ROBUST REAL-TIME SEGMENTATION OF IMAGES AND VIDEOS USING A SMOOTHING-SPLINE SNAKE-BASED ALGORITHM

F. Precioso, M. Barlaud
Laboratoire I3S - UPRES-A 6070 CNRS
Université de Nice - Sophia Antipolis
2000 route des Lucioles - F-06903 Sophia-Antipolis FRANCE
Phone: +33 (0)4 92 94 27 32, Fax: +33 (0)4 92 94 28 98
{Frederic.Precioso, Michel.Barlaud}@i3s.unice.fr

T. Blu, M. Unser
Biomedical Imaging Group, STI/IOA
Swiss Federal Institute of Technology, Lausanne
CH-1015 Lausanne EPFL, SWITZERLAND
Phone: +41 21 693 51 85, Fax: +41 21 693 37 01
{Thierry.Blu, Michael.Unser}@epfl.ch

***Abstract*— This paper deals with fast image and video segmentation using active contours. Region based active contours using level-sets are powerful techniques for video segmentation but they suffer from large computational cost. A parametric active contour method based on B-Spline interpolation has been proposed in [26] to highly reduce the computational cost but this method is sensitive to noise. Here, we choose to re-lax the rigid interpolation constraint in order to robustify our method in the presence of noise: by using smoothing splines, we trade a tunable amount of interpolation error for a smoother spline curve. We show by experiments on natural sequences that this new flexibility yields segmentation results of higher quality at no additional computational cost. Hence real time processing for moving objects segmentation is preserved.**

I. INTRODUCTION

We address the problem of image and video segmentation using region-based active contours. The goal is to extract image regions corresponding to semantic objects. Image and Video segmentation can be cast in a minimization framework by choosing a criterion which includes region and boundary functionals. Boundary functionals were first proposed by Kass et al. [21] and geodesic active contours by Caselles et al. [3], [4] for active contour segmentation. Region-based active contours were first introduced by Ronfard et al. [30] and Cohen et al. [10]. Chakraborty et al. [5] combined both boundary and region information for medical images segmentation. Then Chesnaud et al. [9], Chan et al. [6], Zhu et al. [35], Paragios et al. [24] and Debreuve et al. [12] introduce region-based statistic descriptors for image or video segmentation. Jehan-Besson et al. [17], [20] address the segmentation problem where features of the region to be segmented are embedded in region functionals. In this framework, Gastaud et al. [13] propose a new approach introducing shape prior. This method uses a variational approach as opposed to previous work on shape prior, based on probabilistic methods [11]. The shape prior allows free form deformation [13] and is not restricted to a parametric deformation as in [8].

All these contour or region-based methods use a level-set approach which is accurate but time consuming. In this paper, we propose a parametric active contour evolution based on a cubic spline contour [2].

In Section 2, we present a survey of the region-based criterion, the derivation of the criterion and computation of the velocity vector.

In Section 3, we propose a cubic B-spline implementation. Cubic B-splines preserve C^2 regularity and have excellent approximation properties [32] which means that, for a given accuracy, fewer samples are needed than with other parametric methods; moreover, fast algorithms are available for B-splines, which greatly reduces the computation cost.

Unfortunately, interpolation methods are not robust to noise. This is why we propose to use smoothing splines [33] in the B-spline interpolation approach of [27]. These curves preserve the implementation advantages as the B-splines while softening the interpolation constraint. The relaxation of the interpolation condition is traded for an *optimal* increase of the smoothness of the spline snake. A smoothness parameter controls the amount of relaxation.

In Section 4, we compare the influence of the smoothing spline parameter with the curve-length regularization coefficient. Finally, we show some experiments on real video sequences.

II. REGION-BASED ACTIVE CONTOURS

A. Criterion and velocity

Let us define a general segmentation criterion. For each frame of the sequence, we search a background region Ω_{out}, and object regions Ω_{in} with a common boundary Γ (**Fig. 1**). Thus the criterion includes both region and boundary functionals:

$$J(\Omega_{\text{out}}, \Omega_{\text{in}}, \Gamma) = \underbrace{\int_{\Omega_{\text{out}}} k_{\text{out}}(\Omega_{\text{out}})\, d\sigma + \int_{\Omega_{\text{in}}} k_{\text{in}}(\Omega_{\text{in}})\, d\sigma}_{Region\ terms} + \underbrace{\int_{\Gamma} \beta\, ds}_{Boundary\ term} \quad (1)$$

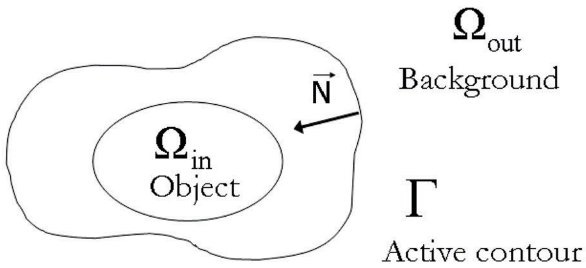


Fig. 1. Domain definition

In this criterion , k_{out} is the descriptor of the unknown background domain Ω_{out}, k_{in} is the descriptor of the unknown object domain Ω_{in} and β is the weight of the regularization.

Since we use statistical descriptors (for k_{in} and k_{out}) such as mean, variance or region histogram, the entropy descriptors are globally attached to the region Ω [9]. In the variational Eulerian method proposed by Jehan-Besson et al. [20] for a region-based active contour segmentation, the authors introduce a dynamical scheme (shape gradient method) in the criterion. Hence regions become continuously dependent on an evolution parameter τ.
The criterion $J(\Omega_{\text{out}}(\tau), \Omega_{\text{in}}(\tau), \Gamma(\tau))$ is denoted by $J(\tau)$. Thus the computation of the derivative provides:

$$J'(\tau) = \underbrace{\int_{\Omega_{\text{out}}(\tau)} \frac{\partial k_{\text{out}}}{\partial \tau}\, d\sigma + \int_{\Omega_{\text{in}}(\tau)} \frac{\partial k_{\text{in}}}{\partial \tau}\, d\sigma}_{(a)} + \underbrace{\int_{\Gamma(\tau)} (\ k_{\text{out}} - k_{\text{in}}\ )(\boldsymbol{v}.\boldsymbol{N})\ ds}_{(b)} + \underbrace{\int_{\Gamma(\tau)} (\ -\beta.\kappa + \nabla\beta.\boldsymbol{N})(\boldsymbol{v}.\boldsymbol{N})\ ds}_{(c)} \quad (2)$$

where κ is the curvature of the contour, v is the velocity of $\Gamma(\tau)$ and $\boldsymbol{N}$ is the unit inward normal to $\Gamma(\tau)$.

The terms *(a)* are deduced from the variation of the descriptors with the region. The term *(b)* is deduced from the variation of the region. And the classical term *(c)* comes from the derivation of the Boundary term in (1) [4]. Complete proofs are available in [17], [19], [20].

The active contour $\Gamma(\tau)$ evolves from an initial position $\Gamma(0)$ towards the object with a velocity v in the direction of $\boldsymbol{N}$, the inward normal vector of the active contour:

$$\begin{cases} \frac{\partial \Gamma(\tau)}{\partial \tau} &= v\boldsymbol{N} \\ \Gamma(0) &= \Gamma_0 \end{cases} \quad (3)$$

The velocity expression is deduced from the derivative (2):

$$v = \mathbf{A}\ +\ k_{\text{in}}\ -\ k_{\text{out}}\ +\ \beta\kappa \quad (4)$$

- **A** represents local terms which are computed from the two first terms (a) in (2). In Section 4, we will detail the expression of **A** and then the velocity v for two applications: segmentation of homogenous regions and segmentation of moving objects.
- β is a constant.

B. Implementation

Region-based active contour evolution can be implemented in two different ways:

- *Implicitly*, based on the level-set approach [23], [19], [20]. Such a method provides an implicit management of topological changes and yields accurate results, but it suffers from a high computational cost.
- *Explicitly*, using active parametric contours. Such a method reduces the computational cost substantially and provides a complete control of the data size. The accuracy of the results is dependent on the noise level of the sequence. Using smoothing splines is likely to introduce robustness in this method.

III. TOWARDS CUBIC SMOOTHING SPLINES

A. Cubic Spline Interpolation

The evolution velocity is now computed only at sampling points along a spline active contour. Cubic spline curves are parametric curves $S(t) = (x(t), y(t))$ where $x(t)$ and $y(t)$ are cubic polynomials on each segment $t_k \leq t \leq t_{k+1}$, and are smoothly (twice continuously differentiable) connected between segments. Here, we assume that there are n such segments parametered by $t_0, t_1, \ldots t_{n-1}$ with the assumption $t_n = t_0$, and that we are given the n sampling points $P_k = S(t_k)$.

Each segment $t_k \leq t \leq t_{k+1}$ is expressed as a cubic polynomial [1]:

$$S\left(t\right) = Q_{k-1}B^3_{k-3}\left(t\right) + Q_kB^3_{k-2}\left(t\right) + Q_{k+1}B^3_{k-1}\left(t\right) + Q_{k+2}B^3_{k}\left(t\right) \quad (5)$$

$B^3_k\left(t\right)$ is a nonuniform B-spline function; the n parameters of the model are the B-spline coefficients Q_k called *control points*. These coefficients can be specified by solving for Q_k the n equations $S(t_k) = P_k$.

The B-spline function $B^3_k\left(t\right)$ is a piecewise cubic polynomial that depends on the n values t_k of the curve parameter at the sampling points.

