Non-Euclidean Image-Adaptive Radial Basis Functions for 3D Interactive Segmentation

Benoît Mory, Roberto Ardon, Anthony Yezzi, Jean-Philippe Thiran

Non-Euclidean Image-Adaptive Radial Basis Functions for 3D Interactive Segmentation

Benoit Mory Roberto Ardon
Medisys Research Laboratory, Philips Healthcare, 33 rue de Verdun, B.P. 313, F-92156 Suresnes Cedex, France
{benoit.mory,roberto.ardon}@philips.com

Anthony J. Yezzi
Georgia Institute of Technology
777 Atlantic Drive NW Atlanta, United States
anthony.yezzi@ece.gatech.edu

Jean-Philippe Thiran
Ecole Polytechnique Fdrale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
jp.thiran@epfl.ch

Abstract

In the context of variational image segmentation, we propose a new finite-dimensional implicit surface representation. The key idea is to span a subset of implicit functions with linear combinations of spatially-localized kernels that follow image features. This is achieved by replacing the Euclidean distance in conventional Radial Basis Functions with non-Euclidean, image-dependent distances. For the minimization of an objective region-based criterion, this representation yields more accurate results with fewer control points than its Euclidean counterpart. If the user positions these control points, the non-Euclidean distance enables to further specify our localized kernels for a target object in the image. Moreover, an intuitive control of the result of the segmentation is obtained by casting inside/outside labels as linear inequality constraints. Finally, we discuss several algorithmic aspects needed for a responsive interactive workflow. We have applied this framework to 3D medical imaging and built a real-time prototype with which the segmentation of whole organs is only a few clicks away.

1. Introduction

In the hope of reproducing human skills to localize and identify specific objects in images, many research efforts have been focused on the development of fully automatic segmentation algorithms. This goal can sometimes be reached, for instance when the shape of the object is known and modeled, by exploiting far more information than the image alone provides. Unfortunately, in many cases, such prior information is not available and the user has to be involved in the segmentation process. In medical imaging for instance, the knowledge of an expert practitioner is often irreplaceable. To be accepted in daily medical practice, in particular with volumetric data, general-purpose segmentation tools should not only be interactive but also provide intuitive, robust and responsive control.

In the past few years, very efficient interactive segmentation algorithms have been proposed [1, 6, 17]. User interactions are typically handled in 2D by drawing so-called "scribbles", associating inside/outside labels to parts of the image. In this work, we present an interactive segmentation framework based on the selection of only a few points, inside or outside the object of interest. Our technique relies on the minimization of an objective criterion over a sensible, low-dimensional subset of possible implicit functions. This subset is spanned by a novel class of non-Euclidean *Radial Basis Functions*, built from image-dependent metrics using local image features, such as intensity distributions or edge information. To the best of our knowledge, this is the first introduction of image-dependent non-Euclidean distances into Radial Basis Functions. For segmentation with implicit surfaces, spatially-localized image-adaptive kernels achieve better accuracy with far fewer basis elements, as soon as they are properly positioned. Consequently, we propose an interactive scenario in which the dimension of the basis, hence the complexity of the optimization problem, increases progressively as the user introduces control points, depending on the difficulty of the segmentation task. Moreover, the association of inside/outside labels to control points is formulated through additional linear inequality constraints. Finally, as the objective criterion may have local minima, we introduce an auxiliary quadratic programming problem that, solved in linear time, allows the user to guide the process toward a local minimum of his or her choice. Minimizing over a restricted low-dimensional space bears some similarities with the recent *GeoS* algorithm [6], in which optimization is performed over a two-dimensional space built from two *geodesic* morphological operators.

2009 IEEE 12th International Conference on Computer Vision (ICCV)
978-1-4244-4419-9/09/$25.00 ©2009 IEEE

In section 2, we recall the implicit representation framework and the variational principles involved in the optimization of a region-based criterion. Recent formulations using Radial Basis Functions are also discussed. In section 3, we introduce an extension with non-Euclidean distances in order to build an image-adaptive basis of implicit functions. In section 4, we cast user interactions as linear inequality constraints. Finally, we describe in section 5 the complete sequential workflow of a real-time 3D interactive tool.

2. Segmentation with Radial Basis Functions

A variational formulation for partitioning a d-dimensional image $I : \Omega \subset \mathbb{R}^d \mapsto \mathbb{R}$ into two disjoint homogeneous regions $\mathcal{A} \subset \Omega$ and $\Omega \backslash \mathcal{A}$ typically involves the choice of two application-specific components: a pixel-wise qualitative measure of the inhomogeneity for each region (r_1, r_2), and a space of admissible partitions $\mathcal{P}$, with an associated regularity constraint $\mathcal{R} : \mathcal{P} \mapsto \mathbb{R}$. Formally, the minimization problem reads:

$$\min_{\mathcal{A} \in \mathcal{P}} E(\mathcal{A}) = \mathcal{R}(\mathcal{A}) + \int_{\mathbf{x} \in \mathcal{A}} r_1(I(\mathbf{x})) + \int_{\mathbf{x} \in \Omega \backslash \mathcal{A}} r_2(I(\mathbf{x})) \quad (1)$$

Our main focus is the design of a partition space ($\mathcal{P}$) adapted to interactive segmentation, even though meaningful homogeneity models are also crucial. Consequently, we will not emphasize the choice of r_i and consider, as an example, a maximum-likelihood criterion [22]:

$$r_i(I(\mathbf{x})) = -\log p_i(I(\mathbf{x})) \quad (2)$$

where p_1 and p_2 are non-parametric intensity distributions [10, 14] in $\mathcal{A}$ and $\Omega \backslash \mathcal{A}$, either known in advance or estimated during the minimization process.

