in time and space.

In addition, we will require the operator $dJ(\Omega)$ to be continuous in u for some suitable topology. This can be the usual topology on smooth compactly supported functions (leading $dJ(\Omega)$ to be a distribution). In fact, we will require $dJ(\Omega)$ to be continuous for a stronger topology, namely for the one associated to an admissible Hilbert space $\mathcal{V}$ so that for any $u \in \mathcal{V}$

$$dJ(\Omega)(u) \leq \text{const} \|u\|_{\mathcal{V}}.$$

When this is true, we have, by definition of $\hat{K}$ in (2.3)

$$dJ(\Omega)(u) = \left\langle \hat{K}(dJ(\Omega)), u \right\rangle_{\mathcal{V}}$$

and $\hat{K}(dJ(\Omega))$ will be referred to as the *shape gradient* of J relative to $\mathcal{V}$ and denoted $\nabla_{\mathcal{V}} J(\Omega)$ or $\nabla J(\Omega)$ if $\mathcal{V}$ is clear from the context.

It is important to notice that this definition can be interpreted (at least formally; a rigorous discussion of this topic being beyond the scope of this paper) as a standard Riemannian gradient. (In the following, the discussion will be similar to the one given in [47]). Indeed, because we want to impose topological constraints, we will always apply the function J to sets Ω that belong in the orbit of a fixed set Ω_0 , under the action of the subgroup $\mathcal{G}_{\mathcal{V}}$, i.e., $\Omega = \psi(\Omega_0)$ for some diffeomorphism $\psi \in \mathcal{G}_{\mathcal{V}}$. Therefore, it will be natural to consider J as a function defined over diffeomorphisms, i.e., $\tilde{J}(\psi) := J(\psi(\Omega_0))$. With this setting, the shape derivative of J at Ω will be nothing but the derivative at $\varepsilon = 0$ of the shape function $\tilde{J}_{\psi}(\chi(\varepsilon)) := \tilde{J}(\chi(\varepsilon, \psi))$ considered as an operator on vector fields, i.e., we can identify $dJ(\Omega)$ and $d\tilde{J}_{\psi}(id)$.

Assume now that the subgroup of diffeomorphisms $\mathcal{G}_{\mathcal{V}}$ is a Riemannian manifold with a right-invariant metric

$$\|u\|_{\psi} := \|u \circ \psi^{-1}\|_{\mathcal{V}}$$

(tangent vectors to diffeomorphisms being identified to vector fields on $\mathbb{R}^d$). We have (still working formally)

$$dJ(\Omega)(u) = \frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \tilde{J}(\chi(\varepsilon, \psi)) = d\tilde{J}(\psi)(u \circ \psi) \quad \text{by} \quad (2.6)$$

Then

$$d\tilde{J}(\psi)(u \circ \psi) = \left\langle \widehat{K}(dJ(\Omega)), u \right\rangle_{\mathcal{V}} = \left\langle \widehat{K}(dJ(\Omega)) \circ \psi, u \circ \psi \right\rangle_{\psi} = \left\langle \bar{\nabla} J(\Omega) \circ \psi, u \circ \psi \right\rangle_{\psi}$$

the last step by right-invariance. This implies that the Riemannian gradient of $\tilde{J}$ is related to the shape gradient $\bar{\nabla} J(\Omega)$ via

$$\nabla \tilde{J}(\psi) = \bar{\nabla} J(\Omega) \circ \psi.$$

This is an important remark since it allows us to express the known gradient descent algorithm in the Riemannian manifold $\mathcal{G}_{\mathcal{V}}$, namely $d\varphi(t, \cdot)/dt = -\nabla J(\varphi(t, \cdot))$, in terms of the shape gradient, yielding

$$\frac{d\varphi(t, \cdot)}{dt} = -\bar{\nabla} J(\Omega(t)) \circ \varphi(t, \cdot) \quad (2.7)$$

where $\Omega(t) = \varphi(t, \Omega_0)$. Since the shape gradient is a vector field

$$v^{\varphi} = \widehat{K}(dJ(\Omega(t)))$$

this implies that, for any point x , $y(t) = \varphi(t, x)$ is a solution of the ODE $dy(t)/dt = -v^{\varphi}(y(t))$ and then $\varphi(t, \cdot)$ is the flow associated to the time-dependent vector field $w(t, \cdot) = -v^{\varphi}(\cdot)$. Such gradient descent algorithms provide a diffeomorphic evolution of Ω .

2.3. Segmentation Energy

At this stage we introduce some notation regarding the segmentation energy that we used in our experiments. Let us consider a finite set

$$\mathcal{L} = \{l_0, l_1, \dots, l_M\}$$

of labels, or *tissue types*, where l_0 is interpreted as a label for the background. We assume that images are preprocessed so that they provide a family of functions $f_1, \dots, f_M$ such that $f_n(x)$ has a low value (perhaps negative) when the image intensity at x belongs to the tissue l_n , and $f_n(x)$ is large otherwise. A typical choice uses intensity segmentation methods with mixtures of Gaussians, typically resulting in posterior probabilities $p(l_n|x)$ at each point x , and sets

$$f_n(x) = \log \frac{p(l_0|x)}{p(l_n|x)}.$$

Our goal is to estimate a family of non-overlapping sets $\Omega^1, \dots, \Omega^M$, in order to minimize

$$J(\Omega^1, \dots, \Omega^M) = \sum_{n=1}^M \int_{\Omega^n} f_n(y) dy. \quad (2.8)$$

Assuming that all the f_n 's are differentiable, we can compute the shape derivative of the energy J by introducing a time-dependent vector field $v(\varepsilon, \cdot)$ with $v(0, \cdot) = u(\cdot)$, and the associated flow $\chi(\varepsilon, \cdot)$, taking the derivative of

$$J(\varepsilon) := J(\chi(\varepsilon, \Omega^1), \dots, \chi(\varepsilon, \Omega^M))$$

at $\varepsilon = 0$ (with a straightforward extension of the previous discussion to functions depending on several sets). Using a change of variables, equation (2.8) becomes

$$J(\varepsilon) = \sum_{n=1}^M \int_{\chi(\varepsilon, \Omega^n)} f_n(\chi(\varepsilon, y)) |D\chi(\varepsilon, y)| dy \quad (2.9)$$