Irregular sampling of t is intuitively more pertinent, as regards active contour propagation, than uniform sampling. This is the option chosen by, e.g., Pottmann et al. [25] and Yang et al. [34], who propose to optimize the parameterization of the spline curve for approximating a target curve, as well as in other approaches based on arbitrary parameterizations (chord length, centripetal, Foley, ...). However, building the nonuniform spline curve requires the computation of n different polynomials B^3_k which is time consuming. To overcome this problem, we have proposed a regular sampling approach [27],i.e. $t_k = k$, to represent the active contour using uniform B-spline functions. In that case, after reparameterizing the curve, the B-spline function $B^3_k(t)$ is independent of the segment considered on the curve. We can write $B^3_k(t) =$

$\beta^3(t-k)$ where the centered B-spline of degree 3, $\beta^3(t)$, is a bell-shaped, symmetrical function, as shown on **Fig. 2**, defined by:

$$\beta^3(t) = \begin{cases} \frac{2}{3} - |t|^2 + \frac{|t|^3}{2} & , \quad 0 \le |t| < 1 \\ \frac{(2-|t|)^3}{6} & , \quad 1 \le |t| < 2 \\ 0 & , \quad 2 \le |t| \end{cases} \tag{6}$$

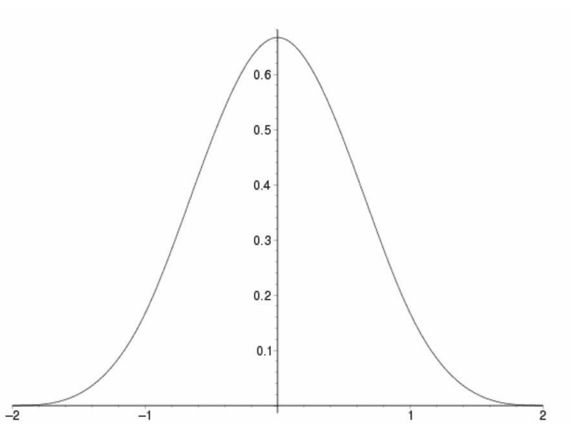


Fig. 2. Centered B-spline of degree 3

The arc equation (5) with (6) becomes [1]:

$$\begin{aligned} S(t) = &\left(-\frac{1}{6}\,Q_{k-1} + \frac{1}{2}\,Q_k - \frac{1}{2}\,Q_{k+1} + \frac{1}{6}\,Q_{k+2}\right) t^3 \\ &+ \left(\frac{1}{2}\,Q_{k-1} - Q_k + \frac{1}{2}\,Q_{k+1}\right) t^2 \\ &+ \left(-\frac{1}{2}\,Q_{k-1} + \frac{1}{2}\,Q_{k+1}\right) t \\ &+ \frac{1}{6}\,Q_{k-1} + \frac{2}{3}\,Q_k + \frac{1}{6}\,Q_{k+1} \end{aligned}$$

$$\text{for } t \in [k, k+1] \tag{7}$$

The computation cost of regular sampling is lower than handling a specific equation for each segment $t_k \le t \le t_{k+1}$ of a nonuniform B-Spline curve.

Moreover, the control points Q_k can be obtained from the sampling points P_k using a fast filtering algorithm when the curve parameter is sampled uniformly (see Appendix I).

Indeed, each interpolated point P_k corresponds to the polynomial expression value $S(t)$ when $t = k$. From the expression (7) we obtain the relation between sampling points P_k and control points (B-spline coefficients) Q_k:

$$S(k) = P_k = \frac{1}{6}\left(Q_{k-1} + 4Q_k + Q_{k+1}\right) \tag{8}$$

This relation can be written as a convolution:

$$Q_k = (B_1^3)^{-1} * P_k \tag{9}$$

where B_1^3 is the discrete cubic B-spline kernel.

The inverse convolution operator is defined by:

$$B_1^3(z)^{-1} = \frac{6}{z + 4 + z^{-1}} \tag{10}$$

Using the prefiltering approach exposed in [33], the inverse convolution operator $(B_1^3)^{-1}$ is computed efficiently from a cascade of first order causal and anti-causal recursive filters (see details in Appendix I). The control points Q_k are computed from sampling points P_k using this fast filtering algorithm.

Cubic splines provide good interpolation accuracy at low computational cost [32]. Moreover these curves have several interesting properties: They are twice continuously differentiable, which allows to build a C^2-regular curve. Thus the normal vector and the curvature, involved in the velocity equation, can be computed exactly at every sampling point. In addition, such curves minimize the following criterion:

$$\int_\Gamma \|\mathcal{C}''(t)\|^2 dt \tag{11}$$

under interpolatory constraints [31]. Here $\mathcal{C}(t) = (x(t), y(t))$ is a parametric description of the curve Γ, and $\mathcal{C}''(t)$ the second derivative of $\mathcal{C}(t)$ w.r.t. t; this functional is actually very close to the (squared) curvature κ^2 when the parameter t is close to the curvilinear abscissa, as shown in [16].

Although we have obtained real time accurate results with an implementation based on these B-spline curves [26], interpolation is not robust enough in the presence of noise. Thus we propose to use a less constrained approximation method; namely, the smoothing spline method [28].

B. Cubic Spline Approximation

A smoothing spline is an approximation curve controlled by a parameter trading interpolation error for smoothness [29]. It minimizes the following criterion:

$$\int_\Gamma \|\mathcal{C}''(t)\|^2 dt + \frac{1}{\lambda}\sum (P_k - \mathcal{C}_k)^2 \tag{12}$$

where the P_k's are the measured data points and $\mathcal{C}_k$'s are points, on the curve, joining of polynomial pieces.

The result is still a cubic spline, but it does not satisfy anymore the interpolation condition exactly. The interpolation error has been converted into increased smoothness—smaller energy of the second derivative.

The first term of (12) can be developed as:

$$\begin{aligned} \|\mathcal{C}''\|^2 &= \frac{|x'x'' + y'y''|^2}{x'^2 + y'^2} + \frac{|x'y'' - x''y'|^2}{x'^2 + y'^2} \\ &= \left|\frac{d}{dt}\sqrt{x'^2 + y'^2}\right|^2 + (x'^2 + y'^2)\,\left|\frac{d}{dt}\arctan\left(\frac{y'}{x'}\right)\right|^2 \\ &= s''^2 + s'^2\phi'^2 \end{aligned}$$

where s is the curvilinear abscissa of the curve Γ, and $\phi(t)$ is the angle of the tangent to the curve at $\mathcal{C}(t)$. This shows that the smoothing part in (12) can be rewritten as a sum of two positive terms:

$$\int_\Gamma \|\mathcal{C}''(t)\|^2 dt = \int_\Gamma s''^2(t) dt + \int_\Gamma s'^2(t)\phi'^2(t) dt \tag{13}$$

Thus the decrease of $\int_\Gamma \|C''(t)\|^2 dt$ facilitates the decrease of $\int_\Gamma s''^2(t)dt$ and $\int_\Gamma s'^2(t)\phi'^2(t)dt$:

- the first term of (13) represents the average variation of curvilinear abscissa. Thus decreasing this term tends to improve the curve sampling uniformity;
- when the sampling is nearly uniform—i.e., $s'(t) \approx$ Constant, which is favored by the decrease of the first term—the second term of (13) is lower bounded by the square of (using Cauchy-Schwartz inequality)

$$\frac{\text{length}(\Gamma)}{(\#\ \text{control points})^{3/2}} \int_\Gamma |\phi'(t)|\, dt$$

which is the average tangent angle variation over the curve. Preventing this quantity from being large also prevents loops in the curves. This is because each loop increases the value of $\int_\Gamma |\phi'(t)|\, dt$ by π at least.

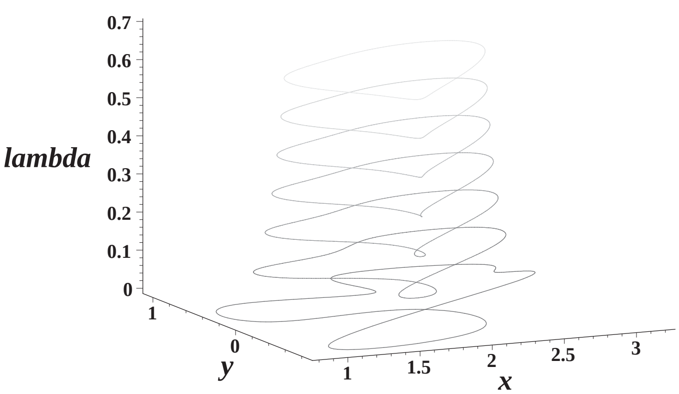


Fig. 3. Evolution from Cubic Spline interpolation ($\lambda = 0$) towards Cubic Smoothing Spline approximation ($\lambda = 0.7$)

Fig. 3 shows the evolution of a cubic spline curve, based on 10 data points, from regular interpolation (for $\lambda = 0$) towards a smoothing spline approximation (for $\lambda = 0.7$). This figure shows that loops are avoided as the smoothness parameter λ increases.

In **Fig. 4** and **Fig. 5**, we have plotted the variance of the curvilinear abscissa $\int_\Gamma s''(t)^2\, dt$ and the interpolation error of smoothing splines as functions of the smoothness parameter λ.

The curve in **Fig. 4** confirms that the interpolation error increases *slightly* with λ. Indeed, the amount of missclassified pixels, between the smoothing spline segmentation and a segmentation of reference, increases with λ but only up to 1.6% of the size of the object.

Fig. 5 shows that, on the contrary, the curvilinear abscissa variance decreases with λ.