Numerous spaces of admissible partitions $\mathcal{P}$ have been studied in the literature, including discrete graphs [3] as well as explicit [11] or implicit [4] boundary representations. In the latter case, $\mathcal{P}$ is the set of every partition $\mathcal{A}$ defined as the zero super-level set of a real function Φ, $\mathcal{A} = \{x \in \Omega, \Phi(x) \geq 0\}$. (1) can be formulated as a minimization over a functional set $\mathcal{F}$:

$$\min_{\Phi \in \mathcal{F}} E(\Phi) = \mathcal{R}(\Phi) + \int_{\mathbf{x} \in \Omega} H(\Phi) r_1 + \int_{\mathbf{x} \in \Omega} (1 - H(\Phi)) r_2 \quad (3)$$

where H is the Heaviside step function. In this continuous setting, $\mathcal{F}$ is generally defined as an infinite-dimensional space such as Lipschitz functions. But relatively early in the history of implicit models, the idea of building $\mathcal{F}$ as a space spanned by a finite basis was proposed [19], using for instance hyperquadrics [9]. Recently, this approach has regained interest and authors have proposed to generate $\mathcal{F}$ with B-Splines [2] or Radial Basis Functions [8, 18].

Widely used for scattered data interpolation and surface reconstruction [21], Radial Basis Functions offer built-in smoothness and do not require a regular sampling of the image domain Ω. The implicit function Φ is built up as a linear combination of translated and scaled versions of a radially-symmetric non-negative kernel φ centered around N points $\mathbf{x_i}$ (see Fig.1):

$$\Phi_{\boldsymbol{\rho}}(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i \varphi \left(\frac{\|\mathbf{x} - \mathbf{x}_i\|}{\sigma_i} \right) = \sum_{i=1}^{N} \lambda_i \varphi_i(\mathbf{x}) \quad (4)$$

where the scalar weights λ_i, positions $\mathbf{x_i}$ and scales σ_i constitute the discrete set of parameters $\boldsymbol{\rho} = \{\lambda_i, \mathbf{x}_i, \sigma_i\}$. The *functional* E in (3) becomes a *function* of $\boldsymbol{\rho}$:

$$E(\boldsymbol{\rho}) = \int_{\Omega} H(\Phi_{\boldsymbol{\rho}}) r_1(I) + \int_{\Omega} (1 - H(\Phi_{\boldsymbol{\rho}})) r_2(I) \quad (5)$$

where the regularization term $\mathcal{R}$ is usually omitted since the basis functions are intrinsically smooth. Obviating regularization and minimizing E over a finite set are the key advantages of this parametric formulation. Compared to infinite-dimensional level-set techniques, this low-dimensional representation yields more efficient optimization algorithms while keeping topological flexibility in 2D and 3D.

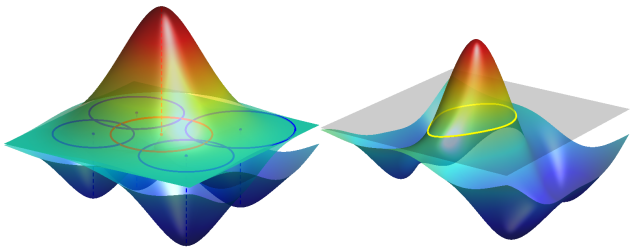


Figure 1. An implicit contour with Euclidean RBFs. On the left, few basis functions with positive (red) and negative (blue) weights. On the right, an implicit function obtained by linear blending.

Along these lines, Gelas et.al [8] have successfully used compactly-supported radial functions for segmentation, optimizing the weights λ_i for given positions $\mathbf{x}_i$ and scales σ_i. However, as pointed out in the context of surface reconstruction [7], radially-symmetric kernels fail to model the asymmetric nature of sharp features such as straight edges. For image segmentation applications, this tends to compromise the low-dimensionality advantage, as the number of basis functions rapidly grows to accurately recover high curvature objects. To overcome this limitation, Slabaugh et.al [18] have proposed to use *anisotropic* Gaussian kernels and optimize their orientation as well as their weight, position and scales. Their 2D experiments show a better capture of image details at the price of increasing the dimensionality of the optimization space (6N parameters).

3. Switching to non-Euclidean Kernels

In order to obtain more accurate segmentation results without increasing the number of parameters, we believe that the basis functions should be designed according to the actual image content. This is a major difference with previous works in which the kernels were purely geometric (spherical or elliptical). The basic idea is to span the space of implicit functions Φ by a richer basis in which each element φ_i is localized in space and also incorporates meaningful image information for segmentation. This can be obtained in the RBF framework by a modification inspired by front propagation theory [12]. By construction, Radial Basis Functions only depend on the *distance* to their center. Their spherical shape is only a consequence of the choice of the *Euclidean* distance. Instead, we propose to switch to an image-dependent *non-Euclidean* distance to build the kernels. This extension opens up the design of basis functions with iso-levels that are no longer spherical and naturally follow the image features (see Fig.2). Each φ_i becomes:

$$\varphi_i(\mathbf{x}) = \varphi\left(\frac{\|\mathbf{x}-\mathbf{x}_i\|_{g_i}}{\sigma_i}\right). \tag{6}$$

To each control point $\mathbf{x}_i$ can correspond a different metric function $g_i : \Omega \mapsto \mathbb{R}$, required to be strictly positive and smooth. A physically meaningful definition of the associated non-Euclidean distance is:

$$\|\mathbf{x}-\mathbf{x}_i\|_{g_i} = \inf_{\mathcal{C}\in\Gamma(\mathbf{x}_i,\mathbf{x})} \int_0^1 g_i\left(\mathcal{C}(s)\right)\|\mathcal{C}'(s)\|ds \tag{7}$$

where the infimum extends over the set Γ of all differentiable curves $\mathcal{C}$ beginning at $\mathbf{x}_i$ and ending at $\mathbf{x}$. This definition allows a rather intuitive design of g_i so that the level-sets of $\|\mathbf{x}-\mathbf{x}_i\|_{g_i}$ tend to fit the image features. Indeed, as shown in [5], they have a physical interpretation of fronts propagating from $\mathbf{x}_i$ with the image-dependent speed function $1/g_i$, the Euclidean case being re-obtained by setting $g_i = 1$. A popular choice [5] that would snap level-sets to salient contours of the image is $g_i(\mathbf{x}) = 1 + \|\nabla I\|^2$, but an essential property arises from the fact that each metric g_i can also be adapted to the local image content around each control point $\mathbf{x}_i$. Prior assumptions on the targeted class of images should drive the choice of adapted metrics, such as $g_i(\mathbf{x}) = 1 + (I(\mathbf{x}) - I(\mathbf{x}_i))^2$ in the simple case of piecewise-constant images. In more general situations, we use the local image intensity distribution $P_{\mathbf{x}_i}$ estimated in the neighborhood of $\mathbf{x}_i$ to define