To compute the derivative, we use $(d/d\varepsilon) |D\chi(\varepsilon, y)| = \operatorname{div}(v(\varepsilon, \chi(\varepsilon, y))) |D\chi(\varepsilon, y)|$, which is equal to $\operatorname{div}(u)$ at $\varepsilon = 0$, and $\nabla f_n \cdot u + f_n \operatorname{div}(u) = \operatorname{div}(f_n u)$, to obtain

$$dJ(\Omega^1, \dots, \Omega^M)(u) = \frac{dJ(\varepsilon)}{d\varepsilon} \Big|_{\varepsilon=0} = \sum_{n=1}^M \int_{\Omega^n} \operatorname{div}(f_n u) dy \quad (2.10)$$

Using the divergence theorem (assuming that $\Omega^1, \dots, \Omega^M$ are smooth and bounded), an alternate expression for equation (2.10) is

$$dJ(\Omega^1, \dots, \Omega^M)(u) = \sum_{n=1}^M \int_{\partial\Omega^n} f_n u \cdot \nu^n ds \quad (2.11)$$

where ν^n is the outward normal to $\partial\Omega^n$.

From (2.7) and (2.10), the associated Riemannian gradient flow on diffeomorphisms can now be written as

$$\frac{d\varphi(t, \cdot)}{dt} = - \sum_{n=1}^M \int_{\Omega^n} (K(y, \varphi(t, \cdot)) \nabla f_n(y) + f_n(y) \nabla_1 K(y, \varphi(t, \cdot))) dy \quad (2.12)$$

where $\nabla_1 K$ refers to the (usual) gradient of K with respect to its first variable (y). We can equivalently use (2.11) which yields

$$\frac{d\varphi(t, \cdot)}{dt} = - \sum_{n=1}^M \int_{\partial\Omega^n} K(y, \varphi(t, \cdot)) f_n(y) \nu^n ds(y) \quad (2.13)$$

(Equation (2.13) can obviously be deduced from (2.12) using the divergence theorem.)

3. Projected Gradient Descent Flow

We now complete the gradient descent algorithm defined in (2.7) with a projection step that drives the evolution according to the motion of a finite set of control points. This will speed up the process and will control the level of deformation. To this end, assume that a set of

control points $x_1^0, x_2^0, \dots, x_N^0$ has been defined on the boundary of each Ω^n . Introduce a new scalar kernel K_0 and the finite-dimensional vector space

$$\mathcal{W}(t) = \left\{ \sum_{i=1}^N K_0(\cdot, \varphi(t, x_i^0)) \alpha_i(t); \alpha_i(t) \in \mathbb{R}^d \forall i \right\}$$

We will choose K_0 smooth enough to ensure that $\mathcal{W} \subset \mathcal{V}$. The shape gradient will be replaced by its best approximation of an element of $\mathcal{W}(t)$, i.e., its orthogonal projection on $\mathcal{W}(t)$ for the dot product on $\mathcal{V}$.

Denote, for short, $\nabla J(t) := \nabla J(\Omega(t))$ with $\Omega(t) = \varphi(t, \Omega_0)$ for a template set Ω_0 that will be deformed to at the observed image, and $\tilde{\nabla} J(t)$ its projection on $\mathcal{W}(t)$, which takes the form

$$\begin{aligned} \tilde{\nabla} J(t)(\cdot) &= \sum_{i=1}^N K_0(\cdot, \varphi(t, x_i^0)) \alpha_i(t) \alpha_i(t) \in \mathbb{R}^d \\ &= \underbrace{[K_0(\cdot, \varphi(t, x_1^0)) Id_d, \dots, K_0(\cdot, \varphi(t, x_N^0)) Id_d]}_{H_0(\cdot, X(t))} \underbrace{\begin{pmatrix} \alpha_1(t) \\ \vdots \\ \alpha_N(t) \end{pmatrix}}_{\alpha(t)} \\ &= H_0(\cdot, X(t)) \alpha(t) \end{aligned} \quad (3.1)$$

where $X(t) = (\varphi(t, x_1^0), \dots, \varphi(t, x_N^0))$.

Now choose a canonical orthonormal basis $\{a_j\}_{j=1}^{Nd}$ on $\mathbb{R}^{Nd}$. Then denoting $dJ(t) := dJ(\Omega(t))$, the orthogonality projection constraint will require that for all j

$$\begin{aligned} \langle \tilde{\nabla} J(t), H_0(\cdot, X(t)) a_j \rangle_{\mathcal{V}} &= \langle \tilde{\nabla} J(t), H_0(\cdot, X(t)) a_j \rangle_{\mathcal{V}} \\ \text{yielding } dJ(t)(H_0(\cdot, X(t)) a_j) &= \langle H_0(\cdot, X(t)) \alpha(t), H_0(\cdot, X(t)) a_j \rangle_{\mathcal{V}}. \end{aligned}$$

Expressing $\alpha(t) = \sum_{i=1}^{Nd} b_i(t) a_i$ for a set of Nd scalars $b_i(t)$, the last equation can be rewritten as the following system of equations

$$dJ(t)(H_0(\cdot, X(t)) a_j) = \sum_{i=1}^{Nd} b_i(t) \langle H_0(\cdot, X(t)) a_i, H_0(\cdot, X(t)) a_j \rangle_{\mathcal{V}}. \quad (3.2)$$

Solving this system for all $b_i(t)$ yields $\alpha(t)$ (and $\tilde{\nabla} J(t)$ via equation (3.1)) and a projected gradient descent process can be performed.

The right-hand side of equation (3.2) involves a time dependent, symmetric, positive definite matrix $M(t)$ of dimension $Nd \times Nd$

$$M_{ij}(t) = \langle H_0(\cdot, X(t)) a_i, H_0(\cdot, X(t)) a_j \rangle_{\mathcal{V}}.$$

The expression of $M(t)$ as a function of $X(T)$ only depends on the choice made for the kernels K_0 and K . This expression simplifies when $K_0(x, y)$ is chosen as

$$K_0(x, y) = (K \odot \tilde{K})(x, y) = \int_{\Omega} K(x, z) \tilde{K}(z, y) dz \quad (\tilde{K} \text{ another kernel}). \quad (3.3)$$

