



**HAL**  
open science

# Using the shape gradient for active contour segmentation: From the continuous to the discrete formulation

Eric Debreuve, Muriel Gastaud, Michel Barlaud, Gilles Aubert

► **To cite this version:**

Eric Debreuve, Muriel Gastaud, Michel Barlaud, Gilles Aubert. Using the shape gradient for active contour segmentation: From the continuous to the discrete formulation. *Journal of Mathematical Imaging and Vision*, 2007, 28 (1), pp.47-66. 10.1007/s10851-007-0012-y . hal-00330547

**HAL Id: hal-00330547**

**<https://hal.science/hal-00330547>**

Submitted on 2 Apr 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Using the shape gradient for active contour segmentation: from the continuous to the discrete formulation

É. Debreuve\*      M. Gastaud\*      M. Barlaud\*      G. Aubert†

PREPRINT – Published in *Journal of Mathematical Imaging and Vision*

## Abstract

A variational approach to image or video segmentation consists in defining an energy depending on local or global image characteristics, the minimum of which being reached for objects of interest. This study focuses on energies written as an integral on a domain of a function which can depend on this domain. The derivative of the energy with respect to the domain, the so-called shape derivative, is a function of a velocity field applied to the domain boundary. For a given, non-optimal domain, the velocity should be chosen such that the shape derivative is negative, thus indicating a way to deform the domain in order to decrease its energy. Minimizing the energy through an iterative deformation process is known as the active contour method. In the continuous framework, setting the velocity to the opposite of the gradient associated with the  $L^2$  inner product is a common practice. In this paper, it is noted that the negativity of the shape derivative is not preserved, in general, by the discretization of this velocity required by implementation. In order to guarantee that the negativity condition holds in the discrete framework, it is proposed to choose the velocity as a linear combination of pre-defined velocities. This approach also gives more flexibility to the active contour process by allowing to introduce some *a priori* knowledge about the optimal domain. Some experimental results illustrate the differences between the classical and the proposed approach.

**Keywords:** Segmentation, shape optimization, shape derivative, active contour, discrete formulation, variational approach.

## 1 Introduction

The shape gradient approach is presented in Sections 1 and 2 and Appendices A and B. Apart from presenting the context, these sections emphasize the fact that

this approach can be seen as a general framework for boundary-based and region-based segmentation. They also provide developments already published by the authors in different articles. The proposed study actually begins with Section 3.

### 1.1 A variational approach to segmentation

Image or video segmentation can be performed with the following variational approach: the problem is formulated as the minimization of an energy depending on assumed characteristics of objects of interest [5, 8, 10, 23, 25, 26, 29]. For simplicity, it is supposed that there is a unique object. Typically, the energy is a sum of a domain integral and a boundary integral [16, 29, 31]

$$E(\Gamma) = \int_{\Omega} \phi_f(\Gamma, x) \, dx + \int_{\Gamma} \varphi_f(s) \, ds \quad (1)$$

where  $\Omega$  is an open set of  $\mathbb{R}^2$ ,  $\Gamma$  is the oriented boundary  $\partial\Omega$  of  $\Omega$ ,  $s$  is the arc-length parameterization of  $\Gamma$ , and  $f$  is the image or video to be segmented<sup>1</sup>. Energy (1) is designed to have a unique global minimum<sup>2</sup> at  $\Omega^*$ , the domain of the object of interest. For an *ideal* object, if  $\phi$  is equal to zero,  $\phi$  is sometimes referred to as the descriptor of the object. For instance, if  $\phi$  is equal to

$$\phi(\Gamma, x) = (f(x) - \mu(\Gamma))^2 \quad (2)$$

where  $\mu(\Gamma)$  is the average value of  $f$  in  $\Omega$ , then  $\phi$  is equal to zero on  $\Omega$  if and only if  $f$  is constant on  $\Omega$ . Therefore,  $\phi$  is a descriptor of objects of constant intensity. Similarly,  $\varphi$  is the descriptor of the object boundary.

If the object background can also be characterized, then the energy can be *symmetrized* as follows

$$E(\Gamma) = \int_{\Omega} \phi(\Gamma, x) \, dx + \int_{\Gamma} \varphi(s) \, ds + \int_{\Omega^c} \phi^c(\Gamma^c, x) \, dx \quad (3)$$

\*Laboratoire I3S, UMR CNRS 6070, Sophia Antipolis, France (debreuve, gastaud, barlaud@i3s.unice.fr).

†Laboratoire Dieudonné, UMR CNRS 6621, Nice, France (gaubert@math.unice.fr).

<sup>1</sup>For convenience,  $\phi$  and  $\varphi$  will be used instead of  $\phi_f$  and  $\varphi_f$ .

<sup>2</sup>A problem of maximization is trivially turned into a problem of minimization.

where  $\Omega^c$  is the complement  $D \setminus \bar{\Omega}$  of  $\Omega$  in image or frame domain  $D$ . This combination of an integral on  $\Omega$  and an integral on  $\Omega^c$  is called region competition [9, 11, 41]. Nevertheless, in the following developments, the last integral of (3) will be discarded since its handling is similar to the one of the first integral.

## 1.2 Active contour

A possible approach to minimize energy (1) is to set an initial contour and to deform it iteratively in such a way that its energy decreases from one iteration to the next. The contour eventually converges toward a (possibly local) minimizer. This process is known as active contour [4, 5, 6, 24]. When  $\phi$  does not depend on  $\Gamma$ , the minimization procedure can be based on a local strategy involving energy evaluations only [17]. Otherwise, the influences of local deformations are linked together and the contour deformation should be performed at once. The appropriate deformation can be derived from the derivative of the energy with respect to the contour [4, 5, 24].

## 1.3 Shape derivative and evolution equation

The derivative of energy (1) with respect to  $\Gamma$  can be obtained by a calculus of variations [2, 5]. However, if descriptors  $\phi$  or  $\varphi$  depends on  $\Gamma$ , this can be complex. Some studies on shape gradients [13, 23, 35, 36] offer a convenient and general basis for this differentiation. The shape derivative of (1) is a function  $dE(\Gamma, V)$  of  $\Gamma$  and a velocity  $V$  defined on  $\Omega$  but applied to  $\Gamma$  only. For a given  $\Gamma$ , the velocity should be chosen such that the shape derivative is negative, thus indicating a way to deform  $\Gamma$  in order to decrease its energy. As noted by [7], the straightforward choice is to take the opposite of the gradient of (1) associated with the  $L^2$  inner product on  $\Gamma$ . However, one might want to use other descent directions, for example, to improve the convergence rate [19], to increase spatial coherence and avoid as much as possible to converge to irrelevant local minima [7], to simplify implementation [30, 37], or to respect the principles of an underlying physical model or for improved stability and convergence rate [15]. These alternatives may result from (i) designing other inner products or from (ii) directly designing descent directions  $V$  verifying [30, 37]

$$dE(\Gamma, V) \leq 0. \quad (4)$$

In any case, the active contour evolution equation has the form

$$\frac{\partial \Gamma}{\partial \tau} = V(\tau). \quad (5)$$

When implementing this evolution equation, the negativity of the shape derivative is not preserved, in gen-

eral, by the discretization of velocity  $V$ . This paper proposes to select other descent directions allowing to solve this problem in the context of (ii). Moreover, the proposed approach offers the possibility to introduce some *a priori* knowledge about the optimal contour.

Section 2 reminds some results about the shape derivative. Section 3 describes the above mentioned discretization error corresponding to the so-called direct approach. It proposes a solution, called the constrained approach, based on choosing  $V$  as a linear combination of pre-defined velocities. Section 4 provides additional remarks about the constrained approach. Section 5 presents some experimental results to illustrate the differences between the direct approach and the constrained approach, and to illustrate how the constrained approach allows to introduce some *a priori* knowledge about the solution of (5).

## 1.4 Notations

In the following,  $D$  is a subset of  $\mathbb{R}^2$  and the image to segment,  $f$ , is a function from  $D$  to  $\mathbb{R}^m$ . Domain  $\Omega$  is an open set of  $D$ .  $\Gamma$  is the oriented boundary  $\partial\Omega$  of  $\Omega$ , and  $s$  is the arc-length parameterization of  $\Gamma$ . For convenience, the notation  $a(s)$  refers to  $a(\Gamma(s))$ . Samples on  $\Gamma$  are denoted by  $\Gamma_i$ ,  $i \in [1, n]$ . Arc-length  $s_i$  is such that  $\Gamma(s_i) = \Gamma_i$ . Then, both notations  $a(s_i)$  and  $a(\Gamma_i)$  are equivalent. Note that  $s_1$  is equal to 0 and  $s_n$  is equal to  $L - (s_{n+1} - s_n)$  where  $L$  is the length of  $\Gamma$  and  $\Gamma(s_{n+1}) = \Gamma(s_1)$ . The contour segment between  $\Gamma_i$  and  $\Gamma_{i+1}$  (a line segment, a spline segment... depending on the contour representation) is denoted by  $\gamma_i$ .

The  $L^2$  inner product on the space of velocities will be denoted

$$\langle U, V \rangle = \int_{\Gamma} U(s) \cdot V(s) \, ds \quad (6)$$

where  $\cdot$  is the dot product.

## 2 Expression of the shape derivative

### 2.1 General expression

Let us consider the typical energy

$$E(\Gamma) = \int_{\Omega} \phi(\Gamma, x) \, dx + \int_{\Gamma} \varphi(s) \, ds. \quad (7)$$

Energy (7) is a function of a contour and the set of contours is not a vector space. As a consequence, the derivative of (7) with respect to  $\Gamma$  cannot be expressed in the usual way. Let  $\Omega(\tau)$ ,  $\tau \geq 0$ , be a family of

domains such that  $\Omega(\tau = 0) = \Omega$ . Then, when  $\tau$  increases,  $\Gamma(\tau)$  can be considered as a deforming interface in a medium characterized by  $\phi$  and  $\varphi$ . Hence, a development inspired by works in continuum mechanics [18] can be made in order to determine the derivative of (7) with respect to  $\tau$  at  $\tau$  equal to zero [11]. The study of such energies and their variations was further developed in the framework of shape optimization [13, 19, 23, 35, 36]. In this context, the following expression is known as the shape derivative of (7)

$$dE(\Gamma, V) = \int_{\Omega} \frac{\partial \phi(\Gamma(\tau), x)}{\partial \tau} \Big|_{\tau=0} dx - \int_{\Gamma} \left( \phi(\Gamma, s) - \frac{\partial \varphi(s)}{\partial N} + \varphi(s) \kappa(s) \right) N(s) \cdot V(s) ds \quad (8)$$

where vector field  $V$  is by definition a velocity defined on  $\Omega$ ,  $N$  is the inward unit normal of  $\Gamma$ , and  $\kappa$  is the curvature of  $\Gamma$  (some details are given in Appendices A and B).