$$g_i(\mathbf{x}) = 1 - \beta \log P_{\mathbf{x}_i}\left(I(\mathbf{x})\right) \tag{8}$$

where $\beta > 0$ controls the non-Euclidean part of the metric: the smaller β is the more spherical the basis function will be. This is illustrated in Fig.2 where we show various basis functions obtained with increasing values of β. The numerical computation of geodesic distances is very efficient, using fast marching [16] or sweeping methods [20].

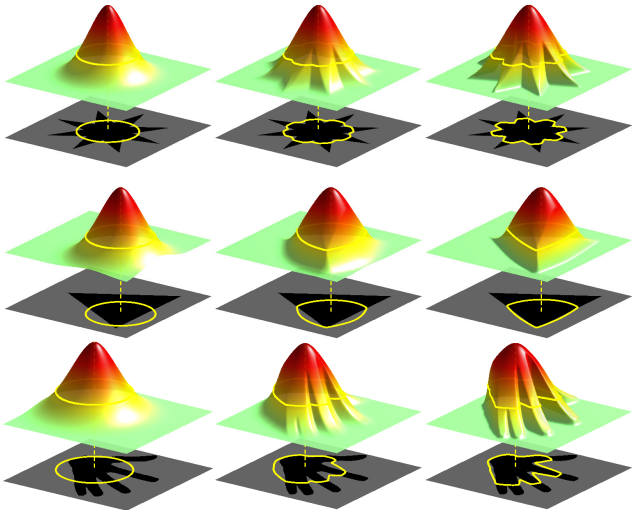


Figure 2. Switching to non-Euclidean distances: From left to right, increasing the image adaptation of the non-Euclidean metric g_i turns each basis element φ_i from a spherical-shaped RBF ($\beta = 0$, left) into feature-aligned kernels ($\beta > 0$, middle/right).

By construction, such non-Euclidean distances are meaningful only in a local neighborhood of the control point $\mathbf{x}_i$. As a consequence, the function φ must not only be non-negative as in the Euclidean case but also *monotonically-decreasing*, to discard meaningless high distance values. The localization of each basis function φ_i in (6) can then be controlled by its scale parameter σ_i. A Gaussian kernel would be a valid choice, but we use for complexity reasons the $\mathcal{C}^2$ compactly-supported Wendland function [21, 8]:

$$\forall a \in \mathbb{R},\ \ \varphi(a) = \begin{cases} (a-1)^4(4a+1) & \text{if } a \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Non-Euclidean distances have already been applied to image segmentation [13], in particular for interactivity [1, 6], but not for the purpose of generalizing radial basis functions to build implicit surfaces. A major advantage is that the segmentation is not directly obtained from the propagation. Instead, multiple propagations serve to span a restricted set of admissible solutions for an independent, region-based, variational formulation. Any function $\Phi_{\boldsymbol{\lambda}}$ of this finite-dimensional set is obtained by blending multiple localized fronts propagated at possibly different speeds:

$$\Phi_{\boldsymbol{\lambda}}(\mathbf{x}) = \lambda_0 + \sum_{i=1}^{N} \lambda_i \varphi\left(\frac{\|\mathbf{x}-\mathbf{x}_i\|_{g_i}}{\sigma_i}\right) \tag{10}$$

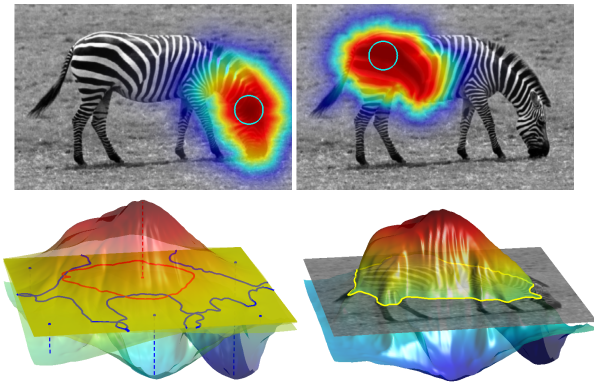


Figure 3. Implicit contour based on non-Euclidean RBFs. Top: Two different basis functions on the *zebra* image, localized fronts propagated using local intensity distributions (8) estimated in the small circles. Bottom: on the left, several basis functions with positive (red) or negative (blue) weights λ_i; on the right, the implicit function $\Phi_{\boldsymbol{\lambda}}$ obtained by a linear blending of the basis functions.

where $\mathbf{x}_i$, σ_i are given control points and scales (see Fig.3). The weights λ_i are the only unknown parameters, with an additional negative scalar $\lambda_0 < 0$ introduced as boundary condition. Indeed, we can assume that away from all control points, every pixel should eventually be included in the background region $\{\Phi_{\boldsymbol{\lambda}} < 0\}$. Since, as already mentioned, the function φ is required to vanish, then $\Phi_{\boldsymbol{\lambda}}(\mathbf{x}) \rightarrow \lambda_0$ as $\|\mathbf{x} - \mathbf{x}_i\|_{g_i} \rightarrow +\infty$, which imposes the sign of λ_0. With the above definition of the implicit function $\Phi_{\boldsymbol{\lambda}}$, we can now formulate the segmentation criterion (5) as

$$E(\boldsymbol{\lambda}) = \int_{\Omega} H\left(\lambda_0 + \sum_{i=1}^{N} \lambda_i \varphi\left(\frac{\|\mathbf{x}-\mathbf{x}_i\|_{g_i}}{\sigma_i}\right)\right) r(\mathbf{x}) d\mathbf{x} \tag{11}$$

where $r = r_1 - r_2$ and we have omitted the constant term $\int r_2$, irrelevant for the minimization with respect to $\boldsymbol{\lambda}$. Using (2), the function $r(\mathbf{x})$ is a pixel-wise likelihood test but the framework can be applied with any suitable homogeneity model [14, 4]. Finally, since multiplication by a positive scalar does not change the sign of the implicit function, $\forall \alpha > 0$, $E(\alpha\boldsymbol{\lambda}) = E(\boldsymbol{\lambda})$, minimizing E is an ill-posed problem. However, this can be fixed as in [8] by an arbitrary normalization of the vector $\boldsymbol{\lambda}$, such as $\|\boldsymbol{\lambda}\| = 1$.