Then, letting $\mathbf{a}_i^k$ denote the k th d -dimensional block of $\mathbf{a}_i$,

$$\begin{aligned} M_{ij}(t) &= \sum_{k,l}^N \langle K_0(\cdot, \varphi(t, x_k^0)) \mathbf{a}_i^k, K_0(\cdot, \varphi(t, x_l^0)) \mathbf{a}_j^l \rangle_{\mathcal{V}} \\ &= \sum_{k,l}^N \int_{\Omega} \langle K(\cdot, z) \tilde{K}(z, \varphi(t, x_k^0)) \mathbf{a}_i^k, K_0(\cdot, \varphi(t, x_l^0)) \mathbf{a}_j^l \rangle_{\mathcal{V}} dz \\ &= \sum_{k,l}^N \int_{\Omega} (\tilde{K}(z, \varphi(t, x_k^0)) \mathbf{a}_i^k)^T K_0(z, \varphi(t, x_l^0)) dz \mathbf{a}_j^l \\ &= \sum_{k,l}^N \mathbf{a}_i^{k^T} (\tilde{K} \odot K_0)(\varphi(t, x_k^0), \varphi(t, x_l^0)) \mathbf{a}_j^l \end{aligned} \quad (3.4)$$

with

$$(\tilde{K} \odot K_0)(x, y) = \int \tilde{K}(x, z) K_0(z, y) dz.$$

If we choose Gaussian kernels in (3.3), with $K(x, y) = g_{\sigma^2}(x - y)Id_d$ and

$\tilde{K}(x, y) = g_{\sigma_0^2 - \sigma^2}(x - y)Id_d$ ($\sigma_0 > \sigma \Rightarrow \mathcal{W} \subset \mathcal{V}$), we can define the following $N \times N$ matrix $G(t)$

$$G_{kl}(t) = \int_{\Omega} \tilde{K}(\varphi(t, x_k^0), z) K_0(z, \varphi(t, x_l^0)) dz = g_{2\sigma_0^2 - \sigma^2}(\varphi(t, x_k^0) - \varphi(t, x_l^0)).$$

Equation (3.4) for the matrix $M(t)$, then becomes

$$M_{ij}(t) = \sum_{k,l}^N \mathbf{a}_i^{k^T} \mathbf{a}_j^l G_{kl}(t).$$

With the vector $F(t) = (F_1(t), \dots, F_{Nd}(t))^T$ and $F_j(t) := dJ(t)(H_0(\cdot, X(t)) \mathbf{a}_j)$, the system of equations (3.2) can now be reformulated as

$$M(t) \alpha(t) = F(t). \quad (3.5)$$

Remark: If we denote by $\{e_i\}_{i=1}^d$ the canonical basis on $\mathbb{R}^d$ and $\mathbf{0}$ as its zero vector, given the canonical orthonormal basis $\{\mathbf{a}_j\}_{j=1}^{Nd}$ and $\mathbf{a}_j^k \in \mathbb{R}^d$, we have

$$\begin{aligned} \mathbf{a}_j &= \begin{pmatrix} \mathbf{a}_j^1 & \dots & \mathbf{a}_j^{\lfloor \frac{j-1}{d} \rfloor} & \mathbf{a}_j^{\lfloor \frac{j-1}{d} \rfloor + 1} & \mathbf{a}_j^{\lfloor \frac{j-1}{d} \rfloor + 2} & \dots & \mathbf{a}_j^N \end{pmatrix}^T \\ &= \begin{pmatrix} \mathbf{0} & \dots & \mathbf{0} & e_{((j-1) \bmod d) + 1} & \mathbf{0} & \dots & \mathbf{0} \end{pmatrix}^T \end{aligned}$$

Then, for any i there is only one $k = \lfloor \frac{i-1}{d} \rfloor + 1$ such that α_i^k is different than 0, so the expression for $M(t)$ becomes

$$\begin{aligned} M_{ij}(t) &= \left(\alpha_i^{\lfloor \frac{i-1}{d} \rfloor + 1} \right)^T \left(\alpha_j^{\lfloor \frac{j-1}{d} \rfloor + 1} \right) G(t)_{\lfloor \frac{i-1}{d} \rfloor + 1, \lfloor \frac{j-1}{d} \rfloor + 1} \\ &= \delta \left(\lfloor \frac{i-1}{d} \rfloor - \lfloor \frac{j-1}{d} \rfloor \right) G(t)_{\lfloor \frac{i-1}{d} \rfloor + 1, \lfloor \frac{j-1}{d} \rfloor + 1} \end{aligned} \quad (3.6)$$

This symmetric structure allows the calculation of $\alpha(t)$ using the following simplification of the system of equations (3.5)

$$G(t) \cdot \beta_m(t) = h_m(t) \quad \forall m = 1, \dots, d \quad (3.7)$$

with

$$\beta_m(t) = (\alpha_1^m(t), \dots, \alpha_N^m(t))^T, \quad h_m(t) = (F_m(t), F_{m+d}(t), \dots, F_{m+(N-1)d}(t))^T$$

Segmentation Example. We now illustrate the computation of equation (3.2) using the segmentation cost function introduced in (2.8)

$$J(\Omega^1(t), \dots, \Omega^n(t)) = J(t) = \sum_{n=1}^M \int_{\Omega^n(t)} f_n(y) \, dy \quad (3.8)$$

with $\Omega^n(t) := \varphi(t, \Omega_0^n)$. The calculation of this energy is illustrated in Figure 3.1 for the case $M = 1$.

Since the matrix M is problem-independent, we only need to derive the expression of $F(t)$ in the right-hand side of (3.5). For this, let us take $u = H_0(\cdot, \mathbf{X}(t)) \mathbf{a}_j$ in equation (2.10), yielding

$$\begin{aligned} F_j(t) &= \sum_{n=1}^M \sum_{i=1}^N \int_{\Omega^n(t)} \left[\nabla^T f_n(y) K_0(y, \varphi(t, x_i^0)) + f_n(y) \nabla_1^T K_0(y, \varphi(t, x_i^0)) \right] \mathbf{a}_j^i dy \\ &= \sum_{n=1}^M \sum_{i=1}^N \int_{\partial \Omega^n(t)} \left(f_n(y) K_0(y, \varphi(t, x_i^0)) \right) (\nu^n)^T \mathbf{a}_j^i ds(y) \end{aligned} \quad (3.9)$$

(∇_1 takes the gradient with respect to the first variable).