## 2.2 Expression with only a boundary integral

Under weak assumptions, the shape derivative of the domain integral in (7) has an equivalent expression in the form of a boundary integral [36, 37]. As is clear from Section 3.1.1, such an expression is convenient in the active contour framework. Indeed, the evolution equation of an active contour can be easily deduced from a boundary integral-only expression.

In the following, it is assumed that the shape derivative of (7) has been rewritten into a boundary integral

$$dE(\Gamma, V) = - \int_{\Gamma} \Psi(\Gamma, s) N(s) \cdot V(s) ds, \quad (9)$$

either because one of the two conditions (87) or (104) applies (see Appendix C), or as the result of another development.

## 3 From the continuous to the discrete framework

In this section, the classical, or direct, approach to using the shape derivative in the active contour framework is presented. It corresponds to choosing the gradient associated with the  $L^2$  inner product as the descent direction (among the velocities that ensure the negativity of the shape derivative) and to discretizing it. It will be noted that this direct approach implies a discretization error possibly responsible for the loss of the negativity condition. A constrained approach will be proposed to solve this problem. The constraint is specified in terms of a linear combination of pre-defined velocities. It will be shown that the negativity

condition still holds after discretization when choosing the pre-defined velocities in accordance with the active contour representation.

### 3.1 Direct approach

#### 3.1.1 Negativity of the shape derivative

Shape derivative (9) is a function of a velocity field  $V$ . Since energy (7) must be minimized, it is necessary to choose  $V$  such that (9) is negative. In other words, the following inequality must be solved for  $V$

$$dE(\Gamma, V) \leq 0. \quad (10)$$

Interpreting (9) as the  $L^2$  inner product on the space of velocities, the velocity

$$G = -\Psi N \quad (11)$$

can be identified with the gradient associated with this inner product. It is called the shape gradient of (7) [7, 13, 19, 36]. Then, taking a steepest descent approach, one can think of choosing [11, 23]

$$V(s) = -G(\Gamma, s) \quad (12)$$

in the following evolution equation

$$\frac{\partial \Gamma}{\partial \tau} = V(\tau). \quad (13)$$

#### 3.1.2 Minimization in the continuous framework

The implementation of evolution equation (13) can be based on a finite difference approximation of the derivative with respect to  $\tau$  verifying the CFL-condition. Instead, a line search strategy can be used [19]

$$\begin{cases} \Gamma^0 \\ \Gamma^{+1} = \Gamma + \alpha V \end{cases} \quad (14)$$

where  $\Gamma^0$  is an initial contour, superscript  $+1$  represents the next element of a sequence<sup>3</sup>,  $\alpha$  is a positive constant, and  $V$  is given by (12). Note that  $V$  is defined on  $\Omega$  (see Appendix B) but given on  $\Gamma$  only. However, as is clear from (14), it does not need to be known anywhere else. The optimal value for  $\alpha$  can be computed as follows

$$\alpha = \arg \min_{\alpha \geq 0} E(\Gamma^{+1}(\alpha)). \quad (15)$$

<sup>3</sup>The sequence  $x(n+1) = f(x(n))$ ,  $x(0) = x^0$  is denoted by  $x^{+1} = f(x)$ ,  $x^0$ .

### 3.1.3 Discretization and induced velocity

In practice, active contour  $\Gamma$  is sampled. For instance, it can be represented by a polygon  $\{\Gamma_i, i \in [1, n]\}$  without self-intersection [17]. The corresponding discrete version of evolution equation (14) is

$$\begin{cases} \Gamma_i^0 \\ \Gamma_i^{+1} = \Gamma_i + \alpha V(\Gamma_i) \end{cases} \quad (16)$$

where  $\{\Gamma_i^0\}$  is an initial polygon. Note that (16) does not make use of  $V$  along the edges  $\gamma_i$  of the polygon. Instead, it implicitly defines a velocity  $\tilde{V}$ , called induced velocity, transforming edges  $\gamma_i$  into edges  $\gamma_i^{+1}$ . However, it is unlikely that such a transformation corresponds to the chosen velocity<sup>4</sup> (12) (see Fig. 1a). As

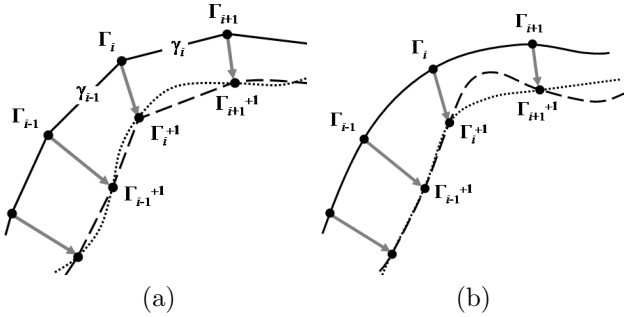


Figure 1: Incorrect deformation due to the sampling of  $V$ . (a) *Disks*: polygon vertices; *Solid line*: polygon before deformation; *Dashed line*: polygon defined by the translated vertices; *Dotted line*: polygon deformed according to  $V$ . (b) *Disks*: curve samples; *Solid line*: curve before deformation; *Dashed line*: curve interpolating the translated samples; *Dotted line*: curve deformed according to  $V$ .

a consequence, the negativity of shape derivative (9) at  $\tilde{V}$  is not guaranteed. In other words, the discrete evolution equation (16) might not generate a minimizing sequence of contours.

This problem is not specific to the polygonal representation. It also arises if the contour is represented by a smooth curve, say, a spline, since it is due to the sampling of  $V$  (see Fig. 1b). Generally speaking, only these contour samples where  $V$  is computed are translated correctly. The contour segments in-between the translated samples are defined *a posteriori* by the chosen interpolation model (a polygon, a uniform cubic B-spline...).

However, as pointed out by a reviewer, if the edge length is small enough,  $dE(\Gamma, \tilde{V})$  is still negative and this direct approach is valid (see Appendix D). The proposed edge length upper bound is global and depends on maximal variations of  $\Psi$  over  $\Gamma$ . It could also

<sup>4</sup>In particular, (12) does certainly not transform, in general, a polygon into another polygon.

be adapted locally to be low in portions of  $\Gamma$  where  $\Psi$  varies a lot and higher where  $\Psi$  varies slowly [38].

Finally, note that (16) might *converge too early* since the condition  $\{V(\Gamma_i) = 0, i \in [1, n]\}$  is less restrictive than  $\{V(s) = 0, s\}$ . But again, the smaller the edge length, the less critical. Nevertheless, instead of a condition on the edge length, one can wonder if there is a way to choose  $V$  such that  $\tilde{V}$  is equal to  $V$ .

## 3.2 Constrained approach

### 3.2.1 Negativity of the shape gradient

In order to guarantee that, after discretization, the induced velocity  $\tilde{V}$  matches the original velocity  $V$ , we propose to restrict, beforehand in the continuous domain, the domain transformations<sup>5</sup> to a linear combination of a set of pre-defined transformations

$$T(\tau) = \sum_i \beta_i T_i(\tau), \quad \beta_i \in \mathbb{R}. \quad (17)$$

The differentiation of (17) with respect to  $\tau$  leads to

$$V = \sum_i \beta_i V_i \quad (18)$$

where  $V_i$  is a so-called pre-defined velocity. In this context, velocity  $V$  will be given by a choice of  $\beta_i$ 's that satisfies the negativity of shape derivative (9).

The shape derivative can be rewritten as

$$dE(\Gamma, V) = dE(\Gamma, \sum_i \beta_i V_i) \quad (19)$$

$$= \sum_i \beta_i dE(\Gamma, V_i) \quad (20)$$

$$= \beta \cdot d\vec{E}(\Gamma, V_i) \quad (21)$$

where  $\cdot$  is the dot product,  $\beta$  is the vector of components  $\beta_i$ , and  $d\vec{E}(\Gamma, V_i)$  is the vector of components  $dE(\Gamma, V_i)$ . Taking a steepest descent approach,  $\beta$  should be such that  $dE(\Gamma, V)$  is as negative as possible. The Cauchy-Schwarz inequality implies that

$$|\beta \cdot d\vec{E}(\Gamma, V_i)| \leq |\beta| |d\vec{E}(\Gamma, V_i)| \quad (22)$$

and, for a given norm of  $\beta$ ,  $dE(\Gamma, V)$  is as large as possible in absolute value when  $\beta$  and  $d\vec{E}(\Gamma, V_i)$  are linearly dependent. Therefore, the  $\beta_i$ 's are taken equal to

$$\beta_i = -\alpha dE(\Gamma, V_i) \quad (23)$$

where  $\alpha$  is a positive constant.

Section 3.2.3 presents a general purpose definition of pre-defined velocities and an application-driven variation of it.

<sup>5</sup>For definitions and notations, see Appendix B.