The relation between sampling points P_k and control points Q_k (B-spline coefficients) can be written as a convolution:

$$Q_k = (S^3_\lambda)^{-1} * P_k \tag{14}$$

with

$$S^3_\lambda(z)^{-1} = \frac{6}{z + 4 + z^{-1} + 6\lambda(z^{-2} - 4z^{-1} + 6 - 4z + z^2)} \tag{15}$$

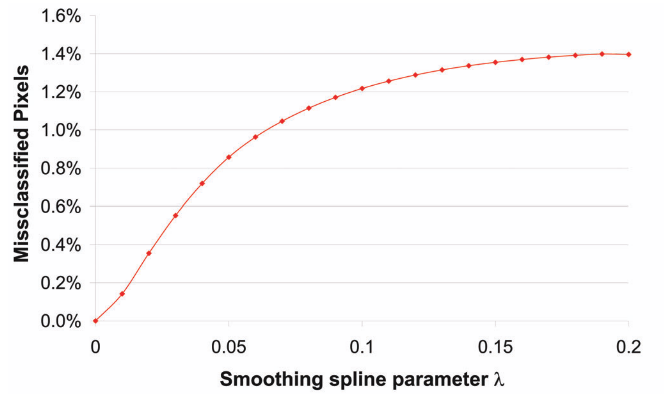


Fig. 4. Accuracy decreases *slightly* with λ

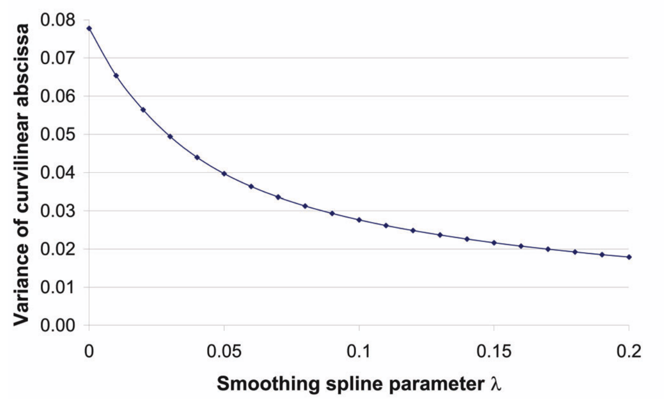


Fig. 5. Sampling uniformity increases with λ

This transfer function is factorized into a product of causal and anticausal responses:

$$\frac{b}{\lambda} \cdot \frac{1}{1 + az^{-1} + bz^{-2}} \cdot \frac{1}{1 + az + bz^2} \tag{16}$$

where $a = -(z_0 + z_1)$ and $b = z_0 z_1$ are real quantities obtained from the two poles z_0 and z_1 of $S^3_\lambda(z)^{-1}$ that are *inside* the unit circle. This prefiltering approach provides an efficient method to compute the smoothing spline coefficients (see details in Appendix II).

The positive parameter λ quantifies the tradeoff between interpolation error and regularity. For $\lambda = 0$, no interpolation error is allowed and thus, we get interpolating splines. When λ increases, a larger amount of interpolation error is allowed, hence the spline snake is smoother and its sampling more regular. We show in the next section the benefits of this new approach for the segmentation of noisy images.

IV. SEGMENTATION OF NOISY DATA

In this section, we present results of segmentation of static images and segmentation of video sequences. We compare the respective influence of the smoothing spline parameter λ and the contour length regularisation parameter β on the segmentation quality. We first show results of a segmentation based on a homogeneity criterion.

A. Segmentation of Homogeneous Regions

1) Grayscale still images: In this example, the images are osteoporosis medical images[1]. The goal is to segment bone regions in the image. We consider the functional (1) where k_{out} and k_{in} are descriptors of the bone homogeneity. The region homogeneity is characterized by a function of the variance of luminance intensity. Let σ^2_{out} and μ_{out} represent respectively the variance and mean of $\Omega_{\text{out}}(\tau)$, σ^2_{in} and μ_{in} represent respectively the variance and mean of $\Omega_{\text{in}}(\tau)$, and $\Phi(r)$ a positive $C^1(\mathbb{R})$ function, for instance $\Phi(r) = \log(1 + r^2)$.
Thus the criterion to be minimized is:

$$J(\Omega_{\text{out}}, \Omega_{\text{in}}, \Gamma) = \int_{\Omega_{\text{out}}} \Phi(\sigma^2_{\text{out}})\ d\sigma + \int_{\Omega_{\text{in}}} \Phi(\sigma^2_{\text{in}})\ d\sigma + \int_{\Gamma} \beta\ ds \quad (17)$$

where β is a positive constant.

Using (2), the derivative of the functional J with respect to τ is:

$$J'(\tau) = -\int_{\Gamma(\tau)} \Big[\Phi(\sigma^2_{\text{in}}) - \Phi(\sigma^2_{\text{out}}) + \beta\kappa + \Phi'(\sigma^2_{\text{in}})\left[(I-\mu_{\text{in}})^2 - \sigma^2_{\text{in}}\right] - \Phi'(\sigma^2_{\text{out}})\left[(I-\mu_{\text{out}})^2 - \sigma^2_{\text{out}}\right]\Big](\boldsymbol{v}.\boldsymbol{N})\ ds \quad (18)$$

where κ is the curvature of the contour and β is a constant.

More details and proofs are available in Jehan-besson et al. [19]. In order to find a local extremum of the criterion (17), as the authors proposed, we evolve a curve using the steepest descent method. Thus, we obtain the following evolution equation:

$$\frac{\partial\Gamma(\tau)}{\partial\tau} = \boldsymbol{v}.\boldsymbol{N}$$

Thus the expression (4) of the velocity $\boldsymbol{v}$ is known:

$$v = \Phi(\sigma^2_{\text{in}}) - \Phi(\sigma^2_{\text{out}}) + \beta\kappa + \Phi'(\sigma^2_{\text{in}})\left[(I-\mu_{\text{in}})^2 - \sigma^2_{\text{in}}\right] - \Phi'(\sigma^2_{\text{out}})\left[(I-\mu_{\text{out}})^2 - \sigma^2_{\text{out}}\right] \quad (19)$$

This velocity makes the B-spline active contour evolve towards the minimum of the energy criterion (17). Thus the competition between the region inside the contour $\Omega_{\text{in}}(\tau)$ and the region outside $\Omega_{\text{out}}(\tau)$ leads to increase the homogeneity of both regions. However the images are corrupted by acquisition noise and by the noise of non-bone tissues (muscles, fat, ...).

Fig. 6 shows the convergence using the cubic spline interpolation with length penalty method. The length penalty provides smoothness to the contour. However the acquisition noise corrupts the segmentation quality in the area of interest for osteoporosis diagnostic. The smoothness of the contour depends only on the length penalty parameter β.

[1]Thanks to DMS for providing osteoprosis-images

Fig. 7 shows the convergence using the new smoothing spline method. The flexibility of the smoothing splines provides an accurate bone segmentation without being corrupted by the noise. The smoothness of the contour depends only on the smoothing spline parameter λ.

Fig. 8 shows the robustness of the new smoothing spline method regarding λ parameter variations. Between the left and the right picture on the top row, λ is only increased from 0 to 0.01 (**Fig. 8(a)** and **Fig. 8(b)**), but still the accuracy of the segmentation is highly improved and the most efficient results are almost reached. If we increase again λ ten times, up to 0.1, (**Fig. 8(c)**), the segmentation is smooth but still very good. By increasing λ ten times (**Fig. 8(d)**), the contour is too smooth but remains robust. Additional experiments with $\lambda = 10$ and 100 indicate that even if the contour is too smooth to preserve a sufficient segmentation accuracy, the structure of the contour remains stable which is not true with variations of β, the weight on the length penalty. Such relative robustness of the snake with respect to λ suggests that we can determine a range for standard values of this parameter. This first experiment provides a range of $[\,0.1\,,\,1\,]$.

The contour is sampled with 512 knots and the size of the image is 512×512. The segmentation is obtained in 25 seconds with a Pentium IV at 2.6 GHz. It has to be pointed out that most of this computation time is spent in evaluating the variance of the object domain and of the background. Indeed segmentations with 256 and 128 knots, obtained in 24 seconds, confirm it: the cost of the approximation amounts to a few percents of the full computation.
To improve the computation time of our algorithm with such descriptors based on area moments (mean of the intensity, variance, ...), a perspective would be to implement the method of Jacob et al. [15] for an exact computation of the area moments of spline curves.

2) Color Video Sequences: As detailed by Jehan et al. in [19], [20], the homogeneity in color images is related to the determinant of the covariance matrix for Gaussian distributions.
Yezzi et al. [22] and Herbulot et al. [14], extended this framework to the more general case of Entropy descriptors without Gaussian distribution hypothesis.

In these experiments, regions of interest are regions of homogeneous color, like the face on the sequence *Erik*. The color images are in the RGB color space. Let us define the joint probability distribution

$$q(I_R(x), I_G(x), I_B(x), x \in \Omega)$$

and the joint entropy, between the three channels of the image, on the domain Ω:

$$H_{RGB}(\Omega) = -\int_{\Omega} q(I_R(x), I_G(x), I_B(x), x \in \Omega) \ln q(I_R(x), I_G(x), I_B(x), x \in \Omega)\, dx \quad (20)$$

The segmentation of homogeneous regions of a color video sequence is achieved by region competition between the background Ω_{out} and the object Ω_{in}, minimizing the following

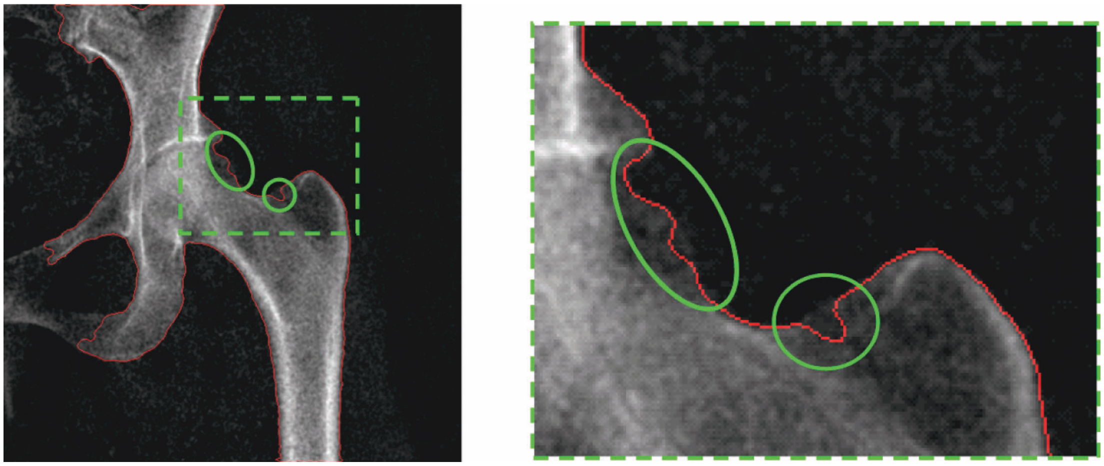


Fig. 6. Regular spline segmentation with $\lambda = 0$ and length penalty $\beta = 10$