Remark: For simplicity, we have chosen to restrict to area (in two dimensions) or volume (in three dimensions) costs in the segmentation energy. It would be easy to also include costs associated to integrals along the boundary of the region. For example, one can penalize the length (or the area in the three-dimensional case), which induces a term involving the mean curvature in the L^2 gradient (cf. [41], [51]). One then needs to apply the kernel K to this L^2 gradient to obtain the corresponding shape gradient. Notice that, although a length- or area-shortening term is needed for stability in standard versions of active contours, we did not have to include it in our experiments, thanks to the regularizing effect of the diffeomorphic evolution.

4. Numerical Implementation

4.1. Boundary-based discrete gradient

A simple way to implement diffeomorphic active contours is to directly discretize the expression of the gradient, which was defined in §2.2. Here we describe such an implementation based on the expression of the gradient as a boundary integral (equation (2.13)). To simplify, we restrict to the simple case of two-dimensional images, with a region delimited by a single curve ($M = 1$). Letting γ represent the evolving curve, assumed to be parametrized over $[0, 1]$, equation (2.13) yields

$$\frac{d\varphi(t, \cdot)}{dt} = - \int_0^1 K(\gamma(t, \tilde{u}), \varphi(t, \cdot)) f(\gamma(t, \tilde{u})) \hat{\nu}(t, \tilde{u}) d\tilde{u}$$

where $\hat{\nu}$ is the unnormalized normal, obtained by applying a $\frac{\pi}{2}$ rotation to the vector $\dot{\gamma}_u(t, u)$. Since we are only interested in the evolution of the curve, and $\gamma(t, u) = \varphi(t, \gamma(0, u))$, we can restrict this evolution to the active contour, and obtain

$$\frac{d\gamma(t, u)}{dt} = - \int_0^1 K(\gamma(t, \tilde{u}), \gamma(t, u)) f(\gamma(t, \tilde{u})) \hat{\nu}(t, \tilde{u}) d\tilde{u} \quad (4.1)$$

Assume that γ is discretized as a polygon, with vertices $\gamma_1, \dots, \gamma_N$. Then one can define the discrete evolution

$$\frac{d\gamma_k}{dt} = - \sum_{l=1}^N K(\gamma_l, \gamma_k) f(\gamma_l) \hat{\nu}_l$$

where $\hat{\nu}_l = R_{\pi/2}(\gamma_{k+1} - \gamma_k)$, $R_{\pi/2}$ being the $\frac{\pi}{2}$ rotation. In this expression, the evolution is obtained by applying, at each step, the matrix $\Sigma_{kl} = K(\gamma_l, \gamma_k)$ to what would constitute the discrete L^2 gradient. This implementation has a complexity comparable to the one associated with Sobolev active contours [45], which results from the application of a smoothing operator to the L^2 gradient. The difference is that with Sobolev active contours, this smoothing operator is intrinsically defined along the curve, as the inverse of a differential operator acting on the arc length. For diffeomorphic active contours, the smoothing operator results from a two-dimensional kernel, the one that defines the considered group of diffeomorphisms on the plane. It is this two-dimensional structure that allows for global interactions within the curve, slowing down the motion of points that would be on a collision trajectory and that would potentially create self-intersections in the curve.

One also can introduce control points in the discrete framework (cf. Section 3). Being equation (3.5) already finite dimensional with an explicit matrix $M(t)$ (at least if Gaussian kernels are chosen), the only part that needs to be discretized is the computation of the right-hand side ($F(t)$), which, as can be seen from equation (3.9), can be obtained as above, with K_0 replacing K .

The presented approach can be easily extended to several curves and label types and to three dimensions, where polygons have to be replaced by triangulated surfaces.

Finally, we point out the fact that the discretization of the gradient (which replaces integrals by sums) becomes inaccurate when points get too separated. Like most active contours

methods, one needs to resample the curves every time such an event occurs. The implementation that led to Figure 4.1 included such a resampling step.

4.2. Region-based discretization of the energy

We now consider an alternative approach for the numerical implementation, in which one starts with a discretized version of the variational problem before using an adapted optimization algorithm (the previous approach, like most algorithms that have been proposed for active contours, operate by discretizing the gradient descent evolution derived from the continuous problem).

Recall that, at time t of the minimization algorithm, the sets $\Omega^n(t)$ are, by construction, diffeomorphic to their original templates Ω_0^n , i.e., $\Omega^n(t) = \varphi(t, \Omega_0^n)$. At the same time, the energy term (3.8) can be written as

$$J(t) = \sum_{n=1}^M \int_{\varphi(t, \Omega_0^n)} f_n(\varphi(t, x)) |D\varphi(t, x)| dx$$

To discretize this energy, we assume that each template is decomposed in regular volume elements (pixels or voxels), with R_n elements of tissue l_n for every $n \in \{1, \dots, M\}$. Define particles $z_1^{(n)}(t), \dots, z_{R_n}^{(n)}(t)$, with $z_1^{(n)}(t) = \varphi(t, z_1^{(n)}(0))$ initially positioned at the center of each volume element of tissue n . Define also $V_r^{(n)}(t) = |D\varphi(t, z_r^{(n)}(0))|$ as the evolving volume associated to element r of tissue n , at time t . Up to a scale factor, the energy term can then be approximated by

$$J(t) \approx U(t) = \sum_{n=1}^M \sum_{r=1}^{R_n} f_n(z_r^{(n)}(t)) V_r^{(n)}(t). \quad (4.2)$$

This forms our discrete energy function, which, by definition of the z and V variables, is still a function of the evolving diffeomorphism, φ .

Following a procedure similar than when calculating $dJ(\Omega(t)) (H_0(\cdot, \mathbf{X}(t)) \mathbf{a}_j)$ in equation (3.9), if we denote $B_j(t) = dU(t) (H_0(\cdot, \mathbf{X}(t)) \mathbf{a}_j)$ we obtain

$$B_j(t) := \sum_{n=1}^M \sum_{i=1}^N \sum_{r=1}^{R_n} \left[\nabla^T f_n(z_r^{(n)}(t)) K_0(z_r^{(n)}(t), x_i(t)) + f_n(z_r^{(n)}(t)) \nabla_1 K_0(z_r^{(n)}(t), x_i(t)) \right] \mathbf{a}_j^i \cdot V_r^{(n)}(t) \quad (4.3)$$

The calculation of this last term requires the knowledge of the evolution of the center of each voxel $z_r^{(n)}(t)$ and volume $V_r^{(n)}(t)$, for all M tissues. Their evolution, in combination with the evolution of the control points $x_j(t)$ is detailed as follows:

$\mathbf{x}_j(t)$: The N control points $x_1^0, \dots, x_N^0$ will be defined at time 0, typically on the boundary of each Ω_0^n . As it can be derived from previous sections, their position in time can be expressed in terms of the action of diffeomorphisms on their initial position. If we denote $x_j(t) = \varphi(t, x_j^0)$ and if we let v denote the velocity of the diffeomorphic evolution, we get

$$\begin{aligned}\frac{dx_j(t)}{dt} &= \frac{d\varphi(t, x_j^0)}{dt} = v(t, \varphi(t, x_j^0)) \\ &= v(t, x_j(t))\end{aligned}$$

$z_r^{(n)}(t)$: The evolution of the centers of each voxel follows a similar equation. Denoting $z_r^{(n)}(t) = \varphi(t, z_r^{(n)}(0))$ we get

$$\begin{aligned}\frac{dz_r^{(n)}(t)}{dt} &= \frac{d\varphi(t, z_r^{(n)}(0))}{dt} = v(t, \varphi(t, z_r^{(n)}(0))) \\ &= v(t, z_r^{(n)}(t))\end{aligned}$$

$V_r^{(n)}(t)$: In a similar procedure, the evolution of the volumes becomes also a function of $\varphi(t, \cdot)$

$$\begin{aligned}\frac{dV_r^{(n)}(t)}{dt} &= \frac{d|D\varphi(t, z_r^{(n)}(0))|}{dt} \\ &= |D\varphi(t, z_r^{(n)}(0))| \cdot \operatorname{div}(v(t, \varphi(t, z_r^{(n)}(0)))) \\ &= V_r^{(n)}(t) \cdot \operatorname{div}(v(t, z_r^{(n)}(t)))\end{aligned}$$

Denoting the projected gradient of $U(t)$ by $\tilde{\nabla} U(t) = \sum_{i=1}^N K_0(\cdot, x_i(t)) \alpha_i(t) = v(t, \cdot)$, the last three equations become

$$\frac{dx_j(t)}{dt} = \sum_{i=1}^N K_0(x_j(t), x_i(t)) \alpha_i(t) \quad (4.4)$$

$$\frac{dz_r^{(n)}(t)}{dt} = \sum_{i=1}^N K_0(z_r^{(n)}(t), x_i(t)) \alpha_i(t) \quad (4.5)$$

$$\frac{dV_r^{(n)}(t)}{dt} = V_r^{(n)}(t) \cdot \sum_{i=1}^N \nabla_1^T K_0(z_r^{(n)}(t), x_i(t)) \alpha_i(t) \quad (4.6)$$

The coefficients $\alpha_1(t), \dots, \alpha_N(t)$ are defined by solving

$$M(t) \alpha(t) = B(t) \quad (4.7)$$

(cf. (3.5)), where B is defined in (4.3).

In the experiments shown in §4.5, the evolution of the images was computed running Algorithm 1, using the Euler method for time integration together with a line search for the selection of the time step Δt .

It is worth noticing that we chose as a stopping time for Algorithm 1 the occurrence of a very small norm for the projected gradient $\tilde{\nabla}U(t)$. This norm is calculated as follows

$$\begin{aligned}\|\tilde{\nabla}U(t)\|_{\mathcal{V}}^2 &= \langle H_0(\cdot, \mathbf{X}(t))\alpha(t), H_0(\cdot, \mathbf{X}(t))\alpha(t) \rangle_{\mathcal{V}} \\ &= \sum_{i,j}^N \int_{\Omega} \langle K(\cdot, z) \tilde{K}(z, x_i(t)) \alpha_i(t), K_0(\cdot, x_j(t)) \alpha_j(t) \rangle_{\mathcal{V}} dz \\ &= \sum_{i,j}^N \alpha_i^T(t) \int_{\Omega} \tilde{K}^T(z, x_i(t)) K_0(z, x_j(t)) dz \alpha_j(t) \\ &= \alpha^T(t) M(t) \alpha(t)\end{aligned}$$

Algorithm 1

PROJECTED GRADIENT DESCENT

Input: $\sigma, \sigma_0, \Omega_0^n, f_n, \{x_i^0\}_{j=1}^N$ (N Control points), and tolerance ε for the norm of $\tilde{\nabla}U(t)$

Output: $\varphi(t, \Omega_0^n)$ for each $n = 1, \dots, M$ (Deformed templates)

Set $k = 1$

- 1: **While** ($\|\tilde{\nabla}U(t)\|_{\mathcal{V}} > \varepsilon$) **do**
- 2: Calculate vector $B^{(k)}$ using equation (4.3)
- 3: Calculate the matrix $M^{(k)}$ using equation (3.6)
- 4: Obtain the vector $\alpha^{(k)}$ and then the gradient $\tilde{\nabla}U(t)$ solving the linear system (4.7)
- 5: For $j = 1, \dots, N$, flow the control points following

$$x_j(k+1) = x_j(k) + \Delta t \left(\sum_{i=1}^N K_0(x_j(k), x_i(k)) \alpha_i^{(k)} \right) \quad (\text{Eq. (4.4)})$$

- 6: For every tissue n and for every $r = 1, \dots, R_n$, flow the voxels centers following

$$z_r^{(n)}(k+1) = z_r^{(n)}(k) + \Delta t \left(\sum_{i=1}^N K_0(z_r^{(n)}(k), x_i(k)) \alpha_i^{(k)} \right) \quad (\text{Eq. (4.5)})$$

- 7: For every tissue n and for every $r = 1, \dots, R_n$, evolve the voxel volumes following

$$V_r^{(n)}(k+1) = V_r^{(n)}(k) + \Delta t \left(V_r^{(n)}(k) \cdot \sum_{i=1}^N \nabla_1^T K_0(z_r^{(n)}(k), x_i(k)) \alpha_i^{(k)} \right) \quad (\text{Eq. (4.6)})$$

- 8: $k = k + 1$

- 9: **End While**

- 10: Save the last position of every voxel center.

4.3. Re-gridding

Algorithm 1 is an exact gradient descent algorithm for the minimization of the energy term $U(t)$ (equation (4.2)) over diffeomorphisms. It is not difficult to see that $U(t)$ becomes a poor

approximation of the initial functional $J(t)$ when the volume $V_r^{(n)}(t)$ exceeds acceptable limits, say V_{min} and V_{max} . When one of these thresholds is reached, we implemented a re-gridding algorithm that defines a new lattice over the deformed template. The procedure is explained in Algorithm 2, and depicted in Figure 4.2.