### 3.2.2 Minimization in the continuous framework and discretization

In the continuous framework, a possible minimizing sequence is

$$\begin{cases} \Gamma^0 \\ \Gamma^{+1} = \Gamma - \alpha \sum_i dE(\Gamma, V_i) V_i \end{cases} \quad (24)$$

where the optimal value for  $\alpha$  can be computed as follows

$$\alpha = \arg \min_{\alpha \geq 0} E(\Gamma^{+1}(\alpha)) . \quad (25)$$

With the polygonal representation mentioned in Section 3.1.3, the discretization of (24) leads to the following evolution equation

$$\begin{cases} \Gamma_i^0 \\ \Gamma_i^{+1} = \Gamma_i - \alpha \sum_j dE(\Gamma, V_j) V_j(\Gamma_i) \end{cases} . \quad (26)$$

However, for the pre-defined velocities proposed in Section 3.2.3, the sum over the pre-defined velocities reduces to a single term since at  $\Gamma_i$ , only  $V_i$  is not equal to zero

$$\begin{cases} \Gamma_i^0 \\ \Gamma_i^{+1} = \Gamma_i - \alpha dE(\Gamma, V_i) V_i(\Gamma_i) \end{cases} . \quad (27)$$

### 3.2.3 Examples of pre-defined velocities for the polygonal representation

According to the remarks of Section 3.1.3, it can be deduced that the pre-defined velocities must be consistent with the contour representation: the representation must be preserved when the contour is deformed by a linear combination of the  $V_i$ 's. Moreover, ideally, the  $V_i$ 's should be able to generate any velocity  $V$  by linear combination (at least, if there is no *a priori* knowledge about the optimal contour). There is no such basis of velocities. However, the  $V_i$ 's must allow to represent a reasonable variety of velocities. If the contour is represented by a polygon with  $n$  vertices, the following definitions can be considered.

(i) At vertex  $\Gamma_i$ , a pseudo-normal  $N(\Gamma_i)$  is defined [14, 27] and  $V_i$  is the velocity collinear to  $N(\Gamma_i)$  at  $\Gamma_i$  and transforming  $(\Gamma_{i-1}, \Gamma_i, \Gamma_{i+1})$  into  $(\Gamma_{i-1}, (\Gamma_i + N(\Gamma_i)), \Gamma_{i+1})$  (see Fig. 2). In other words,  $V_i$  is a vector field with support  $[\Gamma_{i-1}, \Gamma_{i+1}]$ , linear from zero (at  $\Gamma_{i-1}$ ) to  $N(\Gamma_i)$  (at  $\Gamma_i$ ), and linear again back to zero (at  $\Gamma_{i+1}$ ). This definition involves no *a priori* knowledge (see Section 5.1 for an example of application).

(ii) The previous definition can be modified to introduce some *a priori* knowledge. For instance, in tracking, the approximate motion of the tracked object might be known. In particular, a joint segmentation and motion computation method [10] certainly requires computation of the motion for a given, fixed segmentation (and vice-versa). Therefore, if  $m_i$  is the

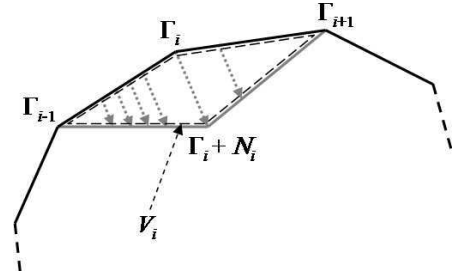


Figure 2: A possible choice for the pre-defined velocity  $V_i$ .

estimated motion of  $\Gamma_i$ , then  $V_i$  can be defined similarly to (i) by replacing  $N(\Gamma_i)$  with  $m_i$  or  $m_i/|m_i|$  (see Section 5.2 for an example of application).

### 3.2.4 Coherence between the continuous and discrete evolutions

In order to establish that (24) and (27) lead to identical evolutions, it must be shown that the deformations induced by (27) on contour segments  $\gamma_i$  are equal to the deformations of the continuous evolution (24). This would also prove that the negativity of the shape derivative is preserved in the discrete framework. Indeed, if both evolutions are identical, the induced velocity  $\tilde{V}$  matches the constrained velocity  $V$ .

**Polygonal representation** Let  $x$  be a point on edge  $\gamma_i$  (see Fig. 3). There exists  $t \in [0, 1]$  such that

$$x = \gamma_i(t) = (1-t) \Gamma_i + t \Gamma_{i+1} . \quad (28)$$

Let  $V_i$  be a pre-defined velocity as defined in Sec-

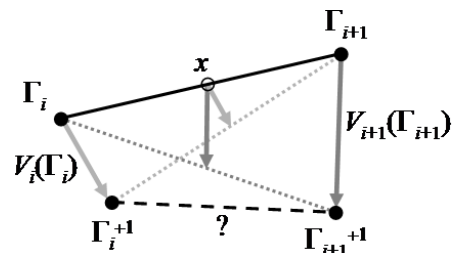


Figure 3: Transformation of point  $x$  on edge  $\gamma_i$ .

tion 3.2.3 (either of the two definitions). The velocity  $V(t)$  at  $\gamma_i(t)$  is the combination of two velocities

$$V(t) = \beta_i V_i(t) + \beta_{i+1} V_{i+1}(t) . \quad (29)$$

Therefore, in the continuous framework, point  $\gamma_i(t)$  is translated to

$$\gamma_i(t) + V(t) = (1-t) \Gamma_i + \beta_i V_i(t) + t \Gamma_{i+1} + \beta_{i+1} V_{i+1}(t) . \quad (30)$$

The pre-defined velocities are such that

$$\begin{cases} V_i(t) = (1-t) V_i(\Gamma_i) \\ V_{i+1}(t) = t V_{i+1}(\Gamma_{i+1}) \end{cases} . \quad (31)$$

Combining (30) and (31) leads to

$$\begin{aligned} \gamma_i(t) + V(t) &= (1-t) (\Gamma_i + \beta_i V_i(\Gamma_i)) \\ &\quad + t (\Gamma_{i+1} + \beta_{i+1} V_{i+1}(\Gamma_{i+1})) \quad (32) \\ &= (1-t) \Gamma_i^{+1} + t \Gamma_{i+1}^{+1}, \quad (33) \end{aligned}$$

meaning that the translation of point  $\gamma_i(t)$  in the continuous framework (left-hand side) is a point on line segment  $[\Gamma_i^{+1}, \Gamma_{i+1}^{+1}]$  obtained by joining the *discrete* translations of  $\Gamma_i$  and  $\Gamma_{i+1}$  (right-hand side). Moreover, when  $\gamma_i(t)$  describes edge  $\gamma_i$ , then the translation of  $\gamma_i(t)$  describes the whole line segment. As a conclusion, the continuous equation (24) and the discrete equation (27) lead to identical evolutions.

**Uniform cubic B-spline representation** If the contour is represented by a uniform cubic B-spline with  $n$  segments [3, 21, 32, 40] (see Appendix E for the notations and properties used here), then one can think of basing the pre-defined velocities on blending function  $B$  (see Fig. 9). For example,

$$V_i(t) = B_i^i(t) N(\Gamma_i), \quad t \in [-n/2, n/2] \quad (34)$$

where  $B_i^i$  represents blending function  $B$  shifted to be “centered” on  $\Gamma_i$  and “seen” from spline segment  $\gamma_i$ . More specifically,

$$\begin{cases} B_i^i(\Gamma_i) = B(0) \\ B_i^i(\Gamma_{i-1}) = B_i^i(\Gamma_{i+1}) = B(1) \\ B_i^i(\Gamma_{i-2}) = B_i^i(\Gamma_{i+2}) = B(2) \end{cases} \quad (35)$$

and  $\Gamma_j$  corresponds to  $t = j - i$ . Therefore, velocity  $V$  on spline segment  $\gamma_i$  parameterized as described above is equal to

$$V(t) = \sum_j \beta_j V_j(t) \quad (36)$$

$$= \sum_{j=i-1}^{i+2} \beta_j V_j(t) \quad (37)$$

$$= \sum_{j=i-1}^{i+2} \beta_j B_j^i(t) N(\Gamma_j) . \quad (38)$$

Moreover,  $\gamma_i$  is defined by

$$\gamma_i(t) = \sum_{j=i-1}^{i+2} q_j B_j^i(t), \quad t \in [0, 1] . \quad (39)$$

Therefore, after evolution, point  $\gamma_i(t)$  is translated to

$$\gamma_i^{+1}(t) = \gamma_i(t) + V(t) \quad (40)$$

$$= \sum_{j=i-1}^{i+2} \underbrace{(q_j + \beta_j N(\Gamma_j))}_{q_j^{+1}} B_j^i(t) . \quad (41)$$

Equation (41) represents a spline uniquely defined by the new control points  $q_i^{+1}$ . It must be checked whether the sampling points  $\Gamma_i^{+1}$  this spline interpolates are also obtained by the discrete evolution equation. Since (41) represents a closed, uniform cubic B-spline, the following property holds

$$\Gamma_i^{+1} = \frac{1}{6}(q_{i-1}^{+1} + 4q_i^{+1} + q_{i+1}^{+1}) \quad (42)$$

$$= \frac{1}{6}[(q_{i-1} + 4q_i + q_{i+1}) + \beta_{i-1}N(\Gamma_{i-1}) + 4\beta_iN(\Gamma_i) + \beta_{i+1}N(\Gamma_{i+1})] \quad (43)$$

$$= \Gamma_i + \beta_{i-1}B_{i-1}^i(0)N(\Gamma_{i-1}) + \beta_iB_i^i(0)N(\Gamma_i) + \beta_{i+1}B_{i+1}^i(0)N(\Gamma_{i+1}) \quad (44)$$

$$= \Gamma_i + \sum_{j=i-1}^{i+1} \beta_j B_j^i(0) N(\Gamma_j) \quad (45)$$

Noting that  $B_{i+2}^i(0)$  is equal to zero, (45) is equivalent to

$$\Gamma_i^{+1} = \Gamma_i + \sum_{j=i-1}^{i+2} \beta_j B_j^i(0) N(\Gamma_j) \quad (46)$$

$$= \Gamma_i + V(0) \quad (47)$$

which is the discrete evolution equation (26). Therefore, this spline representation associated with appropriate pre-defined velocities guarantees that the discrete evolution matches the continuous evolution (24).

## 4 Some remarks about the pre-defined velocities

### 4.1 Normalization

The weight of pre-defined velocity  $V_i$  is equal to  $-\alpha dE(\Gamma, V_i)$ . The weight of  $k$   $V_i$  is  $k$  times larger. Therefore, a pre-defined velocity can be made artificially preponderant in evolution process (26). As a consequence, it seems appropriate to normalize the pre-defined velocities (unless otherwise imposed by some *a priori*). Actually, they are defined up to a multiplicative constant. Their norm does not need to be equal to one. It should only be the same for all  $V_i$ 's. The pre-defined velocities proposed in Section 3.2.3 are such that

$$|V_i|^2 = \langle V_i, V_i \rangle = \frac{2l}{3} \quad (48)$$

where  $l$  is the edge length.