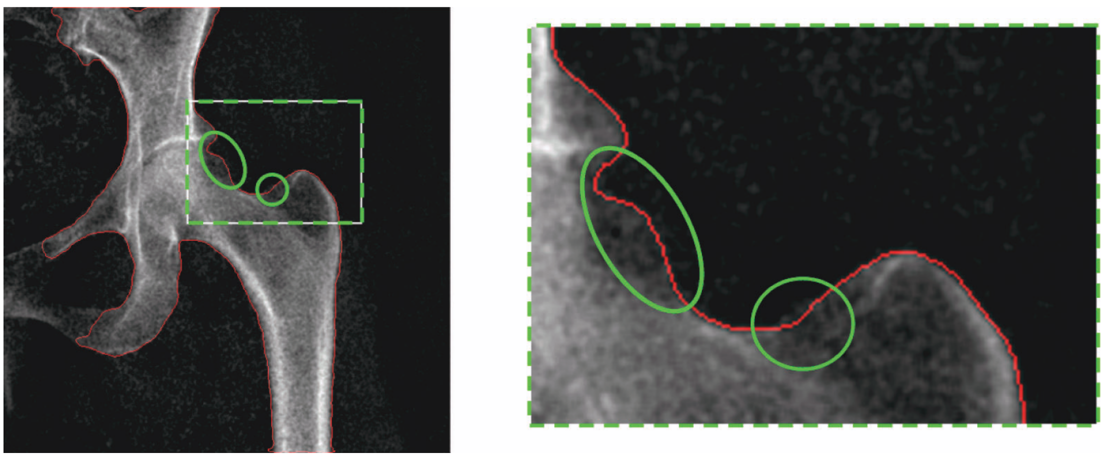


Fig. 7. Smoothing spline segmentation with $\lambda = 0.08$ and $\beta = 0$

criterion:

$$J(\Omega_{in}, \Omega_{out}, \Gamma) = H_{RGB}(\Omega_{in}) + H_{RGB}(\Omega_{out}) + \int_\Gamma \beta ds \quad (21)$$

Thus, applying DREAMS method [20] to (21), the evolution equation based on the joint probability distributions is [14]:

$$\frac{\partial \Gamma}{\partial \tau}(\tilde{x}) = \Big[-q(I_R(\tilde{x}), I_G(\tilde{x}), I_B(\tilde{x}), \Omega)(\ln q(I_R(\tilde{x}), I_G(\tilde{x}), I_B(\tilde{x}), \Omega) + 1) - \frac{1}{|\Omega|}\Big(H_{RGB}(\Omega) - 1 + \int_\Omega K(I_R(x) - I_R(\tilde{x}), I_G(x) - I_G(\tilde{x}), I_B(x) - I_B(\tilde{x})) \ln q(I_R(x), I_G(x), I_B(x), \Omega) dx\Big)\Big] \mathbf{N} \quad (22)$$

where

$$K(x, y, z) = \frac{1}{2\pi\sigma^2} \exp -\frac{x^2 + y^2 + z^2}{2\sigma^2} \quad (23)$$

is the Gaussian kernel involved in the Parsen window method.

Fig. 9 shows the evolution of the curve and **Fig. 10** the segmentation of some frames of a sequence.

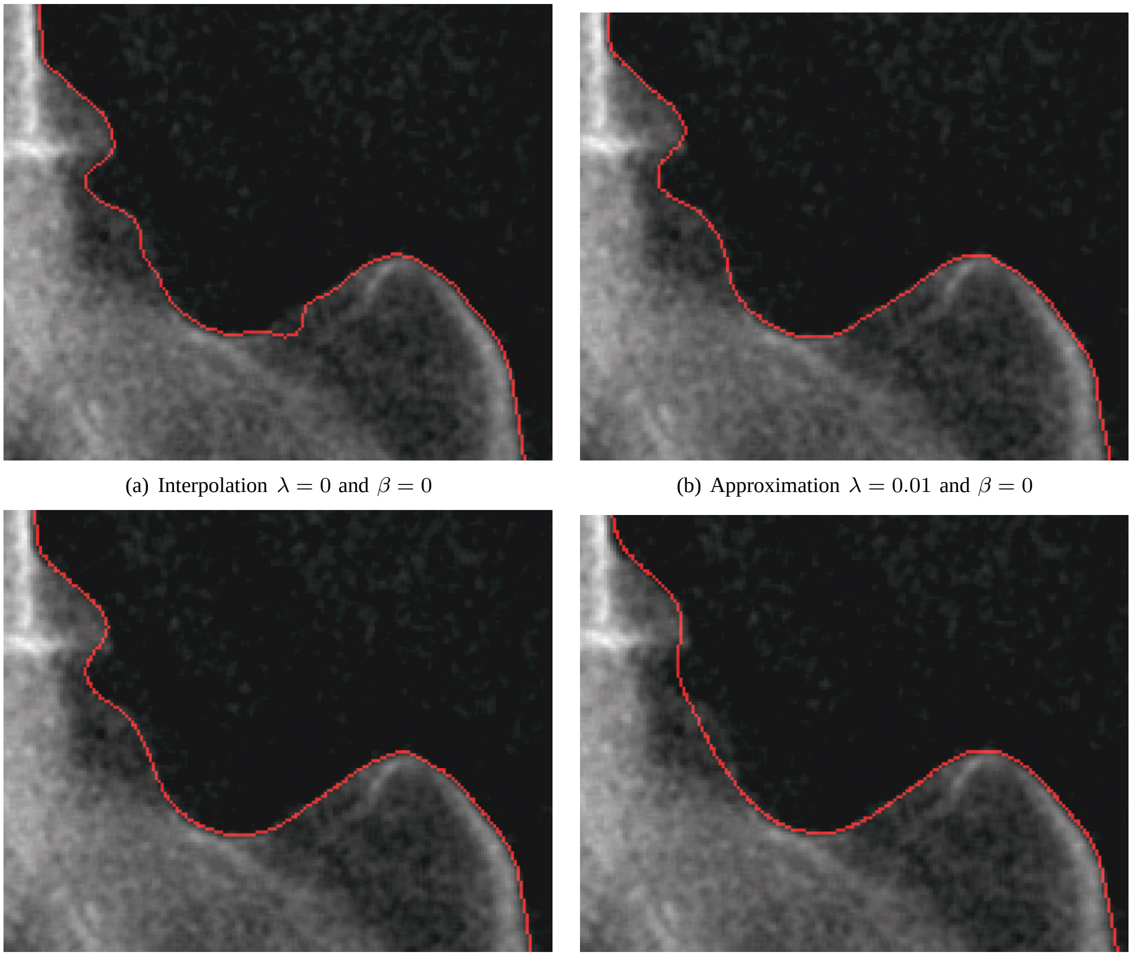
The data extracted from the histogram evolution are very sensitive to noise. This is why we use a smoothing B-spline approach which combines a very low computational cost and a global robustness to noisy data. The parameter $\lambda = 0.1$ is in the standard range determined in the previous experiment, i.e., $[0.1\,,\,1]$.

B. Moving Objects Segmentation

Now, we present results obtained for the segmentation of moving objects in video sequences. This segmentation is based on motion detection. Our method is applied to the real "coastguard" video. The goal is to detect the boat of the coastguards in the sequence. We consider the functional (1) where k_{in} and k_{out} are respectivelly descriptor of moving objects and descriptor of the background. The descriptors have to take into account the camera motion in this sequence. Thus the criterion to be minimized is:

$$J(\Omega_{\text{out}}, \Omega_{\text{in}}, \Gamma) = \int_{\Omega_{\text{out}}} |I_n(\sigma) - \text{Proj}(I_{n-1}(\sigma))| \, d\sigma + \int_{\Omega_{\text{in}}} \alpha \, d\sigma + \int_\Gamma \beta \, ds \quad (24)$$

where $\text{Proj}(I_{n-1}(\sigma))$ is the projection of the image I_{n-1} onto the referential of image I_n in order to compensate for the motion of the camera. The camera motion model is based on a 6-parameter affine model. These parameters are computed with



(a) Interpolation $\lambda = 0$ and $\beta = 0$

(b) Approximation $\lambda = 0.01$ and $\beta = 0$

(c) Approximation $\lambda = 0.1$ and $\beta = 0$

(d) Approximation $\lambda = 1$ and $\beta = 0$

Fig. 8. Robustness regarding Smoothing spline parameter λ variations

a robust estimation using motion vectors [7]. The motion field is evaluated by a classical Block Matching algorithm between frames I_n and I_{n-1} [18]. β and α are two positive constants.

The descriptors are region-independent. Thus the local terms of the differentiation (2) disappear and the velocity equation (4) reduces to:

$$v \;=\; k_{\text{in}} \;-\; k_{\text{out}} \;+\; \beta\,\kappa \tag{25}$$

Since the descriptor $k_{\text{out}} = |I_n - \text{Proj}(I_{n-1})|$ is a temporal gradient, this local term is noise sensitive.

In the "coastguard" sequence **Fig. 11(a)**, the wake of the boat behaves like noise for the background descriptor k_{out}. Thus the contour evolution equation is corrupted by noise.

Fig. 11(b) shows the results using the cubic spline interpolation method [27]. The smoothness of the contour depends only on the contour length regularization parameter β. However, the foam in the wake of the boat is kept as part of the object.