Algorithm 2

RE-GRIDDING

Input: Position of the center of all voxels, deformed templates $\varphi(t, \Omega_0^n)$, V_{min} , V_{max} and $V_r^{(n)} \forall r, n$

Output: New Ω_0^n for all $n = 1, \dots, M$

- 1: Start when, for some r and k , $V_r^{(n)}(k) \notin [V_{min}, V_{max}]$
 - 2: Retrieve the maximum cube where all the tissues are located at step k
 - 3: Subdivide the volume surrounding these tissues in voxels of size 1, and define each new center as $z_i^{(n)}(0)$
 - 4: Redefine the tissue at $z_j^{(n)}(0)$ as $I(\varphi^{-1}(t, z_i^{(n)}(0)))$
 - 5: Output the new grid and tissues labels
-

4.4. Parameter Selection

The selection of a suitable set of parameters in the segmentation algorithm can considerably improve the results. We now briefly describe how these values were selected:

σ : Its magnitude is mainly associated with the amount of deformation allowed on each original template Ω_0^n . In general, a value for σ smaller than the smallest juncture in the target tissue represents a space $\mathcal{V}$ and a family of diffeomorphisms $\mathcal{G}_{\mathcal{V}}$ that encourages a strong deformation, like a fluid with low viscosity.

σ_0 : Closely tied with the selection of σ , a value of $\sigma_0 \gg \sigma$ will promote large deformation of the original templates, making the evolution robust to noise and artifacts.

Ω_0^n : The initial binary templates are based on shapes with structures anatomically close to the ones in the target image. In this study we assumed pre-processed templates that match the pose of each target image. The user then defines an initial position for each template and the algorithm deforms them accordingly.

$\{x_i^0\}_{i=1}^N$: The selection of a large number of control points on the boundary of each template region, besides slowing down the segmentation process will only improve the accuracy when their initial position is regularly placed over the surface. In general, the more control points are put close to places where a bigger final deformation occurs, the better the segmentation results are when the measure of the likelihood of voxels belonging to the different tissues $f_n(\cdot)$ is accurate. This fact encourages the use of the model described in §4.1, where a redefinition of the control points is regularly performed.

4.5. Results

4.5.1. Two-dimensional segmentation—As a first example we show the segmentation of a two-dimensional dumbbell in Figure 4.3. The size of the target image is 124×252 pixels with a 20% level of Gaussian noise over a binary original image, resulting in a noisy function f_n over the defined domain. The template is a binary disc of diameter 30 pixels, with 30 control points regularly positioned on its boundary. In the figure we show the path taken by every control point. The algorithm took 200 iterations with one re-gridding performed after 120 iterations, due mainly to the big enlargement of the original disc. In Figure 4.4 we show the results of a segmentation problem posed by Sundaramoorthi et al. [45], using the same Gaussian noise on the function f_n as in the last example. In general, we can say that the obtained results are similar to the ones using Sobolev active contours in two dimensions, though in our case the evolution took much less time to converge, and all the fingers grew simultaneously.

As it has been shown [21,39], myocardial wall identification has become vital for calculating ejection-fraction ratio, heart output, ventricular volume ratio and muscle wall thickness, contributing to the identification of several medical pathologies. Then, as an example of a segmentation of this type of biological tissue, we show results of the segmentation of the left ventricle in human hearts in two and three dimensions. The likelihood function $f_n(\cdot)$ was obtained using the Alternate Kernel Mixture method (AKM) [36], that basically estimates the probability of each point on the target configuration to belong to a specific class of tissue, i.e., $p(I_n|x)$. This robust classification method is based on the intensity levels of the original MRI and models each tissue as a semi-parametric mixture of Gaussian distributions [26]. The segmentation of a simple two-dimensional image is presented in Figure 4.5, with the corresponding value of the probability $p(I_1|x)$ shown in gray intensities. The initial two-dimensional template used 34 control points on its boundary and corresponds to a slice at the same height (long axis) of a healthy heart at end systole (ES). The target image is a sick heart at end diastole (ED) that has an implanted pacemaker.

4.5.2. Three-dimensional segmentation—We implemented the segmentation procedure in three dimensions, using basically the same structure of the algorithm for two dimensions. In Figure 4.6, an original binary three-dimensional dumbbell shape of size $62 \times 62 \times 126$ voxels is corrupted with 20% of Gaussian noise, giving qualitatively the same function f_n as in the synthetic examples in two dimensions. The segmentation starts from a small sphere of radius 10 voxels with 100 control points on its boundary. The method shows a stable behavior near the edges. The small values for $\sigma = 7$ and $\sigma_0 = 14$ are due to the strong deformation required on the small sphere, and the need to at the neck on the final dumbbell. An example of three-dimensional segmentation of the left ventricle of a sick human heart at ED is shown in Figure 4.7. The template is a healthy left ventricle of a human heart of size $100 \times 86 \times 74$ at ES, and the target is an MRI of a heart with a pacemaker, of size $103 \times 142 \times 102$ voxels at ED that was originally masked at the right of the image to avoid including muscle tissue. The segmentation of the pacemaker, noise and papillary muscle is avoided by the robustness of the model that required just 100 control points in the surface boundary of the initial binary template. Notice in particular how the shape of the initial template is preserved at the end of the algorithm, mainly due to the selection of $\sigma = 15$ and $\sigma_0 = 30$.

5. Conclusions and Future Work

In this study we proposed a diffeomorphic active contour solution to the segmentation of medical images in which anatomies are presented as deformable templates being transformed by the action of diffeomorphisms. The procedure minimized a proposed segmentation energy based on a measure of the likelihood of every voxel to belong to a

certain tissue, obtained using the AKM method on the intensities of the image. In particular, the definition of a finite set of control points on the boundary of the templates yielded a projected gradient descent on the space *Diff* that allowed a topologically invariant deformation of the initial structures, as well as a robust anatomical correlation between the template and the target. The method was applied to several two- and three-dimensional synthetic examples, as well as to images containing the left ventricle of human hearts.

Due to the common presence of weak boundaries between similar tissue but belonging to different structures, in the future we propose an improvement on the current model with a redefinition of the energy term to take into consideration prior knowledge of the level of deformation allowed, and the optimal position of the set of control points.