### 4.2 Interpretation as a projection

#### 4.2.1 Individual projections and redundancy

Weights (23) can be interpreted as individual projections of the opposite of gradient  $G$  (see Eq. (11)) onto

the  $V_i$ 's

$$\beta_i = \langle -G, V_i \rangle, \quad (49)$$

up to a multiplicative constant.

One can check that the pre-defined velocities proposed in Section 3.2.3 do not form an orthogonal set relative to the  $L^2$  inner product

$$\langle V_i, V_j \rangle = \begin{cases} \frac{2l}{3} & \text{if } i = j \\ \frac{1}{6} N(\Gamma_i) \cdot N(\Gamma_j) & \text{if } |i - j| = 1 \\ 0 & \text{otherwise} \end{cases} \quad (50)$$

where  $N(\Gamma_i) \cdot N(\Gamma_j)$  is equal to the cosine of the angle between  $N(\Gamma_i)$  and  $N(\Gamma_j)$ . In other words, these pre-defined velocities are redundant.

Let us consider three successive vertices  $\Gamma_{i-1}$ ,  $\Gamma_i$ , and  $\Gamma_{i+1}$  and their respective pre-defined velocities. Suppose that  $G$  is equal to  $V_i$ . If the pre-defined velocities did form an orthogonal set, the velocity  $V$  applied to  $\Gamma$  would be proportional to  $-V_i$ , as expected. Instead, the pre-defined velocity weights are equal to (up to a multiplicative constant)  $-N(\Gamma_{i-1}) \cdot N(\Gamma_i)$ ,  $-4$ , and  $-N(\Gamma_i) \cdot N(\Gamma_{i+1})$ , respectively. It means that all three pre-defined velocities get weights different from zero (in general) with a higher weight for  $V_i$  (in absolute value). If the dot products are positive (*i.e.*, if the (pseudo) curvature is not too high in absolute value), this can be interpreted as a smoothing of the response obtained in the case of an orthogonal set of pre-defined velocities, which could, as suggested in [7], increase the spatial coherence of the active contour evolution. Unfortunately, if the dot products are negative,  $V_{i-1}$  and  $V_{i+1}$  get positive weights, leading to an undesirable evolution where  $\Gamma_i$  is translated in a way that decreases the energy and the two other vertices are translated in a way that (although slightly) tends to increase the energy.

As suggested by a reviewer, the redundancy can be suppressed by generating mutually orthogonal pre-defined velocities  $V_i^\perp$  using the Schmidt orthonormalization process, or  $G$  could be projected onto the space of linear combinations of the pre-defined velocities instead of being projected individually onto each pre-defined velocity (see Section 4.2.2).

#### 4.2.2 Alternative linear combination

Instead of computing weights  $\beta_i$  as individual projections (49), they could be computed such that  $V$  is the  $L^2$  projection of  $-G$  onto the space of linear combinations of the  $V_i$ 's

$$\beta = \arg \min_{\beta} \left| -G - \sum_i \beta_i V_i \right|^2 \quad (51)$$

$$= \arg \min_{\beta} \left| \sum_i \beta_i V_i \right|^2 + 2 \sum_i \beta_i \langle G, V_i \rangle \quad (52)$$

$$= \arg \min_{\beta} \sum_i \beta_i^2 |V_i|^2 + 2 \sum_{i < j} \beta_i \beta_j \langle V_i, V_j \rangle + 2 \sum_i \beta_i \langle G, V_i \rangle \quad (53)$$

where  $\beta$  is the vector of components  $\beta_i$ .

If the  $V_i$ 's form an orthonormal set, then (53) is equivalent to

$$\beta = \arg \min_{\beta} |\beta|^2 + 2 \beta \cdot \eta \quad (54)$$

where  $\eta$  is the vector of component  $\langle G, V_i \rangle$ . For a given norm of  $\beta$ , the Cauchy-Schwarz inequality leads to the result based on individual projections  $\beta_i = -\eta_i$ , up to a multiplicative constant.

For the pre-defined velocities proposed in Section 3.2.3, if  $G$  is equal to  $V_i$ , then the projection of  $-G$  is necessarily such that  $\beta_j$  is equal to zero if  $j \notin \{i - 1, i, i + 1\}$ . Let us fix the norm of  $\beta$  to one. Using the fact that  $\beta_{i-1}^2 + \beta_i^2 + \beta_{i+1}^2$  is equal to one, one can check that the projection is unique and given by

$$\begin{cases} \beta_i = -1 \\ \beta_j = 0, j \neq i \end{cases}, \quad (55)$$

thus solving the problem of redundancy mentioned in Section 4.2.1.

The constrained approach using the projection-based weights (53) was implemented and compared with the constrained approach using the weights based on individual projections (49) (see Section 5.1).

## 4.3 Parametric approach

### 4.3.1 Presentation

Another way to avoid the problem mentioned in Section 3.1.3 is to parameterize the active contour and to rewrite energy (7) as a function of these parameters [22, 28, 39]. The problem of minimizing (7) is then restricted to the set of domains whose boundary can be described by such parameters. It can be solved in  $\mathbb{R}^n$  where  $n$  is the number of parameters. If  $\phi$  in (7) does not depend on  $\Gamma$ , then the partial derivatives of (7) with respect to each parameter can be obtained by a calculus of variations [22]. However, it can be more complex if  $\phi$  does depend on  $\Gamma$ , unless noticing that the partial derivatives can be expressed as shape derivatives [12, 28]. Let  $\Gamma$  be a curve described by a set of parameters  $p = \{p_i, i \in [1, n]\}$ , *e.g.*, a spline. Energy (7) can be rewritten as a function of  $p$

$$E(p) = \int_{\Omega(p)} \phi(\Gamma(p), x) dx + \int_{\Gamma(p)} \varphi(s) ds. \quad (56)$$

It can be shown that the gradient of (56) is equal to

$$\nabla E(p) = \sum_i dE \left( \Gamma, \frac{\partial \Gamma}{\partial p_i} \right) e_i \quad (57)$$



where  $\partial\Gamma/\partial p_i$  is a so-called admissible velocity and  $e_i$  is the  $i^{\text{th}}$  element of the canonical basis of  $\mathbb{R}^n$ . Energy (56) can be minimized using the following procedure

$$\begin{cases} p^0 \\ p^{+1} = p - \alpha \nabla E(p) \end{cases} \quad (58)$$

where the optimal value for  $\alpha$  can be computed as follows

$$\alpha = \arg \min_{\alpha \geq 0} E(p^{+1}(\alpha)) . \quad (59)$$

Section 4.3.2 shows that the constrained approach (24)/(26) is equivalent to the parametric approach (58) in certain circumstances.

### 4.3.2 Link with the constrained approach

**Polygonal representation** If the contour is represented by a polygon with  $n$  edges (see Section 3.2.3), then the parameters involved in the parametric approach are the coordinates  $(a_i, b_i)$  of each vertex  $\Gamma_i$ . Referring to the notations of Section 4.3.1, parameters  $p$  are

$$\begin{aligned} p &= \{p_j, j \in [1, 2n]\} \\ &= \{a_1, b_1, a_2, b_2, \dots, a_n, b_n\} \text{ where } (a_i, b_i) = \Gamma_i . \end{aligned} \quad (60)$$

Therefore, there are  $2n$  admissible velocities such that

$$\begin{cases} \frac{\partial\Gamma}{\partial a_i}(\Gamma_i) = e_1 \\ \frac{\partial\Gamma}{\partial b_i}(\Gamma_i) = e_2 \end{cases}, \quad i \in [1, n] \quad (61)$$

where  $(e_1, e_2)$  is the canonical basis of  $\mathbb{R}^2$ . Writing procedure (58) in terms of vertices leads to

$$\begin{aligned} \Gamma_i^{+1} &= \Gamma_i - \alpha \left[ dE \left( \Gamma, \frac{\partial\Gamma}{\partial a_i} \right) e_1 + dE \left( \Gamma, \frac{\partial\Gamma}{\partial b_i} \right) e_2 \right] \\ &= \Gamma_i - \alpha \left[ dE \left( \Gamma, \frac{\partial\Gamma}{\partial a_i} \right) \frac{\partial\Gamma}{\partial a_i}(\Gamma_i) \right. \\ &\quad \left. + dE \left( \Gamma, \frac{\partial\Gamma}{\partial b_i} \right) \frac{\partial\Gamma}{\partial b_i}(\Gamma_i) \right] \end{aligned} \quad (62)$$

which is a discrete, constrained evolution (26). Noting that  $\partial\Gamma/\partial a_i$  and  $\partial\Gamma/\partial b_i$  fit definition (ii) of the pre-defined velocities proposed in Section 3.2.3 and according to the equivalence between discrete and continuous evolutions shown in Section 3.2.4, it can be deduced that the equivalence between (26) and (58) holds in the continuous framework. In conclusion, with the polygonal representation, if the pre-defined velocities are chosen equal to the admissible velocities, then the constrained approach is equivalent to the parametric approach.