Fig. 11(c) shows the results using the new smoothing spline method proposed here. The smoothness of the contour depends only on the smoothing spline parameter λ. Relaxing the rigid interpolation constraint brings an obvious improvement: the foam is not kept anymore, whereas the object is still reasonably well-segmented.

This third experiment confirms the range of $[\,0.1\,,\,1\,]$, we determined, as standard values for the parameter λ.

Fig. 12 shows the computation time and the accuracy of the segmentation on the "coastguard" sequence, for the 5 first images. The contour is sampled with 64 knots and the size of the image is 352×288. The sequence is segmented with a $Pentium\, IV\ at\ 2.6GHz$. The initial contour for the first frame is given by the image boundaries. For the frames $2,\ 3,\ 4\ and\ 5$ the initialisation is provided by the final contour in the previous frame. The object in the first frame takes more time to be segmented because the initial contour is "far" from the object. For the other frames, the segmentation is achieved in less than 0.40 second per frame (with a $Pentium\, IV\ at\ 2.6GHz$). Thus, the whole segmentation process should be 10 times faster to provide results in real time. According to our knowledge, such a factor is not out of reach by optimization for an industry expert.

We can thus say that the smoothing spline method provides global robustness to noise-like data. The accuracy results on a real video sequence show the improvement of our smoothing spline method over a direct regularization of the segmentation criterion.

V. CONCLUSION

In this paper, we address the problem of image and video segmentation by working out a new region-based method using cubic smoothing spline active contours.

Instead of spline interpolation, we have chosen a smoothing spline approximation because we want the method to be more robust in the presence of noise. The smoothing spline parameter λ provides a tunable tradeoff between *interpolation error* and *contour smoothness*. Furthermore, increasing the smoothing spline parameter λ improves the sampling uniformity of the contour and avoids the presence of loops. The structure of the contour remains stable to variations of this parameter which is not true with variations of β, the weight on the length penalty. The robustness of the active contour with respect to λ suggested that we could determine a range for standard variations of this parameter. Our experiments provided a range of $[0.1\,,\,1]$. As a consequence of the very low computational cost of the B-spline implementation, real time segmentation is achieved.

AKNOWLEDGEMENT

We would like to thank A. Herbulot, Ph.D. student at I3S Lab, for her helpful contribution to the section on segmentation of homogeneous regions in color video sequences.

REFERENCES

[1] R. Bartels, J. Beatty, and B. Barsky, *An introduction to Splines for use in Computer Graphics and Geometric Modeling*. Los Altos, Californie: Morgan-Kaufmann, 1987.

[2] P. Brigger, J. Hoeg, and M. Unser, "B-Spline snakes: A flexible tool for parametric contour detection," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1484–1496, September 2000.

[3] V. Caselles, F. Catte, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," in *Numerische Mathematik*, vol. 66, 1993, pp. 1–33.

[4] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *IJCV*, vol. 22, no. 1, pp. 61–79, 1997.

[5] A. Chakraborty, L. Staib, and J. Duncan, "Deformable boundary finding in medical images by integrating gradient and region information," *IEEE Transactions on Medical Imaging*, vol. 15, pp. 859–870, dec. 1996.

[6] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[7] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 2, pp. 298–311, fvrier 1997.

[8] Y. Chen, H. Tagare, S. Thiruvenkadam, F. Huang, D. Wilson, K. Gopinath, R. Briggs, and E. Geiser, "Using prior shapes in geometric active contours in a variational framework," *IJCV*, vol. 50, no. 3, pp. 315–328, 2002.

[9] C. Chesnaud, P. Refregier, and V. Boulet, "Statistical region snake-based segmentation adapted to different physical noise models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1145–1156, 1999.

[10] L. Cohen, E. Bardinet, and N. Ayache, "Surface reconstruction using active contour models," in *SPIE Conference on Geometric Methods in Computer Vision*, San Diego, CA, 1993.

[11] D. Cremers, C. Schnorr, and J. Weickert, "Diffusion-snakes: Combining statistical shape knowledge and image information in a variational framework," in *Proceedings of 1st IEEE Workshop on Variational and Level Set Methods in Computer Vision*, Vancouver, B.C., Canada, 2001, pp. 137–144.

[12] E. Debreuve, M. Barlaud, G. Aubert, and J. Darcourt, "Space time segmentation using level set active contours applied to myocardial gated SPECT," *IEEE Transactions on Medical Imaging*, vol. 20, no. 7, pp. 643–659, July 2001.

[13] M. Gastaud, M. Barlaud, and G. Aubert, "Combining shape prior and statistical features for active contour segmentation," *to appear in IEEE TCSVT special session on Audio and Video Analysis for Interactive Multimedia Services*, May 2004.

[14] A. Herbulot, S. Jehan-Besson, M. Barlaud, and G. Aubert, "Shape gradient for image segmentation using information theory," in *International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Quebec, Canada, May 2004.

[15] M. Jacob, T. Blu, and M. Unser, "An exact method for computing the area moments of wavelet and spline curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 633–642, June 2001.

[16] ——, "A unifying approach and interface for spline-based snakes," vol. 4322, San Diego CA, USA, February 2001, pp. 340–347, part I.

[17] S. Jehan-Besson, M. Barlaud, and G. Aubert, "Video object segmentation using Eulerian region-based active contours," in *International Conference on Computer Vision*, Vancouver, Canada, October 2001.

[18] ——, "A 3-step algorithm using region-based active contours for video objects detection," *EURASIP Journal on Applied Signal Processing, Special issue on Image Analysis for Multimedia Interactive Services*, vol. 2002, no. 6, pp. 572–581, juin 2002.

[19] ——, "DREAM^{2}S: Deformable regions driven by an Eulerian accurate minimization method for image and video segmentation. application to face detection in color video sequences," in *ECCV*, Copenhagen, Denmark, May 2002.

[20] ——, "DREAM^{2}S: Deformable regions driven by an Eulerian accurate minimization method for image and video segmentation," *IJCV*, vol. 53, no. 1, pp. 45–70, June 2003.

[21] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *IJCV*, vol. 1, pp. 321–332, 1988.

[22] J. Kim, J. Fisher III, A. Yezzi, M. Cetin, and A. Willsky, "Nonparametric methods for image segmentation using information theory and curve evolution," in *International Conference on Image Processing*, Rochester, NY, September 2002.

[23] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulation," in *Journal of computational physics*, vol. 79, 1988, pp. 12–49.

[24] N. Paragios and R. Deriche, "Geodesic active regions and level set methods for supervised texture segmentation," *The International Journal of Computer Vision*, vol. 46, no. 3, p. 223, 2002.

[25] H. Pottmann, S. Leopoldseder, and M. Hofer, "Approximation with active B-spline curves and surfaces," in *10th Pacific Conference on Computer Graphics Applications*, Beijing, CHINA, October 2002, pp. 8–24.

[26] F. Precioso and M. Barlaud, "Regular B-spline active contours for fast video segmentation," in *International Conference on Image Processing*, Rochester, NY, September 2002.

[27] ——, "B-spline active contours with handling of topology changes for fast video segmentation," *Eurasip Special issue: Image analysis for multimedia interactive services - PART II*, vol. 2002, no. 6, pp. 555–560, June 2002.

[28] F. Precioso, M. Barlaud, T. Blu, and M. Unser, "Smoothing B-spline active contour for fast and robust image and video segmentation," in *International Conference on Image Processing*, Barcelona, Spain, September 2003.

[29] C. Reinsch, "Smoothing by spline functions," *Numer. Math.*, vol. 10, pp. 177–183, 1967.

[30] R. Ronfard, "Region-based strategies for active contour models," *IJCV*, vol. 13, no. 2, pp. 229–251, 1994.

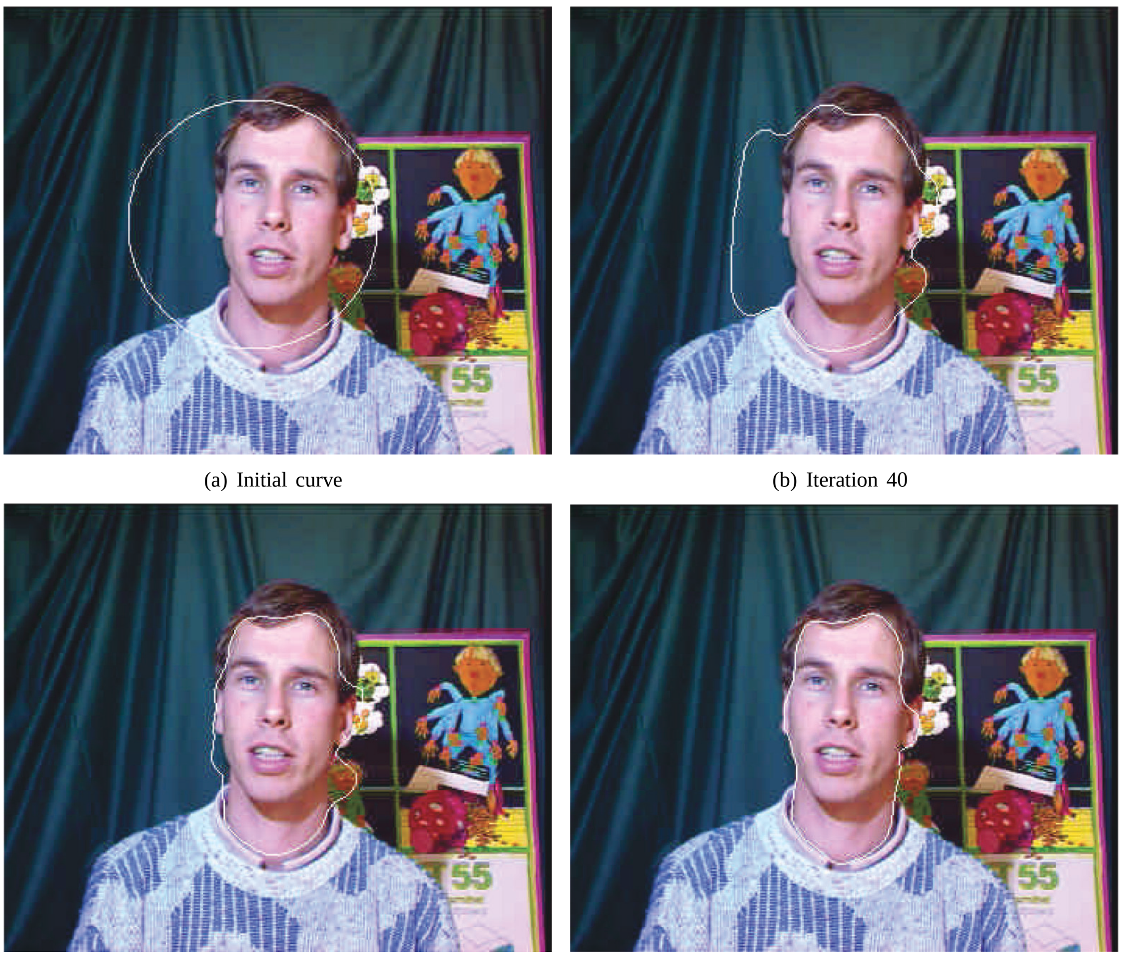
[31] I. Schoenberg, "Spline functions and the problem of graduation," *Proc. Nat. Acad. Sci.*, vol. 52, pp. 947–950, 1964.