Fig.4 illustrates that switching to non-Euclidean kernels significantly increases segmentation accuracy, with the exact same control points.

4. User Interactions = Inequality Constraints

As in the case of Euclidean RBFs, a good positioning of the scattered points is essential to reach a correct segmentation with a minimal number of basis functions. Deriving an automatic scheme to optimally position the control points

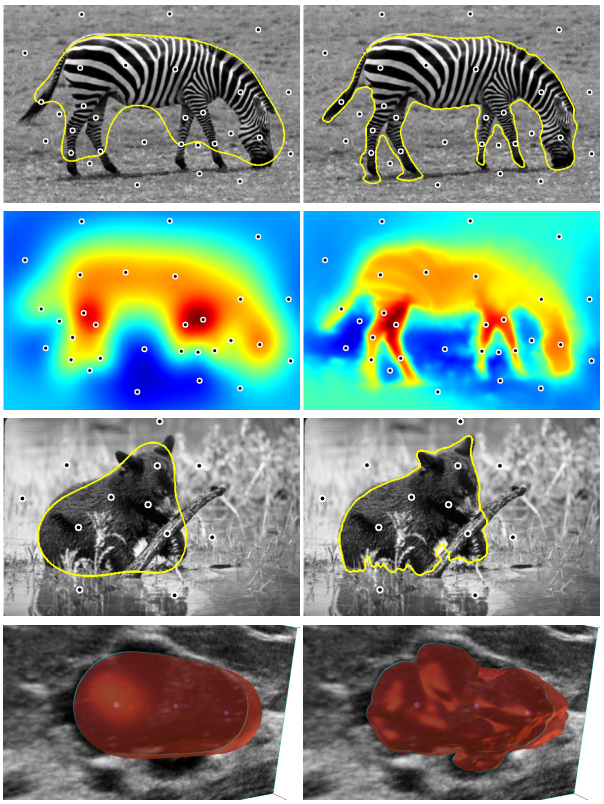


Figure 4. Non-Euclidean kernels increase segmentation accuracy. Control points $\mathbf{x}_i$ are shown as black dots. From top to bottom: Maximum-likelihood segmentation of the *zebra* image with $N = 30$ control points, using Euclidean RBFs (left) and non-Euclidean RBFs (right) built as in Fig.3; Corresponding optimal implicit functions, with Euclidean (left) and non-Euclidean (right) RBFs; segmentation of the *baby bear* image with $N = 12$ control points, with Euclidean (left) and non-Euclidean (right) RBFs; segmentation of a lesion in 3D Ultrasound with only $N = 3$ control points, with Euclidean (left) and non-Euclidean (right) RBFs.

is certainly a challenging research topic. However, in many cases, in particular in medical applications, the subjectivity of the segmentation task is such that this challenge is unreachable without additional prior knowledge. Within the framework introduced in the previous section, giving the user the possibility to position each control point, a dedicated shape space can be built that is not only adapted to the current image but also to the specific target object. Moreover, additional control and robustness can be offered if the user indicates whether control points lie inside or outside the object of interest. The low-dimensionality of our representation with few basis functions is a key feature to keep this interactive selection and labeling process easy and simple, in particular in 3D. With an implicit representation, the inside/outside labeling can be formalized as constraints on the sign of $\Phi_{\boldsymbol{\lambda}}$ at the precise location of the control points:

$$\forall k \in 1..N \qquad C_k(\boldsymbol{\lambda}) \triangleq \gamma_k \Phi_{\boldsymbol{\lambda}}(\mathbf{x}_k) \geq 0 \tag{12}$$

with $\gamma_k = 1$ (resp. -1) for inside (resp. outside) points. Note that in contrast to scattered data interpolation methods, only the sign of $\Phi_{\boldsymbol{\lambda}}$ at control points is important, not its value. Developing $\Phi_{\boldsymbol{\lambda}}$ yields N linear inequality constraints for the vector $\boldsymbol{\lambda} = \{\lambda_i\}_{[i=0\cdots N]}$

$$\forall k \in 1..N \quad \gamma_k \left(\lambda_0 + \sum_{i=1}^{N} \lambda_i \varphi \left(\frac{\|\mathbf{x}_k - \mathbf{x}_i\|_{g_i}}{\sigma_i} \right) \right) \geq 0 \tag{13}$$

Adding the background constraint at infinity $\lambda_0 \leq -\epsilon$, where ϵ is an arbitrary small positive constant, the $N+1$ constraints can be rewritten in matrix form:

$$\boldsymbol{A}\boldsymbol{\lambda} + \mathbf{b} \geq 0 \tag{14}$$

$$\text{with} \quad \boldsymbol{A} = \begin{bmatrix} -1 & & & \\ & \gamma_1 & & (0) \\ & & \ddots & \\ & (0) & & \gamma_N \end{bmatrix} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & & \boldsymbol{M} & \\ 1 & & & \end{bmatrix}$$

$$\text{and} \quad \boldsymbol{M} = \left[\varphi \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_{g_j}}{\sigma_j} \right) \right]_{1 \leq i,j \leq N}, \quad \mathbf{b} = \begin{bmatrix} -\epsilon \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Putting together the objective criterion (11) and the constraints (14) yields the minimization problem:

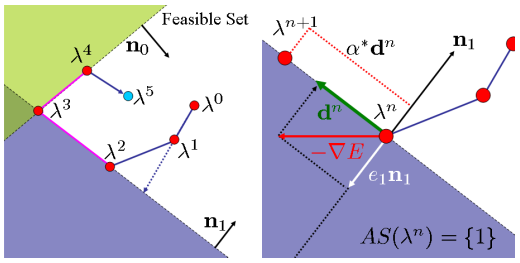
$$\begin{aligned} \min_{\boldsymbol{\lambda} \in \mathbb{R}^{N+1}} \quad & E(\boldsymbol{\lambda}) = \int_{\Omega} H \left(\sum_{i=0}^{N} \lambda_i \varphi_i(\mathbf{x}) \right) r(\mathbf{x}) d\mathbf{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{\lambda} + \mathbf{b} \geq 0 \end{aligned} \tag{15}$$

where the function φ_0 is constant with value 1. It is a non-linear optimization problem with $N+1$ variables and the same number of linear inequality constraints. In numerical textbooks, this corresponds to a particular case of *linearly constrained programming* [15] . Its feasible set, the set of $\boldsymbol{\lambda}$ satisfying the constraints, is a cone (as many constraints as variables) with hyperplane boundaries. As soon as the matrix $\boldsymbol{M}$ is invertible, this cone is non-empty and contains the summit $-\boldsymbol{A}^{-1}\mathbf{b}$. From a user perspective, this ensures that a segmentation can be found that satisfies the labeling constraints. To solve (15) numerically, we use a variant of the *Active Set* method (see Algorithm 1), which generalizes unconstrained non-linear gradient-descent to handle inequality constraints.

Algorithm 1 relies on the computation of the gradient ∇E. Since the Heaviside function is not differentiable in the usual sense, the most popular technique consists in introducing a smooth approximation H_ϵ [4] satisfying

Algorithm 1: Principles of the Active Set Algorithm

A constraint $C_k \geq 0$ (12) is *active* at $\boldsymbol{\lambda}$ if $C_k(\boldsymbol{\lambda}) = 0$.
Let $\mathbf{n}_k = [\varphi_k(\mathbf{x}_i)]_i$ be the normal to the hyperplane $\{C_k(\boldsymbol{\lambda}) = 0\}$.
The *Active Set* $AS(\boldsymbol{\lambda})$ is the set of indices of all the active constraints at $\boldsymbol{\lambda}$: $\{k | C_k(\boldsymbol{\lambda}) = 0\}$.
Given a starting feasible $\boldsymbol{\lambda}^0$ and its $AS(\boldsymbol{\lambda}^0)$,
repeat
  Compute the function gradient $-\nabla E(\boldsymbol{\lambda}^n)$
  Compute its orthogonal projection on the space spanned by $[n_k]_{k \in AS(\boldsymbol{\lambda}^n)}$:
  $$P^{\perp}(-\nabla E(\boldsymbol{\lambda}^n)) = \sum_{k \in AS(\boldsymbol{\lambda}^n)} e_k \mathbf{n}_k$$
  Compute a feasible direction by subtracting blocking components of the active set ($e_k < 0$)
  $$\mathbf{d}^n = -\nabla E(\boldsymbol{\lambda}^n) - \sum_{\{k \in AS(\boldsymbol{\lambda}^n) | e_k < 0\}} e_k \mathbf{n}_k$$
  Find optimal step $\alpha^* \geq 0$ by a bounded line search
  $$\boldsymbol{\lambda}^{n+1} = \boldsymbol{\lambda}^n + \alpha^* \mathbf{d}^n$$
  Update the *Active Set* at $\boldsymbol{\lambda}^{n+1}$: $AS(\boldsymbol{\lambda}^{n+1})$
until $\left\|\boldsymbol{\lambda}^{n+1} - \boldsymbol{\lambda}^n\right\| < \epsilon$



$\lim_{\epsilon \to 0} H_\epsilon = H$. Noting δ_ϵ the derivative of H_ϵ, an approximated gradient $\nabla_\epsilon E$ would be:

$$\nabla_\epsilon E(\boldsymbol{\lambda}) = \left[\frac{\partial E}{\partial \lambda_i} \right]_i = \left[\int_{\mathbf{x} \in \Omega} \delta_\epsilon \left(\Phi_{\boldsymbol{\lambda}} \right) \varphi_i r d\mathbf{x} \right]_i \tag{16}$$

In classical level-set implementations, Φ is usually a signed distance function and $\delta_\epsilon(\Phi)$ defines around the zero level set a *narrow band* of constant width controlled by ϵ. In contrast, parametric implicit functions built over RBFs are *not* distance functions and $\|\nabla \Phi_{\boldsymbol{\lambda}}\|$ may undergo strong variations around the zero level-set, making the band $\delta_\epsilon(\Phi_{\boldsymbol{\lambda}})$ no longer of uniform width (see Fig.5). The numerical precision of the integral computation (16) is strongly affected by the choice of ϵ and might lead to unexpected results such as unwanted topology changes. To find a better approximation and avoid this arbitrary choice, one can study the limit of $\nabla_\epsilon E$ as ϵ goes to 0. Each component of this limit gradient consists of a domain integral of the form

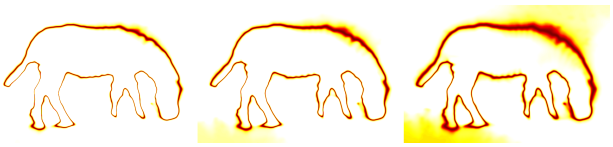


Figure 5. Approximation $\delta_\epsilon(\Phi_{\boldsymbol{\lambda}})$ for $\epsilon = 0.01$, 0.05 and 0.15

$\lim_{\epsilon \to 0} \int_\Omega \delta_\epsilon(\Phi(\mathbf{x}))f(\mathbf{x})d\mathbf{x}$. This limit (see Appendix A) reveals the generalized scaling property of the δ function:

$$\lim_{\epsilon \to 0} \int_\Omega \delta_\epsilon(\Phi(\mathbf{x}))f(\mathbf{x})d\mathbf{x} = \int_{\{\Phi=0\}} \frac{f(\mathbf{s})}{\|\nabla\Phi(\mathbf{s})\|} d\mathbf{s} \quad (17)$$