**Uniform cubic B-spline representation** If the contour is represented by a uniform cubic B-spline with  $n$  segments [3, 21, 32, 40] (see Appendix E for the notations and properties used here), then the parameters involved in the parametric approach are the coordinates  $(a_i, b_i)$  of each control point  $q_i$ . Referring to the notations of Section 4.3.1, parameters  $p$  are

$$\begin{aligned} p &= \{p_j, j \in [1, 2n]\} \\ &= \{a_1, b_1, a_2, b_2, \dots, a_n, b_n\} \text{ where } (a_i, b_i) = q_i . \end{aligned} \quad (64)$$

For clarity, the following notations will be temporarily used

$$\begin{cases} dE_a^i = dE \left( \Gamma, \frac{\partial\Gamma}{\partial a_i} \right) \\ dE_b^i = dE \left( \Gamma, \frac{\partial\Gamma}{\partial b_i} \right) \\ dE_{a,b}^i = (dE_a^i \quad dE_b^i)^T \end{cases} \quad (65)$$

where  $A^T$  is the transpose of matrix  $A$ . According to (64), procedure (58) can be rewritten as follows

$$\begin{cases} a_i^{+1} = a_i - \alpha dE_a^i \\ b_i^{+1} = b_i - \alpha dE_b^i \end{cases}, \quad i \in [1, n] \quad (66)$$

$$\iff q_i^{+1} = q_i - \alpha dE_{a,b}^i, \quad i \in [1, n] . \quad (67)$$

Spline segment  $\gamma_i$  is defined by

$$\gamma_i(t) = P(t) Q_i, \quad t \in [0, 1] . \quad (68)$$

One can verify that, after evolution, the new segment  $\gamma_i^{+1}$  satisfies

$$\begin{aligned} \gamma_i^{+1}(t) &= P(t) Q_i^{+1} \\ &= \gamma_i(t) - \alpha \sum_{j=1}^n \left( dE \left( \Gamma, \frac{\partial\Gamma}{\partial a_j} \right) \frac{\partial\gamma_i}{\partial a_j}(t) \right. \\ &\quad \left. + dE \left( \Gamma, \frac{\partial\Gamma}{\partial b_j} \right) \frac{\partial\gamma_i}{\partial b_j}(t) \right) \end{aligned} \quad (69)$$

which is (locally on segment  $\gamma_i$ ) the continuous, constrained evolution (24). In conclusion, with the uniform cubic B-spline representation, if the pre-defined velocities are chosen equal to the admissible velocities, then the constrained approach is equivalent to the parametric approach.

**About redundancy** The admissible velocities are not necessarily mutually orthogonal. In particular, for the polygonal representation, they are not. Yet, these velocities arise in the computation of the gradient of energy (56), a function from  $\mathbb{R}^n$  to  $\mathbb{R}$ . In this context, the notion of gradient is classical. This might make thinking that redundancy is not always an undesirable property.

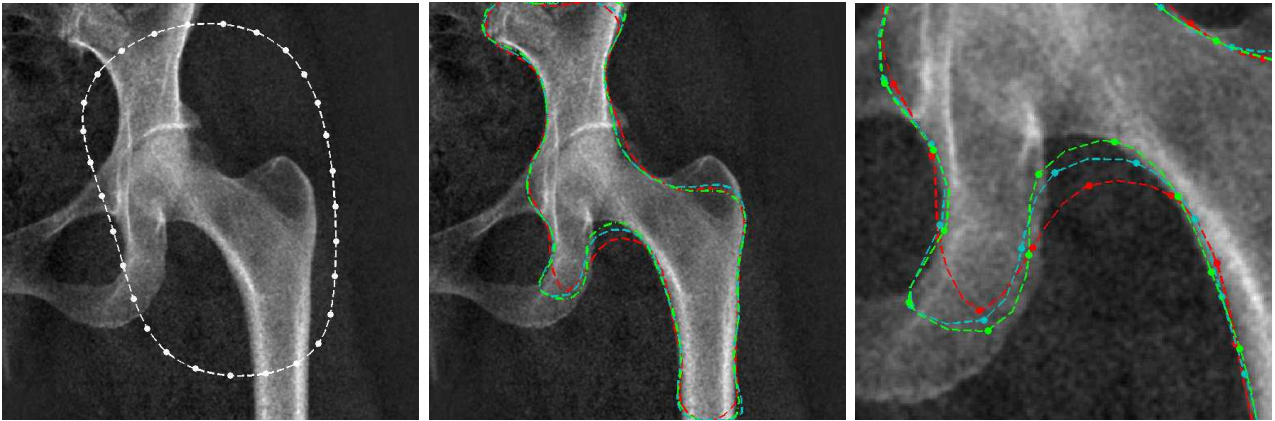


Figure 4: Direct/constrained segmentation comparison at a 40-pixel resolution (purposely low to emphasize the difference between the two approaches): initial contour (white plot), result with the direct approach (red plot), with the constrained approach using normal-based pre-defined velocities weighted according to Section 3.2.1 (cyan plot), and with weights computed as suggested in Section 4.2.2 (green plot).

## 5 Experimental results

### 5.1 Direct vs. constrained approach

A  $512 \times 512$ -pixel image was segmented using the direct then the constrained approach with the same initial contour in a region-competition context (see Fig. 4). Object descriptor (2) was used for both the object and the background and no boundary descriptor was defined. The corresponding energy is

$$E^c(\Gamma) = \int_{\Omega} (f(x) - \mu(\Gamma))^2 dx + \int_{\Omega^c} (f(x) - \mu(\Gamma^c))^2 dx \quad (71)$$

and its shape derivative is equal to

$$dE^c(\Gamma, V) = - \int_{\Gamma} ((f(x) - \mu(\Gamma))^2 - (f(x) - \mu(\Gamma^c))^2) N(s) \cdot V(s) ds. \quad (72)$$

For the constrained approach, the normal-based pre-defined velocities (see Section 3.2.3 - definition (i)) were used so that the comparison with the direct approach based by definition on an evolution along the normals (see Section 3.1) is coherent. The active contour was sampled into a polygon with an edge length  $l$  of 40 pixels. It was periodically resampled to maintain this resolution.

The algorithm of the direct approach is described in Alg. 1. Constant  $\alpha_{\min}$  is homogeneous to a number (possibly not an integer) of pixels. It is related to the achievable accuracy of the segmentation (the lower  $\alpha_{\min}$ , the better the accuracy).

The algorithm of the constrained approach is described in Alg. 2. Note that the weights of the pre-defined velocities are computed as individual projections of the opposite of the gradient  $\langle -G, V_i \rangle$ .

The 40-pixel resolution was purposely low in order to emphasize the difference between the two approaches (which did not allow to segment the object accurately). In fact, in this example, the direct approach segmentation is fairly similar to the segmentation obtained with the constrained approach (see Fig. 4). The constrained approach performed a little bit better, though. Quantitatively, the energy at convergence for the constrained approach (requiring 52 iterations) is 6.5% lower (*i.e.*, better) than the one of the direct approach (requiring 55 iterations). If the weights of the pre-defined velocities are computed so that  $\sum_i \beta_i V_i$  is the projection of  $-G$  onto the space of the linear combinations of the pre-defined velocities (see Section 4.2.2), then the energy at convergence (now requiring 71 iterations) is 5.5% lower than the energy reached with the weights based on individual projections. Therefore, when the pre-defined velocities do not form an orthonormal set, the weights should be computed as suggested in Section 4.2.2 for maximum accuracy. In conclusion, the constrained approach can perform better than the direct approach when the contour resolution is low. However, this slight advantage would vanish at high resolution since the error mentioned in Section 3.1.3 would get negligible (see Fig. 5). Experimentally, the segmentation was performed again with a contour resolution of 16 pixels (corresponding to the same ratio with respect to the object size as the one used in Section 5.2). As expected, the results are almost identical for all three methods (see Fig. 6). Therefore, the interest of the constrained approach is not the gain in precision but rather the coherence it provides between the theory developed in the continuous framework and its discrete implementation, and the flexibility it gives to the evolution process, as illustrated in Section 5.2.

1. Choose an initial polygon  $\Gamma = \{\Gamma_i, i \in [1, n]\}$  with an edge length equal to a given resolution  $l$ ; Choose a threshold  $\alpha_{\min}$ .
2. Compute domain  $\Omega$  as the binary mask of  $\Gamma$ .
3. Compute the mean intensities  $\mu(\Gamma)$  and  $\mu(\Gamma^c)$  in  $\Omega$  and  $\Omega^c$ , respectively.
4. [Optionally, compute energy  $E^c$  to check that it decreases.]
5. Compute the  $L^2$ -gradient at  $\Gamma_i$

$$G(\Gamma, \Gamma_i) = (f(\Gamma_i) - \mu(\Gamma^c))^2 - (f(\Gamma_i) - \mu(\Gamma))^2 N(\Gamma_i). \quad (a)$$

6. Update  $\Gamma$  according to

$$\Gamma_i^{+1} = \Gamma_i - \alpha G(\Gamma, \Gamma_i) \quad (b)$$

where  $\alpha$  is computed as  $\arg \min_{\alpha \geq 0} E^c(\Gamma^{+1}(\alpha))$ .

7. If needed, resample  $\Gamma$  to maintain a resolution of  $l$ .
8. If  $\alpha$  was less than  $\alpha_{\min}$ , then the algorithm is supposed to have converged; Otherwise go back to step 2.

Algorithm 1: Direct approach algorithm for the polygonal representation.

## 5.2 An example of tracking constraint

An application where the possibility to introduce *a priori* knowledge in the evolution process can be useful is tracking. Indeed, a usual procedure is to use the segmentation of the object of interest in a frame as the initialization to segment the next frame. Then, the contour deformation necessary to segment the object of interest in the next frame is correlated to the motion of the object<sup>6</sup>. Therefore, one can think of choosing motion-based pre-defined velocities. To track a player on the standard test sequence ‘‘Football’’, the normal in the normal-based pre-defined velocities was replaced with a unit vector in the direction of the local motion (see Section 3.2.3 - definition (ii)). The choice of the energy is independent of the choice of the pre-defined velocities. However, it is natural in tracking to use an energy which involves motion [10]. Here, considering the complexity of the motion of the object of inter-

<sup>6</sup>Ideally, the motion could even transform the segmentation in a frame directly into the segmentation in the next frame. However, because the sequence is a two-dimensional projection of a three-dimensional scene and because the available motion is usually an apparent motion, this is not the case in practice.

- 1.-4. See Alg. 1.

5. Compute the pre-defined velocities  $V_i$  according to Section 3.2.3 - definition (i) and compute the shape derivatives  $dE^c(\Gamma, V_i)$ .

6. Update  $\Gamma$  according to

$$\Gamma_i^{+1} = \Gamma_i - \alpha dE^c(\Gamma, V_i) V_i(\Gamma_i) \quad (a)$$

$$= \Gamma_i - \alpha dE^c(\Gamma, V_i) N(\Gamma_i) \quad (b)$$

where  $\alpha$  is computed as  $\arg \min_{\alpha \geq 0} E^c(\Gamma^{+1}(\alpha))$ .

- 7.-8. See Alg. 1.