[32] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Transactions on medical imaging*, vol. 19, no. 7, pp. 739–758, July 2000.

[33] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I-theory; part II-efficient design and applications," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 821–848, february 1993.

[34] H. Yang, W. Wang, and J. Sun, "Control point adjustment for B-spline curve approximation," *Computer Aided-Design*, in print, 2004.

[35] S. Zhu and A. Yuille, "Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *PAMI*, vol. 18, pp. 884–900, september 1996.



(a) Initial curve

(b) Iteration 40

(c) Iteration 100

(d) Final segmentation (Iteration 430)

Fig. 9. Evolution of segmentation with the minimization of the criterion (21)

(a) Frame 10 (b) Frame 20

(c) Frame 30 (d) Frame 40

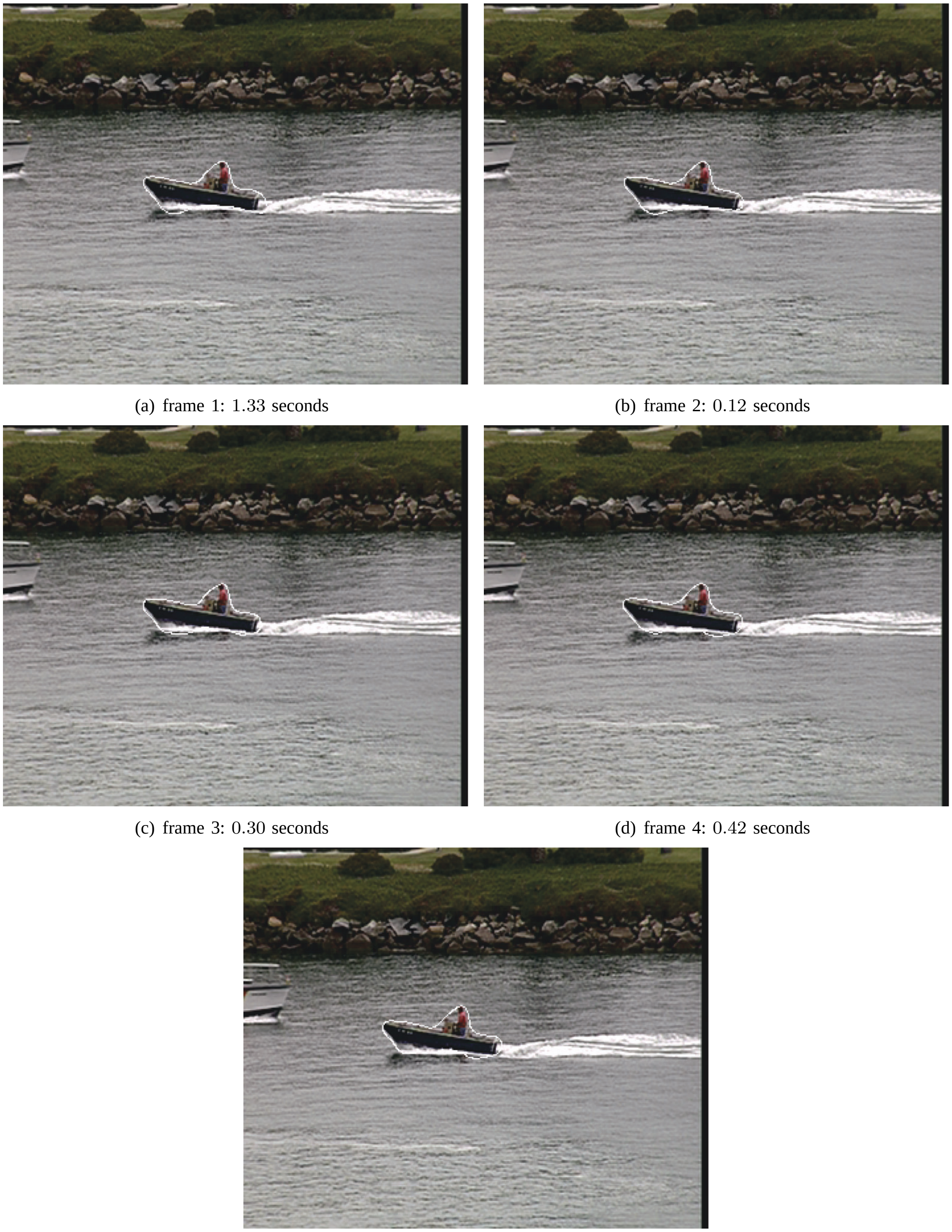
Fig. 10. Segmentation through a sequence

(a) Initial sequence

(b) Interpolation $\lambda = 0$ and $\beta = 20$

(c) Approximation $\lambda = 0.3$ and $\beta = 0$

Fig. 11. Smoothing spline to smooth contours



(a) frame 1: 1.33 seconds

(b) frame 2: 0.12 seconds

(c) frame 3: 0.30 seconds

(d) frame 4: 0.42 seconds

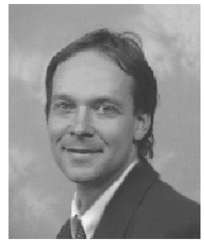
(e) frame 5: 0.25 seconds

Fig. 12. Computation time of the segmentation

Frederic Precioso was born in 1974. He received the M.S. degree in image processing and computer science from University of Nice- Sophia Antipolis in 2000. He is a Ph. D. student at the Laboratory I3S, University of Nice-Sophia Antipolis, France. His research interest deal with image and video processing, and computer vision. He is currently working on parametric active contour methods for images and videos segmentation.

Michel Barlaud received his These d'Etat from the University of Paris XII. He is currently a Professor of Image Processing at the University of Nice-Sophia Antipolis, and the leader of the Image Processing group of I3S. His research topics are: Image and Video coding using Scan Based Wavelet Transform, Inverse problem using Half Quadratic Regularization and, Image and Video Segmentation using Region Based Active Contours and PDE's. He is a regular reviewer for several journals, a member of the technical committees of several scientific conferences. He leads several national research and development projects with French industries, and participates in several international academic collaborations (Universities of Boston, Maryland, Stanford, Louvain La Neuve and Naples). He is the author of a large number of publications in the area of image and video processing, and the Editor of the book ”Wavelets and Image Communication” Elsevier, 1994.

Thierry Blu was born in Orlans, France, in 1964. He graduated from Ecole Polytechnique, Palaiseau, France, in 1986 and from Tlcom Paris (ENST), France, in 1988. In 1996, he obtained the Ph.D. in electrical engineering from ENST for a study on iterated rational filter banks, applied to wide band audio coding. He is currently with the Swiss Federal Institute of Technology, Lausanne, Switzerland, on leave from France Tlcom CNET (National Center for Telecommunication Studies), Issyles-Moulineaux, France. His research interest include (multi-) wavelets, multiresolution analysis, multirate filterbanks, approximation and sampling theory, psychoacoustics.



Michael Unser was born in Zug, Switzerland, on April 9, 1958. He received the M.S. (summa cum laude) and Ph.D. degrees in Electrical Engineering in 1981 and 1984, respectively, from the Swiss Federal Institute of Technology (EPFL) in Lausanne, Switzerland. From 1985 to 1997, he was with the Biomedical Engineering and Instrumentation Program, National Institutes of Health, Bethesda USA, where he was heading the Image Processing Group. He is now Professor and Director of the Biomedical Imaging Group at the EPFL. His main research area is biomedical image processing. He has a strong interest in sampling theories, multiresolution algorithms, wavelets, and the use of splines for image processing. He is the author of over 100 published journal papers in these areas.

Dr. Unser is the associate Editor-in-Chief of the IEEE Transactions on Medical Imaging and the Editor-in-Chief of the Wavelet Digest, the electronic newsletter of the wavelet community.

He is on the editorial boards of the IEEE Signal Processing Magazine, the Journal of Mathematical Imaging and Vision, the Journal of Visual Communication and Image Representation, Sampling Theory in Signal and Image Processing and Pattern Recognition. He is a member of the Image and Multidimensional Signal Processing Committee of the IEEE Signal Processing Society (1993-2000; 2003-present) and was former Associate Editor for the IEEE Transactions on Medical Imaging (1999-2002) including Guest Editor for its special issue on Wavelets in Medical Imaging, the IEEE Transactions on Image Processing (1992-1995) and the IEEE Signal Processing Letters (1994-1998). He co-organized the 1994 IEEE-EMBS Workshop on Wavelets in Medicine and Biology, and serves as regular chair for SPIE's conference on Wavelets, which has been held annually since 1993. He was general co-chair for the first IEEE International Symposium on Biomedical Imaging (ISBI'2002), which was held in Washington, DC, July 7-10, 2002.