This yields the exact expression of the gradient:

$$\nabla E(\boldsymbol{\lambda}) = \left[\int_{\{\Phi_{\boldsymbol{\lambda}}=0\}} \frac{\varphi_i(\mathbf{s}) r(\mathbf{s})}{\|\nabla\Phi_{\boldsymbol{\lambda}}(\mathbf{s})\|} d\mathbf{s} \right]_i \quad (18)$$

where the domain integral (16) reduced to a boundary integral. This suggests a fast computation that does not involve any approximation of δ nor choice of ϵ: extract the zero-level of $\Phi_{\boldsymbol{\lambda}}$ by any standard algorithm and numerically integrate (18) over this boundary using interpolated values of φ_i, r and $\|\nabla\Phi_{\boldsymbol{\lambda}}\|$. Note that omitting the denominator in (18) would be justified for a distance function ($\|\nabla\Phi_{\boldsymbol{\lambda}}\| = 1$) but leads to a wrong approximation in the general case.

5. Interactive Application Workflow

Our goal is to provide as much control of the 3D segmentation process as possible with minimal and intuitive interactions in real-time. A direct application of the framework described previously would consist in first collecting user given labeled (inside/outside) points, then launching the constrained optimization algorithm. However, this workflow would not be optimal. In this section, we describe a more responsive sequential workflow in which dimensionality increases progressively as the user introduces corrections. At each step, a non-convex multidimensional minimization problem (15) should be solved, potentially facing many local minima. Unlike fully automatic segmentation algorithms for which the non-convexity is generally considered problematic, our interactive method can turn local minima into an advantage. The rationale behind the proposed workflow is to let the user drive seamlessly the optimization algorithm toward a minimum of his or her choice by deducing sound initial conditions from previous stages.

This iterative workflow is described in Algorithm 2. The user has to give the first control point required to be set inside the object of interest. The first step will propose an initial segmentation on which the user will interact to introduce corrections. It corresponds to solving problem (15) for two unknowns $\boldsymbol{\lambda} = [\lambda_0, \lambda_1]$ under constraints

Algorithm 2: Sequential WorkFlow

From single *inside* point $\mathbf{x}_1$ compute g_1 and φ_1 (N=1)
Perform 1D optimization (19) $\longrightarrow \widetilde{\boldsymbol{\lambda}}^{(1)} = [\widetilde{\lambda}_0^{(1)}, \widetilde{\lambda}_1^{(1)}]$
repeat
  New control point $\mathbf{x}_{N+1}$ and constraint C_{N+1}
  Compute g_{N+1} and φ_{N+1}
  Solve Constrained Quadratic Programming (20)
  $[\widetilde{\boldsymbol{\lambda}}^N, 0] \longrightarrow \boldsymbol{\lambda}^{N+1}$
  Run non-linear Active Set, Algorithm 1
  $\boldsymbol{\lambda}^{N+1} \longrightarrow \widetilde{\boldsymbol{\lambda}}^{N+1}$
  $N \longrightarrow N+1$
until *User is satisfied*

$\lambda_0 + \lambda_1\varphi_1(\mathbf{x}_1) > 0$ and $\lambda_0 < 0$. Since $\varphi_1 \in [0,1]$ (9), a non-empty segmentation requires $\lambda_1 > 0$. Thus, as already mentioned, $E(\boldsymbol{\lambda}) = E(\boldsymbol{\lambda}/\lambda_1)$ and (15) is equivalent to the following one-dimensional thresholding problem

$$\min_{\theta \in \mathbb{R}^+} \quad F(\theta) = \int_\Omega H\left(\varphi\left(\frac{\|\mathbf{x}-\mathbf{x}_1\|_{g_1}}{\sigma_1}\right) - \theta\right) r(\mathbf{x})d\mathbf{x} \quad (19)$$

φ being monotonous, selecting a threshold θ corresponds to a choice of the front propagated from the seed point $\mathbf{x}_1$ at the image-dependent speed $1/g_1$. This is related to the method proposed in [12], with a fundamental difference: the chosen front minimizes an objective region-based criterion $E(\mathcal{A})$ over the one-dimensional embedding defined by the positions of the propagation. This "one click" step can already provide very decent results if the metric g_1 is able to capture most of the image features (Fig.6).

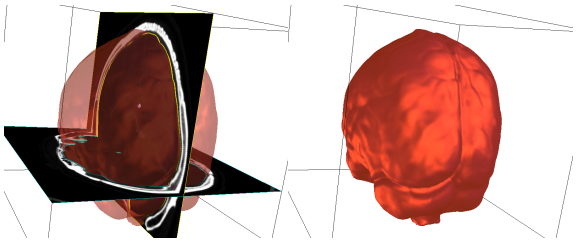


Figure 6. "One-Click" initialization, segmentation of a brain volume in CT to extract the region enclosed by the skull. Optimizing over a one-dimensional embedding of shapes (19) can be enough if the metric g_1 captures most of the image features.

Subsequently, until the user is satisfied with the result, corrections can be made by introducing a new labeled point $\mathbf{x}_{N+1}$ and hence a new constraint. Typically, the user can drop a new *outside* point where a leakage occurred, or an *inside* point where under-segmentation is observed (see Fig.7 and Fig.8). In both cases, the new constraint is violated by the current segmentation. Since Algorithm 1 has to be initialized from a *feasible* position, the difficulty now lays on

finding a suitable initialization $\boldsymbol{\lambda}^{N+1}$ that satisfies not only the new constraint but also the previous ones. This choice is critical, because Algorithm 1 will find the closest local minimum of the non-convex function E (15). To go further, we assume that an intuitive and stable correction should not only satisfy the constraints but also stay close to the current segmentation. Therefore, a sensible strategy is to find a *feasible* $\boldsymbol{\lambda}^{N+1}$ so that $\Phi_{\boldsymbol{\lambda}^{N+1}}$ is *as close as possible* to the previous optimal implicit function $\Phi_{\widetilde{\boldsymbol{\lambda}}^N}$, in the L_2 sense. Formally, this corresponds to solving after each interaction:

$$\boldsymbol{\lambda}^{N+1} = \underset{A\boldsymbol{\lambda}^{N+1}+\mathbf{b}\geq 0}{\operatorname{argmin}} \left\| \Phi_{\boldsymbol{\lambda}^{N+1}} - \Phi_{\widetilde{\boldsymbol{\lambda}}^N} \right\|_2 \tag{20}$$