Algorithm 2: Constrained approach algorithm for the polygonal representation.

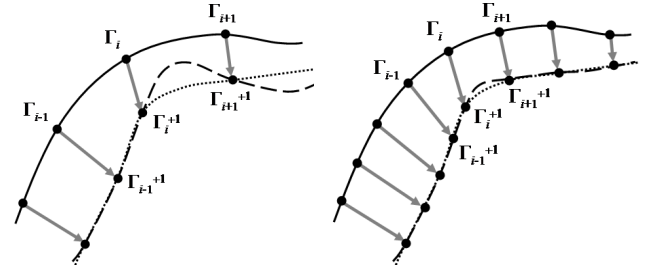


Figure 5: The error between the correctly deformed curve (dotted line) and the wrongly estimated curve (dashed line) decreases when resolution gets higher. *Disks*: curve samples; *Solid line*: curve before deformation; *Dashed line*: curve interpolating the (correctly) translated samples; *Dotted line*: curve correctly deformed.

est and the slight motion blur, it seemed appropriate to confine the motion use to the pre-defined velocities and to use a motion-free energy able to account for the color variability of the object [2]

$$E(\Gamma) = \int_{\mathbb{R}^2} k(h(\Gamma, a), h_{\text{prev}}(a)) da \quad (73)$$

where  $h$  is a smooth, normalized version of the color histogram in  $\Omega$  of the frame  $f_t$  to be segmented

$$h(\Gamma, a) = \frac{1}{|\Omega|} \int_{\Omega} g(f_t(x) - a) dx \quad (74)$$

where  $|\Omega|$  is a measure of  $\Omega$  and  $g$  is a smoothing kernel, *e.g.*, a 2-dimensional Gaussian. Similarly,  $h_{\text{prev}}$  is a smooth, normalized version of the color histogram of the segmentation in the previously segmented frame  $f_{t-1}$ . Note that  $a$  and  $f_t(x)$  should belong to  $\mathbb{R}^3$ . However, to limit the computation load, only the two most significant color components out of

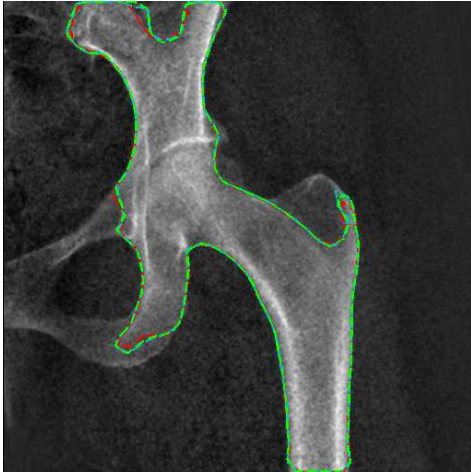


Figure 6: Direct/constrained segmentation comparison at a 16-pixel resolution: result with the direct approach (red plot), with the constrained approach using normal-based pre-defined velocities weighted according to Section 3.2.1 (cyan plot), and with weights computed as suggested in Section 4.2.2 (green plot).

three were considered. Function  $k$  is a positive function from  $\mathbb{R}^2$  to  $\mathbb{R}$  defined as  $k(x, y) = (x - y)^2$ . A maximal area constraint [34] was added to energy (73) since its sensitivity decreases in the inner neighborhood of the correct segmentation

$$E_A(\Gamma) = -\delta \int_{\Omega} dx \quad (75)$$

where  $\delta$  is a weighting parameter. The shape derivative of the sum of (73) and (75) is

$$\begin{aligned} dE(\Gamma, V) = & - \int_{\Gamma} \left( \frac{1}{|\Omega|} (g \star k'(h(\Gamma, \cdot), h_{\text{ref}}(\cdot))) (f_t(s)) \right. \\ & - \frac{1}{|\Omega|^2} \int_{\mathbb{R}^2} h(\Gamma, a) k'(h(\Gamma, a), h_{\text{ref}}(a)) da \\ & \left. - \delta \right) N(s) \cdot V(s) ds \end{aligned} \quad (76)$$

where  $\star$  is the convolution operator and  $k'$  is the partial derivative of  $k$  with respect to its first variable.

The minimization algorithm follows the same steps as Alg. 2. Resolution  $l$  was chosen equal to 10 pixels (to be compared to a size of frame of  $352 \times 288$  pixels). In step 5 of Alg. 2, the local motion at the vertices, required to determine the pre-defined velocities, was estimated by a sub-optimal block matching technique [42] with a  $1/4$ -pixel precision. Here, a block was defined as the intersection between a  $21 \times 21$ -pixel square centered on a vertex and the mask of the current active contour. This prevented the pixels considered to be outside the object from interfering with the motion estimation. This procedure is given for illustrative purposes only.

In a real-world tracking application, motion estimation should certainly be more sophisticated [1, 20, 33]. Figure 7 presents the results obtained with the constrained approach using normal-based pre-defined velocities and motion-based pre-defined velocities. The

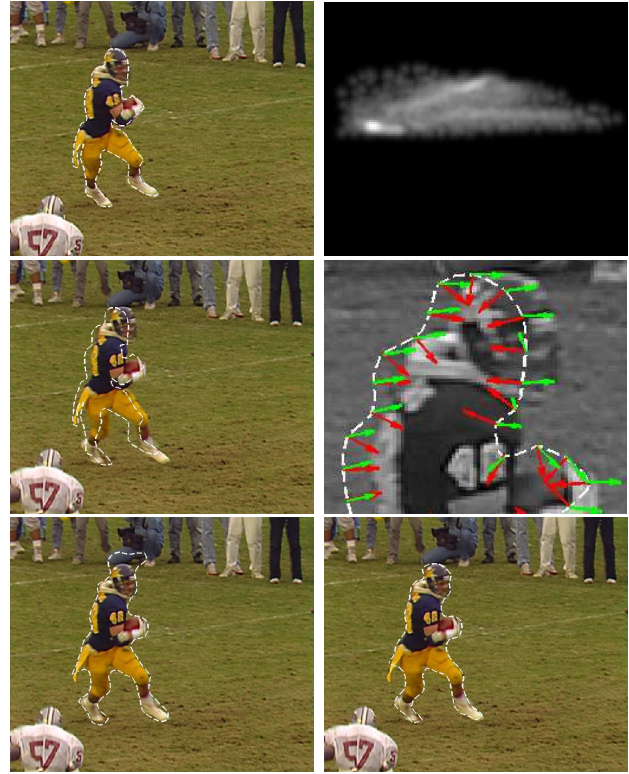


Figure 7: Tracking of a player on frame 75 of the standard test sequence “Football” with the constrained approach. *Top left*: manual segmentation of frame 74 (used as the initial contour for the segmentation of frame 75); *Top right*: histogram (for display purposes, upper and lower zero-valued regions were cut out) of the manually segmented region (used as histogram  $h_{\text{prev}}$ ); *Middle left*: initial contour on frame 75 (copied from frame 74); *Middle right*: a closeup of the normals (red arrows) and the local motions (green arrows) computed at the vertices of the polygon at the first iteration; *Bottom left*: segmentation of frame 75 using normal-based pre-defined velocities; *Bottom right*: segmentation of frame 75 using motion-based pre-defined velocities.

normal-based segmentation is globally similar to the motion-based segmentation except for a small region wrongly included above the player. These segmentations correspond to two local minima, one of which being more relevant. In a way, the motion-based evolution *took the shortest path* toward the object of interest and converged toward a more satisfying minimizer. As a matter of fact, the convergence was reached after 10 iterations while it took 14 iterations for the normal-

based evolution. An intuition of the behavior of the normal-based evolution is given by the normals shown in Fig. 7 compared to the local motions. When a normal has a direction close to the direction or opposite direction of the local motion<sup>7</sup>, then both evolution processes behave similarly. However, when these directions are close to be orthogonal, then the normal-based process could have a tendency to evolve toward a less relevant local minimum and/or to require more iterations to converge. Indeed, the local motion is likely to be a better direction to take in a tracking application.

### 5.3 An example of symmetry constraint

Pre-defined velocities can also be chosen in order to maintain symmetry. Suppose that the initial contour has an axis of symmetry. Although the previous examples used one pre-defined velocity per sample, it is not necessary to do so. A pre-defined velocity could be assigned to each pair of symmetric samples, the velocity being itself *symmetric* with respect to the axis. For example, if the contour is represented by a polygon, then the sum for a pair of symmetric samples of the normal-based pre-defined velocities proposed in Section 3.2.3 - definition (i) could specify one such pre-defined velocity. The weight of this velocity being unique, the displacements of the two related samples will be identical in norm. Since their directions is given by the local normal, which is symmetric with respect to the axis of symmetry, the symmetry of the contour will be maintained during evolution.

To illustrate this procedure, an object artificially made slightly asymmetric was segmented using energy (71) and Alg. 2 (see Fig.8). In this simple example, the axis of symmetry was supposed to be known. (For comparison purposes, the segmentation was also computed using one normal-based pre-defined velocity per sample, as in Section 5.1.) Note that this method does not allow to recover the symmetric object. The computed segmentation is rather a symmetric shape representing a trade-off between asymmetric regions kept inside and left outside the contour.

## 6 Conclusion

A variational approach to segmentation was presented and the shape derivative was highlighted as a convenient framework for deriving the evolution equation of an active contour from the energy to be minimized. It was noted that an error is induced by a discretization

<sup>7</sup>If the direction of the normal is opposite to the direction of the local motion, the shape gradient takes equal values in absolute value for both but with opposite signs, making no difference as far as evolution is concerned.