REFERENCES

1. Alexandrov O, Santosa F. A topology-preserving level set method for shape optimization. *J. Comput. Phys.* 2005; 204:121–130.
2. Arnold VI. Sur un principe variationnel pour les écoulements stationnaires des liquides parfaits et ses applications aux problèmes de stabilité non linéaires. *J. Mécanique.* 1966; 5:29–43.
3. Aronszajn N. Theory of reproducing kernels. *T. Am. Math. Soc.* 1950; 68:337–404.
4. Caselles V, Catté F, Coll T, Dibos F. A geometric model for active contours in image processing. *Numer. Math.* 1993; 66:1–31.
5. Caselles V, Kimmel R, Sapiro G. Geodesic active contours. *Int. J. Comput. Vision.* 1997; 22:61–79.
6. Caselles V, Kimmel R, Sapiro G, Sbert C. Minimal surfaces based object segmentation. *IEEE T. Pattern Anal.* 1997; 19:394–398.
7. Chan TF, Vese LA. Active contours without edges. *IEEE T. Image Process.* 2001; 10:266–277.
8. Chen Y, Tagare H, Thiruvankadam S, Huang F, Wilson D, Gopinath K, Briggs R, Geiser E. Using prior shapes in geometric active contours in a variational framework. *Int. J. Comput. Vision.* 2002; 50:315–328.
9. Christensen GE, Rabbitt RD, Miller MI. Deformable templates using large deformation kinematics. *IEEE T. Image Process.* 1996; 5:1435–1447.
10. Cremers, D.; Fluck, O.; Rousson, M.; Aharon, S. A probabilistic level set formulation for interactive organ segmentation. In: Pluim, Josien PW.; Reinhardt, Joseph M., editors. *Medical Imaging 2007*. Vol. vol. 6512. SPIE; 2007. p. 65120V
11. Cremers D, Tischhäuser F, Weickert J, Schnörr C. Diffusion snakes: Introducing statistical shape knowledge into the Mumford-Shah functional. *Int. J. Comput. Vision.* 2002; 50:295–313.
12. Delfour, MC. Shape Optimization and Free Boundaries (Montreal, PQ), M. C. Delfour and G. Sabidussi, eds., vol. 380 of NATO Adv. Sci. I. C-Math. Kluwer Acad. Publ.; 1992. Shape derivatives and differentiability of min max; p. 36-112.
13. Delfour, MC.; Sabidussi, G. Shape Optimization and Free Boundaries (Montreal, PQ), M. C. Delfour and G. Sabidussi, eds., vol. 380 of NATO Adv. Sci. I. C-Math. Kluwer Acad. Publ.; 1992. Shape optimisation and free boundaries; p. 397-457.
14. Delfour MC, Zolésio J-P. Structure of shape derivatives for nonsmooth domains. *J. Funct. Anal.* 1992; 104:1–33.
15. Delfour MC, Zolésio J-P. Shapes and Geometries. Analysis, Differential Calculus and Optimization. SIAM. 2001
16. Dupuis P, Grenander U, Miller M. Variational problems on flows of diffeomorphisms for image matching. *Q. Appl. Math.* 1998; 56:587–600.
17. Grenander U. Lectures in Pattern Theory I. Applied Mathematical Sciences. 1976; vol. 18
18. Grenander, U.; Chow, Y.; Keenan, DM. Hands: A Pattern Theoretic Study of Biological Shapes., vol. 2 of Research notes in Neural Computing. New York: Springer-Verlag; 1990.
19. Grenander U, Miller MI. Computational anatomy: An emerging discipline. *Q. Appl. Math.*, LVI. 1998:617–694.

20. Gu L, Peters T. 3D segmentation of medical images using a fast multistage hybrid algorithm. *IJCARS*. 2006; 1:23–31.
21. Gupta A, von Kurowski L, Singh A, Geiger D, Liang C-C, Chiu M-Y, Adler LP, Haacke M, Wilson DL. Cardiac MR image segmentation using deformable models. *Comput. Cardiol*. 1993. 1993 Sep.;747–750.
22. Joshi S, Miller M. Landmark matching via large deformation diffeomorphisms. *IEEE T. Image Process*. 2000; 9:1357–1370.
23. Kass M, Witkin A, Terzopoulos D. Snakes: active contour models. *Int. J. Comput. Vision*. 1988; 1:321–331.
24. Kichenassamy S, Kumar A, Olver P, Tannenbaum A, Yezzi A. Gradient flows and geometric active contour models. *Proc. IEEE I. Conf. Comp. Vision*. 1995:810–815.
25. Le Guyader C, Vese L. Self-repelling snakes for topology-preserving segmentation models. *IEEE T. Image Process*. 2008; 17:767–779.
26. Lee NA, Priebe CE, Miller MI, Ratnanather JT. Validation of Alternating Kernel Mixture method: application to tissue segmentation of cortical and subcortical structures. *J. Biomed. Biotechnol*. 2008; 2008:346129. [PubMed: 18695738]
27. Malladi R, Sethian JA, Vemuri BC. Shape modeling with front propagation: a level set approach. *IEEE T. Pattern Anal*. 1995; 17:158–175.
28. Michailovich O, Rath Y, Tannenbaum A. Image segmentation using active contours driven by the Bhattacharyya gradient flow. *IEEE T. Image Process*. 2007; 16:2787–2801.
29. Michor PW, Mumford D. Riemannian geometries on spaces of plane curves. *J. Eur. Math. Soc*. 2006; 8:1–48.
30. Miller MI. Computational anatomy: shape, growth, and atrophy comparison via diffeomorphisms. *NeuroImage*. 2004; 23:19.
31. Miller MI, Priebe C, Qiu A, Fischl B, Kolasny A, Brown T, Park Y, Ratnanather JT, Busa E, Jovicich J, Dickerson B, Yu P, Buckner RL. the Morphometry BIRN. Collaborative computational anatomy: The perfect storm for MRI morphometry study of the human brain via diffeomorphic metric mapping. *Human Brain Mapping*. 2009; 30:2132–2141. [PubMed: 18781592]
32. Miller MI, Trounev A, Younes L. On the metrics and Euler-Lagrange equations of computational anatomy. *Annu. Rev. Biomed. Eng*. 2002; 4:375–405. [PubMed: 12117763]
33. Miller MI, Younes L. Group action, diffeomorphism and matching: a general framework. *Int. J. Comput. Vision*. 2001; 41:61–84.
34. Osher S, Sethian JA. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *J. Comput. Phys*. 1988; 79:12.
35. Paragios N. A level set approach for shape-driven segmentation and tracking of the left ventricle. *IEEE T. Med. Imaging*. 2003; 22:773.
36. Priebe CE, Marchette DJ. Alternating kernel and mixture density estimates. *Computational statistics and data analysis*. 2000; 35:43.
37. Qiu A, Younes L, Miller MI, Csernansky JG. Parallel transport in diffeomorphisms distinguishes the time-dependent pattern of hippocampal surface deformation due to healthy aging and the dementia of the Alzheimer's type. *NeuroImage*. 2008; 40:68. [PubMed: 18249009]
38. Qiu A, Younes L, Miller MI. Intrinsic and extrinsic analysis in computational anatomy. *Neuroimage*. 2008; 39:1804–1814.
39. Ranganath S. Contour extraction from cardiac MRI studies using snakes. *IEEE T. Med. Imaging*. 1995; 14:328–338.
40. Rousson M, Cremers D. Efficient kernel density estimation of shape and intensity priors for level set segmentation. *MICCAI 2005*, vol. 3750 of LNCS. 2005:757–764.
41. Sapiro, G. Geometric Partial Differential Equations and Image Analysis. Cambridge University Press; 2001.
42. Schwartz T, Heimann T, Wolf I, Meinzer HP. 3D heart segmentation and volumetry using deformable shape models. *Comput. Cardiol*. 2007; 34:741–744.
43. Sokołowski, J.; Zolésio, J-P. Introduction to shape optimisation: Shape Sensitivity Analysis. Springer-Verlag; 1992.