He received the Dommer prize for excellence from the Swiss Federal Institute of Technology in 1981, the research prize of the Brown-Boveri Corporation (Switzerland) for his thesis in 1984, the IEEE Signal Processing Society's 1995 Best Paper Award, the IEEE Signal Processing Society's 2000 Magazine Award, and the IEEE Signal Processing Society's 2003 Best Paper Award. In January 1999, he was elected Fellow of the IEEE with the citation: ”for contributions to the theory and practice of splines in signal processing”.

Dr. Unser is a member of IEEE, EURASIP, SPIE and SIAM.

APPENDIX I
CUBIC SPLINE INTERPOLATION

A. Recursive Filter Factorization

Assuming that the data points P are uniformly sampled, the interpolating cubic spline filter $(B_1^3(z))^{-1}$, given in (10), can be factorized into a product of causal and anticausal filters [33]:

$$(B_1^3(z))^{-1} = \frac{6}{z+4+z^{-1}} = \left(\frac{-6z_1}{1-z_1 z}\right)\left(\frac{1}{1-z_1 z^{-1}}\right) \tag{26}$$

where $z_1 = -2+\sqrt{3}$.

This factorization results in a cascade of first-order causal and anticausal recursive filters. Thus, given the data points $\{P(k)\}_{,k=0,\ldots,N-1}$, the right-hand-side factorization provides the cubic spline interpolating coefficients $\{Q(k)\}_{,k=0,\ldots,N-1}$ through the following recursive algorithm:

$$\tilde{Q}(k) = z_1\,\tilde{Q}(k-1) + P(k) \tag{27}$$

$$Q(k) = z_1\,Q(k+1) - 6z_1\,\tilde{Q}(k) \tag{28}$$

for all integer k, and where $\tilde{Q}(k)$ are intermediate coefficients.

We have to specify the initialization for the two recursions.

B. Initialization

1) Causal filtering: The first recursion (27) leads to the following relation:

$$\begin{aligned}\tilde{Q}(N-k) &= P(N-k) + z_1\;P(N-k-1)\\ &+ z_1^2\;P(N-k-2)\\ &+ \ldots\\ &+ z_1^{N-1}\;P(N-k-(N-1))\\ &+ z_1^N\;\tilde{Q}(-k)\end{aligned} \tag{29}$$

Because the curve is closed, the data points are N-periodic, i.e., $P(N-k) = P(-k)$ for all integer k. As a result of the filtering operations, the coefficients $\tilde{Q}(k)$ and $Q(k)$ are N-periodic as well.

Letting $k=0$ in (29) and using the N-periodicity of $\tilde{Q}(k)$ provides $\tilde{Q}(0)$:

$$\tilde{Q}(0) = \left(\frac{1}{1-z_1^N}\right)\sum_{l=0}^{N-1} z_1^l\;P(N-l) \tag{30}$$

The others coefficients $\tilde{Q}(1)$, $\tilde{Q}(2)$, ...,$\tilde{Q}(N-1)$ are obtained by applying the induction (27).

2) Anticausal filtering: We now apply the anticausal filter (28) on the coefficients $\tilde{Q}(k)$ and, in order to trigger the recursion, we need to initialize it by providing the value of $Q(N) = Q(0)$.

Similarly as for the causal case, we obtain the following expression for $Q(0)$:

$$Q(0) = Q(N) = -\left(\frac{6z_1}{1-z_1^N}\right)\sum_{l=0}^{N-1} z_1^l\;\tilde{Q}(l) \tag{31}$$

Then the induction (28) provides $Q(N-1), Q(N-2)\ldots Q(1)$.

We have thus specified the appropriate starting values for both causal and anticausal filtering. We cascade these filters with data points P to compute cubic spline coefficients Q. The recursive algorithm is stable numerically, fast and easy to implement.

APPENDIX II
CUBIC SMOOTHING SPLINE APPROXIMATION

A. Recursive Filter Factorization

Assuming that the data points P are uniformly sampled, the smoothing cubic spline filter $(S_\lambda^3(z))^{-1}$, given in (15), can be factorized into a product of causal and anticausal filters.
Let us consider the denominator of $(S_\lambda^3(z))^{-1}$:

$$D(z) = z+4+z^{-1} + 6\lambda(z^2 - 4z + 6 - 4z^{-1} + z^{-2}) \tag{32}$$

This polynomial can be factorized as

$$\begin{aligned}D(z) &= \frac{6\lambda}{b}(1+az+bz^2)(1+az^{-1}+bz^{-2})\\ &= \frac{6\lambda}{b}S_1(z^{-1})S_1(z)\end{aligned} \tag{33}$$

where a and b are *real* numbers. Moreover, because $D(z)$ does not cancel on the unit circle, $S_1(z)$ has its roots—whether real or complex—strictly inside the unit circle. This shows that the smoothing spline prefilter can be implemented as a cascade of second-order causal and anticausal recursive filters:

$$S_\lambda^3(z) = \frac{b}{\lambda}\cdot\frac{1}{1+az^{-1}+bz^{-2}}\cdot\frac{1}{1+az+bz^2} \tag{34}$$

Note that, by defining $x = z-2+z^{-1}$ we can rewrite (32) as $D(z) = 6\lambda x^2 + x + 6$ the roots of which are either real, when $\lambda \le 1/144$, or complex, when $\lambda > 1/144$. This implies that the roots of $S_1(z)$ are either real, when $\lambda \le 1/144$, or complex, when $\lambda > 1/144$. It is only when $\lambda = 1/144$ that $S_1(z)$ has double roots.

Given the data points $\{P(k)\}_{,k=0,\ldots,N-1}$, the right-hand-side factorization of (34) leads to the cubic smoothing spline coefficients $\{Q(k)\}_{,k=0,\ldots,N-1}$ by the following recursive algorithm:

$$\tilde{Q}(k) = -a\tilde{Q}(k-1) - b\tilde{Q}(k-2) + P(k) \tag{35}$$

$$Q(k) = -aQ(k+1) - bQ(k+2) + \tfrac{b}{\lambda}\tilde{Q}(k) \tag{36}$$

We now have to specify the appropriate initialization for the two recursions.

B. Initialization

In order to determine the initialization of the recursive filtering algorithm, we need to compute the impulse response $s_1(n)$ of the causal filter $S_1(z)^{-1}$ in the cascade expression(34):

- When $\lambda = 1/144$, $S_1(z) = (1-z_0 z^{-1})^2$ with $z_0 = -5+2\sqrt{6}$ and we immediately have

  $$\frac{1}{S_1(z)} = \sum_{n\ge 0}(n+1)z_0^n\,z^{-n},$$

  from which we obtain

  $$s_1(n) = (n+1)z_0^n u(n)$$

  where $u(n)$ is the discrete step sequence $u(n) = 1$ for $n \ge 0$ and $u(n) = 0$ otherwise.

- When $\lambda \neq 1/144$, the two roots z_0 and z_1 of $S_1(z)$ are distinct. We can thus decompose $S_1(z)^{-1}$ in simple fractions:

$$\frac{1}{S_1(z)} = \frac{A}{1 - z_0 z^{-1}} + \frac{B}{1 - z_1 z^{-1}}$$

where $A = (1 - z_1 z_0^{-1})^{-1}$ and $B = (1 - z_0 z_1^{-1})^{-1}$. As a result, the impulse response of $S_1(z)^{-1}$ is given by

$$s_1(n) = A z_0^n u(n) + B z_1^n u(n).$$

Note that the impulse response of the anticausal filter $S_1(z^{-1})^{-1}$ (needed for the initialization of the anticausal recursion) is given by $s_1(-n)$.

1) Causal recursion: The initialization of (35) requires computing $\tilde{Q}(0)$ and $\tilde{Q}(1)$. Using the impulse response of $S_1(z)^{-1}$, we have:

$$\tilde{Q}(0) = \sum_{n \in \mathbb{Z}} s_1(-n) P(n),$$
$$\tilde{Q}(1) = \sum_{n \in \mathbb{Z}} s_1(1-n) P(n),$$

then, using the N-periodicity of $P(n)$ (closed contour)

$$\begin{aligned} \tilde{Q}(0) &= \sum_{n_0=0}^{N-1} P(n_0) \sum_{n \in \mathbb{Z}} s_1(-n_0 + nN), \\ \tilde{Q}(1) &= \sum_{n_0=0}^{N-1} P(n_0) \sum_{n \in \mathbb{Z}} s_1(1 - n_0 + nN). \end{aligned} \tag{37}$$

We thus need to compute an expression of the form $\sum_n s_1(k+nN)$. For this, we consider the functions

$$g_k(r) = \sum_{n \in \mathbb{Z}} r^{k+nN} u(k+nN) \quad \text{with } |r| < 1.$$

Since $g_k(r)$ obviously satisfies $g_k(r) = g_{k+N}(r)$, we can restrict k to $[0, N-1]$ and we find

$$g_k(r) = \sum_{n \geq 0} r^{k+nN} = \frac{r^k}{1 - r^N}.$$

Thanks to the N-periodicity, k has to be replaced by $(k \bmod N)$ in this expression when $k \notin [0, N-1]$. Moreover, by simple differentiation of $g_{k+1}(r)$, we also have that

$$\begin{aligned} \sum_{n \in \mathbb{Z}} (k + nN + 1) r^{k+nN} u(k+nN) &= g'_{k+1}(r) \\ &= \frac{(k+1) r^k}{1 - r^N} + \frac{N r^{N+k}}{(1 - r^N)^2}. \end{aligned} \tag{38}$$

Finally, we find that

- when $\lambda = 1/144$,

$$\sum_{n \in \mathbb{Z}} s_1(k + nN) = g'_{(k \bmod N)+1}(z_0); \tag{39}$$

- when $\lambda \neq 1/144$,

$$\sum_{n \in \mathbb{Z}} s_1(k + nN) = A\, g_{k \bmod N}(z_0) + B\, g_{k \bmod N}(z_1). \tag{40}$$

These expressions can be substituted in (37) to provide the initial conditions to the recursion (35).