Expanding functions over the basis (φ_i) we have:

$$\begin{aligned} \left\| \Phi_{\boldsymbol{\lambda}^{N+1}} - \Phi_{\widetilde{\boldsymbol{\lambda}}^N} \right\|_2 &= \int_\Omega \left(\sum_{i=0}^{N+1} \left(\widetilde{\lambda}_i^N - \lambda_i^{N+1} \right) \varphi_i \right)^2 \\ &= (\widetilde{\boldsymbol{\lambda}}^N - \boldsymbol{\lambda}^{N+1})^T \, \mathbf{G} \, (\widetilde{\boldsymbol{\lambda}}^N - \boldsymbol{\lambda}^{N+1}) \\ \text{with } \mathbf{G} &= \left[\int_\Omega \varphi_i(\mathbf{x}) \varphi_j(\mathbf{x}) \right]_{(i,j)\in[0,\cdots,N+1]} \end{aligned} \tag{21}$$

(20) is a low-dimensional constrained quadratic programming problem, if $\mathbf{G}$ is not singular its unique solution can be found in linear time [15]. Solving this problem is extremely fast, since it only depends on the scalar products between the basis functions, not on the image. The next optimal segmentation of the image that satisfies all the constraints is presented to the user after applying the non-linear Active Set method from this sound initial condition.

6. Conclusion

In this paper, we described a flexible framework for interactive image segmentation based on the selection of a few points inside and outside the object of interest. Our variational formulation is based on the minimization of a broad class of two-phase segmentation functionals, which includes the well-known maximum-likelihood criterion, taken here as an example. The crux of the technique stems from performing the minimization over a low-dimensional, image-adaptive subset of implicit functions. This subset is spanned by localized kernels, constructed with a novel extension of Radial Basis Functions with image-dependent non-Euclidean distances, using local image information around each control point. This extension opens up the design of non-spherical, feature-aligned basis functions with the immediate consequence that far fewer control points are needed to accurately recover sharp details such as straight edges or corners. Fully automatic segmentation algorithms can already benefit from this representation, but if a user provides the control points, the basis functions can be made even more specific to the targeted object. In such an interactive scenario, the inside/outside labeling can be expressed through simple linear inequality constraints in the parameter space. We applied this constrained formulation to build a general-purpose 3D segmentation tool based on a sequential workflow in which the dimension of the optimization problem increases progressively as the user interacts.

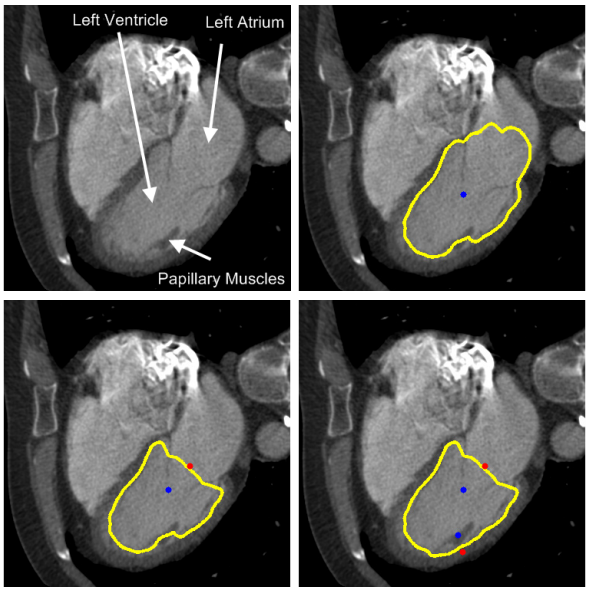


Figure 7. 2D interactive segmentation of the Left Ventricle in a Cardiac CT slice. Top-left: original image. Top-right: one single click *inside* (blue) the ventricle captures most of its shape but also the atrium, which has a very similar intensity distribution. Bottom-left: one single *outside* (red) click on the mitral valve removes the atrium. Bottom-right: Two final control points allow as fine corrections to include the papillary muscles of different intensity.

A. Generalized Scaling Property of δ

Consider a Liptschitz-continuous function Φ such that a.e. level-set is a smooth hypersurface. Let δ_ϵ be an approximation of the Dirac distribution, having compact support in $[-c, c]$. Suppose $\|\nabla\Phi\| \neq 0$ in any measurable subset of $\Phi^{-1}([-c, c])$. Using the coarea formula $\int_\Omega F\|\nabla\Phi\| = \int_{-\infty}^{+\infty} \left(\int_{\{\Phi=a\}} F(\mathbf{s}) d\mathbf{s} \right) da$ with $F = \delta_\epsilon(\Phi)\frac{f}{\|\nabla\Phi\|}$, we have

$$\begin{aligned} \int_\Omega \delta_\epsilon(\Phi) f &= \int_{-c}^{c} \left(\int_{\{\Phi=a\}} \delta_\epsilon(\Phi(\mathbf{s})) \frac{f(\mathbf{s})}{\|\nabla\Phi(\mathbf{s})\|} d\mathbf{s} \right) da \\ &= \int_{-c}^{c} \delta_\epsilon(a) \left(\int_{\{\Phi=a\}} \frac{f(\mathbf{s})}{\|\nabla\Phi(\mathbf{s})\|} d\mathbf{s} \right) da \end{aligned}$$