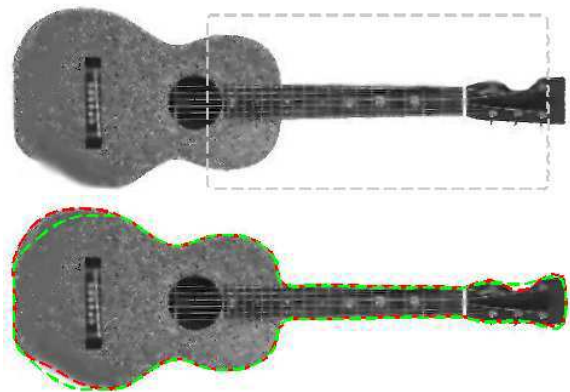


Figure 8: Segmentation of an asymmetric object with (*green contour*) and without (*red contour*) a symmetry constraint imposed by an appropriate choice of pre-defined velocities.

of the evolution equation when the contour velocity is taken equal to the gradient of the energy associated with the  $L^2$  inner product. Although there are no consequences in practice if the contour is discretized finely enough, the constrained approach proposed to solve this problem also gives more flexibility to the active contour process by allowing to introduce some *a priori* knowledge. This property can be useful, *e.g.*, for a tracking application where motion gives an *a priori* on the position of the tracked object in the next frame.

## 7 Acknowledgments

The authors would like to thank Jean-Paul Marmorat, École des Mines de Paris, C.M.A. Sophia Antipolis, France, for a fruitful discussion on the shape derivative in the context of a parametric active contour.

## A Shape gradient of a boundary energy

Let us consider the following boundary energy

$$E(\Gamma) = \int_{\Gamma} \varphi(s) \, ds \quad (77)$$

where  $\Gamma$  is the oriented boundary  $\partial\Omega$  of an open set  $\Omega$  of  $\mathbb{R}^2$  and  $s$  is the arc-length parameterization of  $\Gamma$ . The shape derivative of (77) is equal to [36, 13]

$$dE(\Gamma, V) = \int_{\Gamma} \left( \frac{\partial \varphi(s)}{\partial N} - \varphi(s) \kappa(s) \right) N(s) \cdot V(s) \, ds \quad (78)$$

where  $N$  is the inward unit normal of  $\Gamma$  and  $\kappa$  is the curvature of  $\Gamma$ .

Note that this result can also be obtained by a calculus of variations [5].

## B Shape gradient of a domain energy

Let  $\Omega$  be an open set of  $\mathbb{R}^2$  and let  $T$  be a transformation of  $\Omega$  such that

$$\begin{cases} \Omega = T(\tau = 0, \Omega) \\ \Omega_T(\tau) = T(\tau, \Omega) \\ x_T(\tau) = T(\tau, x), \quad x \in \Omega \end{cases} . \quad (79)$$

The deformation at  $x$  is defined as

$$V_T(x) := \lim_{\tau \rightarrow 0} \frac{x_T(\tau) - x}{\tau} \quad (80)$$

$$= \frac{\partial T}{\partial \tau}(0, x) . \quad (81)$$

For clarity,  $\Omega_T(\tau)$ ,  $x_T(\tau)$ , and  $V_T(x)$  are referred to as  $\Omega(\tau)$ ,  $x(\tau)$ , and  $V(x)$ , respectively. Function  $V$  is called the velocity of  $\Omega$ .

Let us consider the following domain energy

$$E(\Gamma) = \int_{\Omega} \phi(\Gamma, x) \, dx \quad (82)$$

where  $\Gamma$  is the oriented boundary  $\partial\Omega$  of  $\Omega$ . The same energy computed for the transformed domain  $\Omega(\tau)$  is equal to

$$E(\Gamma, T, \tau) = \int_{\Omega(\tau)} \phi(\Gamma(\tau), x) \, dx . \quad (83)$$

Then, the shape derivative of (82) is defined as

$$dE(\Gamma, T) := \lim_{\tau \rightarrow 0} \frac{E(\Gamma, T, \tau) - E(\Gamma, T, 0)}{\tau} . \quad (84)$$

It is equal to [36, 13]

$$\begin{aligned} dE(\Gamma, T) &= \int_{\Omega} \frac{\partial \phi(\Gamma(\tau), x)}{\partial \tau} \Big|_{\tau=0} \, dx \\ &\quad - \int_{\Gamma} \phi(\Gamma, s) \, N(s) \cdot V(s) \, ds \end{aligned} \quad (85)$$

where  $s$  is the arc-length parameterization of  $\Gamma$  and  $N$  is the inward unit normal of  $\Gamma$ . Since  $V$  appears explicitly in the shape derivative expression,  $dE(\Gamma, T)$  can be replaced with  $dE(\Gamma, V)$ .

Note that this result can also be obtained by a calculus of variations [2].

## C Rewriting the shape derivative as a boundary integral

Under weak assumptions, the shape derivative of a domain energy (see Appendix B) has an equivalent expression in the form of a boundary integral [36, 37]. Such an expression is convenient in the active contour

framework (see Section 3.1.1). Indeed, the evolution equation of an active contour can be easily deduced from a boundary integral-only expression.

Two (sets of) conditions are proposed to allow to practically rewrite shape derivative (85) into an expression without any domain integral.

### C.1 Recursive applications of the shape derivative

Under some conditions on the dependency of  $\phi$  on  $\Gamma$ , applying recursively (85) to its first integral leads to an expression containing no domain integral [23]

$$dE(\Gamma, V) = - \int_{\Gamma} \Psi(\Gamma, s) \, N(s) \cdot V(s) \, ds . \quad (86)$$

Let us assume that

$$\begin{cases} \phi_i(\Gamma, x) = \phi_i(g_i(\Gamma), x), \quad i \in [1, n-1] \\ g_i(\Gamma) = \int_{\Omega} \phi_{i+1}(\Gamma, x) \, dx, \quad i \in [1, n-1] \\ \phi_n(\Gamma, x) = \phi_n(x) \end{cases} \quad (87)$$

where  $\phi_1 = \phi$  (note that  $g_i$  has the same form as (82)). The first integral of (85) reads

$$\begin{aligned} \int_{\Omega} \frac{\partial \phi_1(g_1(\Gamma(\tau)), x)}{\partial \tau} \Big|_{\tau=0} \, dx &= \int_{\Omega} \frac{\partial \phi_1(k, x)}{\partial k} \Big|_{g_1(\Gamma)} \\ &\quad \frac{\partial g_1(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \, dx \end{aligned} \quad (88)$$

$$\begin{aligned} &= \frac{\partial g_1(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \\ &\quad \int_{\Omega} \frac{\partial \phi_1(k, x)}{\partial k} \Big|_{g_1(\Gamma)} \, dx \end{aligned} \quad (89)$$

$$\begin{aligned} &:= dg_1(\Gamma, V) \\ &\quad \int_{\Omega} \frac{\partial \phi_1(k, x)}{\partial k} \Big|_{g_1(\Gamma)} \, dx \end{aligned} \quad (90)$$

$$= dg_1(\Gamma, V) \, A_1(\Gamma) \quad (91)$$

where  $dg_1(\Gamma, V)$  is equal to

$$\begin{aligned} dg_1(\Gamma, V) &= \int_{\Omega} \frac{\partial \phi_2(\Gamma(\tau), x)}{\partial \tau} \Big|_{\tau=0} \, dx \\ &\quad - \int_{\Gamma} \phi_2(\Gamma, s) \, N(s) \cdot V(s) \, ds . \end{aligned} \quad (92)$$

Then, the development leading to (91) can be repeated with the successive domain integrals present in  $dg_i(\Gamma, V)$ ,  $i$  increasing, until

$$dg_{n-1}(\Gamma, V) = - \int_{\Gamma} \phi_n(s) \, N(s) \cdot V(s) \, ds \quad (93)$$

which does not contain any domain integral since  $\phi_n$  is independent of  $\Gamma$ . Gathering the successive boundary integrals together, the shape derivative is equal to

$$\begin{aligned} dE(\Gamma, V) &= - \int_{\Gamma} \Psi(\Gamma, s) N(s) \cdot V(s) ds \quad (94) \\ &= - \int_{\Gamma} \left( \sum_{i=1}^n \phi_i(g_i(\Gamma), s) \prod_{j=1}^{i-1} A_j(\Gamma) \right) \\ &\quad N(s) \cdot V(s) ds \quad (95) \end{aligned}$$

where  $A_j(\Gamma)$  is equal to

$$A_j(\Gamma) = \int_{\Omega} \frac{\partial \phi_j(x, k)}{\partial k} \Big|_{g_j(\Gamma)} dx \quad (96)$$

with the following convention

$$\phi_n(g_n(\Gamma), s) = \phi_n(s) . \quad (97)$$

For instance,  $\phi$  may involve the variance  $g_1$  of  $f$  in  $\Omega$ . The variance involves the average value  $g_2$  of  $f$  in  $\Omega$ . Finally, the average value involves  $f$  but no term depending on  $\Gamma$ .

## C.2 Domain integral equal to zero

Under some conditions,  $\Psi$  in (86) is simply equal to  $\phi$  [34]. In other words, the first integral of (85) is equal to zero. For instance, if  $\phi$  is given by (2), *i.e.*,  $(f(x) - \mu(\Gamma))^2$ , the integrand of the first integral of (85) is equal to

$$\begin{aligned} \frac{\partial \phi(\Gamma(\tau), x)}{\partial \tau} \Big|_{\tau=0} &= \frac{\partial ((f(x) - \mu(\Gamma(\tau)))^2)}{\partial \tau} \Big|_{\tau=0} \quad (98) \\ &= -2 (f(x) - \mu(\Gamma)) \frac{\partial \mu(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \quad (99) \end{aligned}$$

where  $\mu(\Gamma)$  can be written as follows

$$\mu(\Gamma) = \int_{\Omega} f(x) dx / \int_{\Omega} dx . \quad (100)$$

Then, the first integral of (85) is equal to

$$\begin{aligned} \int_{\Omega} \frac{\partial \phi(\Gamma(\tau), x)}{\partial \tau} \Big|_{\tau=0} dx &= -2 \frac{\partial \mu(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \\ &\quad \int_{\Omega} (f(x) - \mu(\Gamma)) dx \quad (101) \\ &= -2 \frac{\partial \mu(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \\ &\quad \left( \int_{\Omega} f(x) dx \right. \\ &\quad \left. - \mu(\Gamma) \int_{\Omega} dx \right) \quad (102) \\ &= 0 . \quad (103) \end{aligned}$$

In general, a sufficient condition for the first integral of (85) to be equal to zero is

$$\begin{cases} \phi(\Gamma, x) = \phi(g(\Gamma), x) \\ g(\Gamma) = \arg \min_k \int_{\Omega} \phi(k, x) dx \end{cases} . \quad (104)$$