44. Sundaramoorthi G, Yezzi A. Global regularizing flows with topology preservation for active contours and polygons. *IEEE T. Image Process.* 2007; 16:803.
45. Sundaramoorthi G, Yezzi A, Mennucci A. Sobolev active contours. *Int. J. Comput. Vision.* 2007; 73:345–366.
46. Trouvé A. Infinite dimensional group action and pattern recognition; comptes rendus de l'académie des sciences. *Comptes rendus de l'Académie des sciences. Série I, Mathématique.* 1995; 321:1031.
47. Trouvé A. Diffeomorphism groups and pattern matching in image analysis. *Int. J. Comput. Vision.* 1998; 28:213–221.
48. Wang L, Beg MF, Ratnanather JT, Ceritoglu C, Younes L, Morris J, Csernansky J, Miller MI. Large deformation diffeomorphism and momentum based hippocampal shape discrimination in dementia of the Alzheimer type. *IEEE T. Med. Imaging.* 2007; 26:462–470.
49. Yezzi A, Kichenassamy S, Kumar A, Olver PJ, Tannebaum A. A geometric snake model for segmentation of medical imagery; *ieee transactions on medical imaging.* *IEEE T. Med. Imaging.* 1997; 16:199.
50. Yezzi, Anthony; Mennucci, Andrea. Conformal metrics and true "gradient flows" for curves; *IEEE I. Conf. Comp. Vis;* Los Alamitos, CA, USA: IEEE Computer Society; 2005. p. 913-919.
51. Younes, L. *Shapes and Diffeomorphisms.* Springer; 2010.
52. Younes L, Arrate F, Miller MI. Evolutions equations in computational anatomy. *NeuroImage.* 2009; 45:S40–S50. [PubMed: 19059343]
53. Zhu SC, Yuille A. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE T. Pattern Anal.* 1996; 18:884–900.

a) *Fixed Set Ω_0* b) *Optimum Set* **FIG. 3.1.**

Calculation of the segmentation energy term (3.8) when $M = 1$, for a given time t : The template Ω_0 is shown in a). At time t of the deformation process, the deformed set $\Omega(t) = \varphi(t, \Omega_0)$ is shown in c), superimposed over the optimum set in b) that encloses the highest probability of a point to belong to tissue 1. As it is seen, the evolved point $\varphi(t, q_2)$ falls inside the exact deformed set, helping to decrease the final energy term, but $\varphi(t, q_1)$ falls outside, increasing the energy so more deformation of the template Ω_0 is necessary.

FIG. 4.1. Diffeomorphic and Sobolev active contours applied to the segmentation of a narrow region. First Row: original image with initial contour (left). The same initial contour was used in all experiments. The last two images in the first row illustrate Sobolev active contours. While evolving the initial circle wraps itself around the contour (center) in a way that will progressively lead to self-intersection (right). With diffeomorphic active contours (second row), the evolution begins in a similar way (left), but no self-intersection is observed, leading to the image on the right. The third row shows the evolution of diffeomorphic active contours with control points. The deformation of the curve is similar, but significantly faster.

FIG. 4.2.

Re-gridding Process: At time $t = T$ one of the volumes surpasses the limits $[V_{min}, V_{max}]$ in *a*). Then, a new grid that encloses the deformed template is defined in *b*). After this process, the center of every voxel in the new lattice is flowed back to the original template in *c*), to be labeled accordingly in the new template, as is seen in *d*).

FIG. 4.3.

Dumbbell segmentation in two dimensions. The target is a binary image, corrupted by 20% of Gaussian noise ($\sigma = 10$, $\sigma_0 = 20$).

FIG. 4.4. Synthetic example for two-dimensional segmentation. The target is a binary image corrupted by 20% of Gaussian noise ($\sigma = 5$, $\sigma_0 = 10$).

FIG. 4.5.

Heart segmentation in two dimensions. The target is an MRI of size 142×102 pixels. Note the presence of a pacemaker, that corrupts the right section of the left ventricle wall. ($\sigma = 20$, $\sigma_0 = 30$).

FIG. 4.6.

Dumbbell segmentation in three dimensions. The target is a binary image of size $62 \times 62 \times 126$ voxels, corrupted by 20% of Gaussian noise. ($\sigma = 7$, $\sigma_0 = 14$).

FIG. 4.7.

Heart segmentation in three dimensions. The target image is an MRI of size $103 \times 142 \times 102$ voxels that was initially masked at the right of the image to avoid including muscle tissue. In the 'Initial stage' we show slices of the original MRI image, where in the other two the image has been replaced by the probability $p(l_n|x)$. The small squares with white center following the boundary of the segmentation are there just to help visualize the evolution. Note the presence of a pacemaker and papillary muscle that are not segmented by the algorithm. As input values we used $\sigma = 15$ and $\sigma_0 = 30$.