2) Anticausal recursion: The initialization of (36) requires computing $Q(N-1)$ and $Q(N-2)$. Using the impulse response $s_1(-n)$ of $S_1(z^{-1})^{-1}$, we obtain:

$$Q(N-1) = \frac{b}{\lambda} \sum_{n \in \mathbb{Z}} s_1\big(-(N-1-n)\big) \tilde{Q}(n),$$
$$Q(N-2) = \frac{b}{\lambda} \sum_{n \in \mathbb{Z}} s_1\big(-(N-2-n)\big) \tilde{Q}(n),$$

then, using the N-periodicity of $\tilde{Q}(n)$ (closed contour)

$$\begin{aligned} Q(N-1) &= \frac{b}{\lambda} \sum_{n_0=0}^{N-1} \tilde{Q}(n_0) \sum_{n \in \mathbb{Z}} s_1(n_0 + 1 + nN), \\ Q(N-2) &= \frac{b}{\lambda} \sum_{n_0=0}^{N-1} \tilde{Q}(n_0) \sum_{n \in \mathbb{Z}} s_1(n_0 + 2 + nN), \end{aligned} \tag{41}$$

By substituting the expressions found in (39 and (40) in these equations, (41) provides the initial conditions to the recursion (36).

To summarize the process described above, given data points $\{P(n)\}_{n=0\ldots N-1}$, we cascade a causal and an anti-causal filter to compute the cubic smoothing spline coefficients $\{Q(n)\}_{n=0\ldots N-1}$. As for interpolating spline, the recursive algorithm for smoothing spline is stable numerically, fast and easy to implement.

APPENDIX III
FLOW-CHARTS OF THE ALGORITHMS

*A. Cubic spline interpolation. See **Fig.13***

*B. Cubic Smoothing Spline approximation. See **Fig.14***

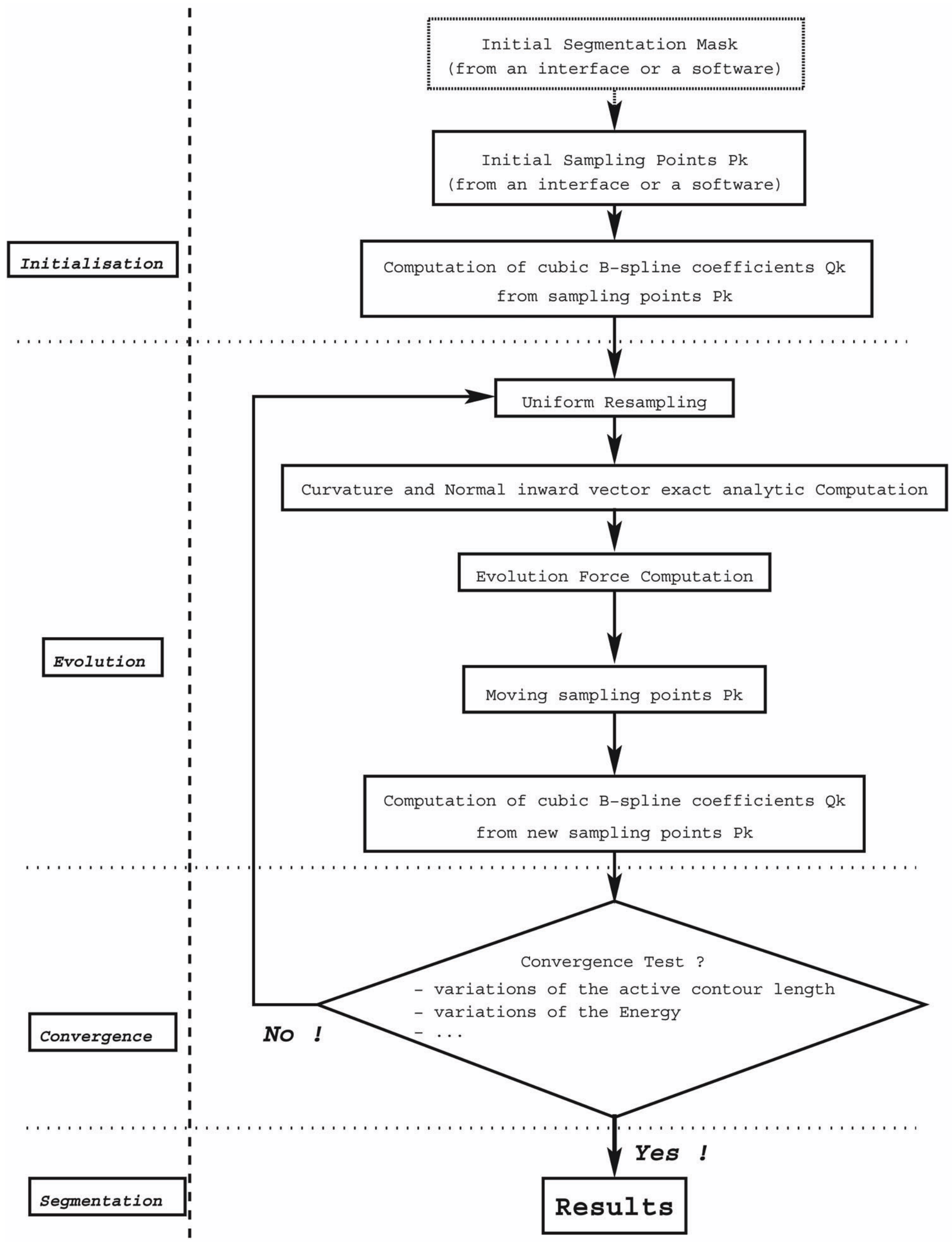


Fig. 13. Flow-chart of our Algorithm based on cubic spline interpolation

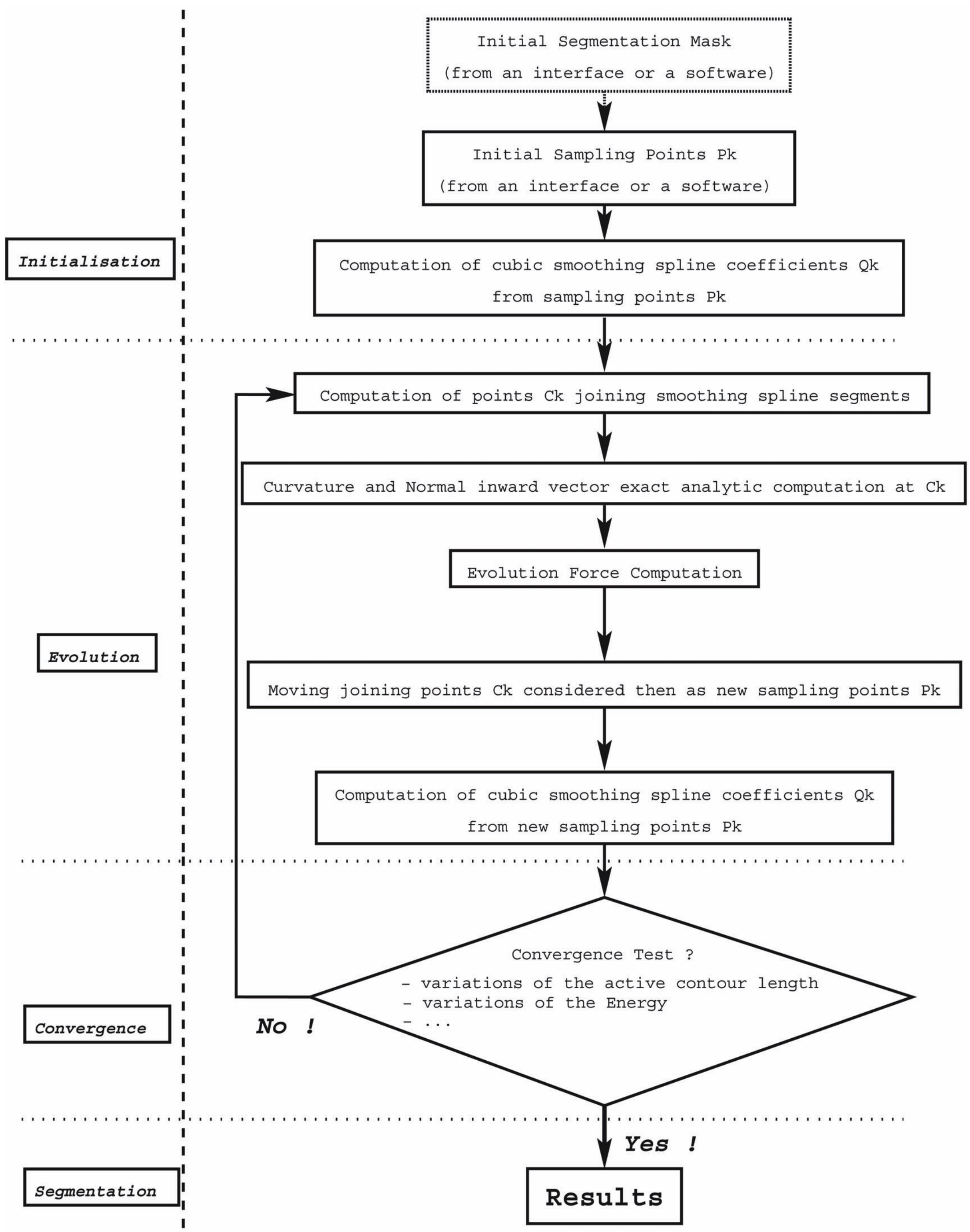


Fig. 14. Flow-chart of our Algorithm based on smoothing spline approximation