Finally, since by definition $\lim_{\epsilon\to 0} \int_{-c}^{c} \delta_\epsilon(a) G(a) = G(0)$,

$$\lim_{\epsilon\to 0} \int_\Omega \delta_\epsilon(\Phi(\mathbf{x})) f(\mathbf{x}) d\mathbf{x} = \int_{\{\Phi=0\}} \frac{f(\mathbf{s})}{\|\nabla\Phi(\mathbf{s})\|} d\mathbf{s}.$$

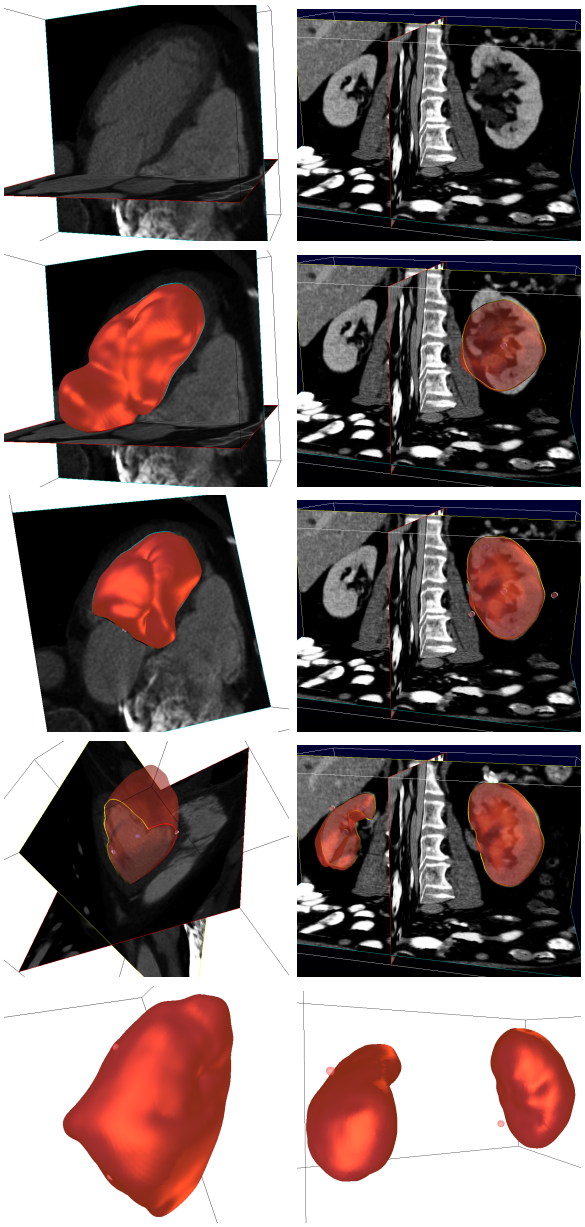


Figure 8. General-purpose 3D interactive segmentation tool, tested on a 2 GHz processor on images of typical size 256^3. Response time after each user interaction is about one second. Left column: Segmentation of the left ventricle in a cardiac CT volume. As in Fig.7, a single click in the ventricle extracts most of the 3D shape but also the atrium. The atrium is first removed with an *outside* click on the valve. 4 subsequent clicks increase the accuracy and include the papillary muscles in the ventricle ($N = 6$). Right column: Segmentation of the kidneys in a CT volume with $N = 10$ control points. Note that the second kidney, with similar intensity distribution, can easily be obtained after the first one, thanks to the topological flexibility of the implicit representation.

References

[1] X. Bai and G. Sapiro. A geodesic framework for fast interactive image and video segmentation and matting. In *ICCV*, pages 1–8, 2007.

[2] O. Bernard, D. Friboulet, P. Thevenaz, and M. Unser. Variational b-spline level-set method for fast image segmentation. In *ISBI*, pages 177–180, May 2008.

[3] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *ICCV*, volume 1, pages 26–33, October 2003.

[4] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. on Im. Proc.*, 10(2):266–277, February 2001.

[5] L. D. Cohen and R. Kimmel. Fast marching the global minimum of active contours. In *ICIP*, pages 473–476, 1996.

[6] A. Criminisi, T. Sharp, and A. Blake. Geos: Geodesic image segmentation. In *ECCV (1)*, pages 99–112, 2008.

[7] Q. Dinh, G. Turk, and G. Slabaugh. Reconstructing surfaces using anisotropic basis functions. In *ICCV*, pages 606–613, 2001.

[8] A. Gelas, O. Bernard, D. Friboulet, and R. Prost. Compactly supported radial basis functions based collocation method for level-set evolution in image segmentation. *IEEE Trans. Image Processing*, 16(7):1873–1887, 2007.

[9] A. Hanson. Hyperquadrics: Smoothly deformable shapes with convex polyhedral bounds. *Computer Vision, Graphics, and Image Processing*, 44:191–210, 1994.

[10] T. Kadir and M. Brady. Unsupervised non-parametric region segmentation using level sets. *ICCV*, 2:1267–1274, 2003.

[11] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, January 1988.

[12] R. Malladi and J. A. Sethian. Level set and fast marching methods in image processing and computer vision. In *ICIP*, pages I: 489–492, 1996.

[13] R. Malladi and J. A. Sethian. A real-time algorithm for medical shape recovery. In *ICCV*, pages 304–310, 1998.

[14] B. Mory and R. Ardon. Variational segmentation using fuzzy region competition and local non-parametric probability density functions. In *ICCV*, 2007.

[15] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, August 1999.

[16] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: algorithms based on the hamilton-jacobi formulation. 79:12–49, 1988.

[17] C. Rother, V. Kolmogorov, and A. Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004.

[18] G. G. Slabaugh, Q. Dinh, and G. Unal. A variational approach to the evolution of radial basis functions for image segmentation. In *CVPR*, pages 1–8, 2007.

[19] G. Taubin. An improved algorithm for algebraic curve and surface fitting. *ICCV*, pages 658–665, June 1993.

[20] P. J. Toivanen. New geodesic distance transforms for gray-scale images. *Patt. Recogn. Lett.*, 17(5):437–450, 1996.

[21] H. Wendland. *Scattered Data Approximation*. Cambridge university press, 2005.

[22] S. C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE PAMI*, 18(9):884–900, Sept 1996.