In other words,  $g(\Gamma)$  is the minimizer of (82) seen as a function of  $g$  with  $\Gamma$  fixed. For convenience, the following notation will be used

$$g(\Gamma) = \arg \min_k E_{\Gamma}(k) . \quad (105)$$

If  $g$  is assumed to be differentiable, the development of (91) can be continued using the new condition (104)

$$\begin{aligned} \int_{\Omega} \frac{\partial \phi(g(\Gamma(\tau)), x)}{\partial \tau} \Big|_{\tau=0} dx &= dg(\Gamma, V) \\ &\quad \int_{\Omega} \frac{\partial \phi(k, x)}{\partial k} \Big|_{g(\Gamma)} dx \quad (106) \\ &= dg(\Gamma, V) \\ &\quad \frac{\partial}{\partial k} \int_{\Omega} \phi(k, x) dx \Big|_{g(\Gamma)} \quad (107) \\ &= dg(\Gamma, V) \frac{\partial E_{\Gamma}(k)}{\partial k} \Big|_{g(\Gamma)} \quad (108) \end{aligned}$$

If  $g(\Gamma)$  is a constraint-free minimizer of  $E_{\Gamma}$ , then it can be immediately concluded that (108) is equal to zero. Otherwise, let us assume that  $g(\Gamma)$  is the minimizer of  $E_{\Gamma}$  under the constraint

$$\xi(g(\Gamma)) = 0 . \quad (109)$$

Then, there exists a Lagrange multiplier  $\lambda$  such that

$$\frac{\partial E_{\Gamma}(k)}{\partial k} \Big|_{g(\Gamma)} = \lambda \frac{\partial \xi(k)}{\partial k} \Big|_{g(\Gamma)} . \quad (110)$$

Therefore,

$$\begin{aligned} \int_{\Omega} \frac{\partial \phi(g(\Gamma(\tau)), x)}{\partial \tau} \Big|_{\tau=0} dx &= \lambda \frac{\partial g(\Gamma(\tau))}{\partial \tau} \Big|_{\tau=0} \\ &\quad \frac{\partial \xi(k)}{\partial k} \Big|_{g(\Gamma(\tau=0))} \quad (111) \\ &= \lambda \frac{\partial \xi(g(\Gamma(\tau)))}{\partial \tau} \Big|_{\tau=0} \quad (112) \end{aligned}$$

By definition, for any  $\tau$ ,  $\xi(g(\Gamma(\tau)))$  has the same value (equal to zero). It can be concluded that (112) is equal to zero.

As suggested by a reviewer, noting that  $E(\Gamma)$  can be written as  $F(\Gamma, g(\Gamma))$  leads to an immediate proof of the result of C.2.

## D Validity of the direct approach

As suggested by a reviewer, despite the discretization error, the direct approach can still be valid (*i.e.*, evolution (16) generates a minimizing sequence) if the contour discretization is fine enough. The following development is largely inspired from the reviewer's proof.

Let  $G$  be the gradient associated with the  $L^2$  inner product

$$G(\Gamma, s) = -\Psi(\Gamma, s) N(s). \quad (113)$$

For convenience,  $G(s)$  and  $\Psi(s)$  will be used instead of  $G(\Gamma, s)$  and  $\Psi(\Gamma, s)$ , respectively. Let us assume that  $\Gamma$  is represented by a polygon with  $n$  edges of equal length  $l$  (see Section 1.4 for notations). Note that  $l$  is equal to  $s_{i+1} - s_i$ . Let  $L$  be the length of  $\Gamma$ . Let  $N$  be a  $C^2$  pseudo-normal of  $\Gamma$ . Function  $\Psi$  is also assumed to be  $C^2$  so that  $G''$  exists and is continuous. With the direct approach, the induced velocity  $\tilde{V}$  at  $s \in [s_i, s_{i+1}]$  is equal to

$$\tilde{V}(s) = -(1 - \bar{s}) G(s_i) - \bar{s} G(s_{i+1}) \quad (114)$$

$$= \Psi(s_i) (1 - \bar{s}) N(s_i) + \Psi(s_{i+1}) \bar{s} N(s_{i+1}) \quad (115)$$

where  $\bar{s}$  is equal to  $(s - s_i)/l$ . Note that  $(1 - \bar{s}) N(s_i)$  and  $\bar{s} N(s_{i+1})$  are equal to  $V_i(s)$  and  $V_{i+1}(s)$ , respectively, the normal-based pre-defined velocities proposed in Section 3.2.3 - definition (i). Also note that  $V_j(s)$  is equal to zero if  $j$  is not equal to  $i$  or  $i + 1$ . Therefore, the induced velocity can be rewritten as follows

$$\tilde{V}(s) = \sum_{i=1}^n \Psi(s_i) V_i(s) \quad (116)$$

where  $s$  can now belong to  $[0, L]$ . The shape derivative (9) at  $\tilde{V}$  is equal to

$$dE(\Gamma, \tilde{V}) = \int_{\Gamma} G(s) \cdot \sum_{i=1}^n \Psi(s_i) V_i(s) ds \quad (117)$$

$$= \sum_{i=1}^n \Psi(s_i) \int_{\Gamma} G(s) \cdot V_i(s) ds. \quad (118)$$

Velocity  $V_i$  being different from zero on  $\gamma_{i-1}$  and  $\gamma_i$  only, the shape derivative is equal to

$$dE(\Gamma, \tilde{V}) = \sum_{i=1}^n \Psi(s_i) \left( \int_{\gamma_{i-1}} G(s) \cdot V_i(s) ds + \int_{\gamma_i} G(s) \cdot V_i(s) ds \right). \quad (119)$$

According to Taylor's theorem with the Lagrange form of the remainder term, there exists  $t_i$  in  $]s_{i-1}, s_{i+1}[$  such that

$$G(s) = G(s_i) + G'(s_i) (s - s_i) + \frac{G''(t_i)}{2} (s - s_i)^2. \quad (120)$$

The first integral of (119) is equal to

$$I_1 = G(s_i) \cdot \int_{\gamma_{i-1}} V_i(s) ds + G'(s_i) \cdot \int_{\gamma_{i-1}} (s - s_i) V_i(s) ds + \frac{G''(t_i)}{2} \cdot \int_{\gamma_{i-1}} (s - s_i)^2 V_i(s) ds. \quad (121)$$

With the change of variable  $t = 2s_i - s$ , noting that  $2s_i - s_{i-1}$  is equal to  $s_{i+1}$ , and due to the symmetry of  $V_i$  with respect to  $s_i$ , one can verify that

$$\int_{\gamma_{i-1}} (s - s_i) V_i(s) ds = - \int_{\gamma_i} (s - s_i) V_i(s) ds. \quad (122)$$

Therefore, (119) is equal to

$$dE(\Gamma, \tilde{V}) = \sum_{i=1}^n \Psi(s_i) \left( G(s_i) \cdot \int_{\Gamma} V_i(s) ds + \frac{G''(t_i)}{2} \cdot \int_{\Gamma} (s - s_i)^2 V_i(s) ds \right) \quad (123)$$

Replacing  $V_i$  with its expression, one can verify that  $\int_{\Gamma} V_i(s) ds$  is equal to  $l N(s_i)$  and that

$$\int_{\Gamma} (s - s_i)^2 V_i(s) ds = \frac{l^2}{6} \int_{\Gamma} V_i(s) ds. \quad (124)$$

Finally, (119) is equal to

$$dE(\Gamma, \tilde{V}) = \sum_{i=1}^n \Psi(s_i) \left( G(s_i) + \frac{l^2}{12} G''(t_i) \right) \cdot \int_{\Gamma} V_i(s) ds \quad (125)$$

$$= \sum_{i=1}^n \Psi(s_i) (-\Psi(s_i) N(s_i) + \frac{l^2}{12} G''(t_i)) \cdot l N(s_i) \quad (126)$$

$$= l \sum_{i=1}^n \Psi(s_i) \frac{l^2}{12} G''(t_i) \cdot N(s_i) - \Psi(s_i)^2. \quad (127)$$

If  $l$  is such that

$$l^2 \left| \sum_{i=1}^n \frac{\Psi(s_i)}{12} G''(t_i) \cdot N(s_i) \right| \leq \sum_{i=1}^n \Psi(s_i)^2, \quad (128)$$

then the negativity of (127) is guaranteed. The following stronger condition can be deduced from (128)

$$l^2 \frac{M}{12} \sum_{i=1}^n |\Psi(s_i)| \leq \sum_{i=1}^n \Psi(s_i)^2 \quad (129)$$

$$\iff l^2 \leq \frac{12}{M} \frac{\sum_{i=1}^n \Psi(s_i)^2}{\sum_{i=1}^n |\Psi(s_i)|} \quad (130)$$

$$\iff l^2 \leq \frac{12}{M} \frac{l \sum_{i=1}^n \Psi(s_i)^2}{\sum_{i=1}^n |\Psi(s_i)|} \quad (131)$$



where  $M$  is equal to

$$M = \sup_s |G''(s)| \quad (132)$$

and  $M$  and  $\sum_{i=1}^n |\Psi(s_i)|$  are assumed to be different from zero (otherwise, the negativity of (127) is trivially verified). Functions  $|\Psi|$  and  $\Psi^2$  being Riemann-integrable, the limit when  $l$  tends toward zero of the right-hand side of (131) exists and is equal to<sup>8</sup>

$$\frac{12}{M} \frac{\int_{\Gamma} \Psi(s)^2 ds}{\int_{\Gamma} |\Psi(s)| ds}, \quad (133)$$

assuming that  $\Psi$  is not identically equal to zero on  $\Gamma$  (otherwise, the negativity of (127) is again trivially verified). Limit (133) being strictly positive, there exists a value  $l_{\max}$  such that if  $l$  is smaller than  $l_{\max}$ , then condition (131) is verified, implying the negativity of (127).

Let  $I_u$  and  $S_u$  be the integral and the sum at the numerators of (133) and (131), respectively. The error between  $I_u$  and  $S_u$  is such that

$$|I_u - S_u| \leq L P l \quad (134)$$

$$\implies I_u - L P l \leq S_u \quad (135)$$

where  $P$  is equal to  $2 \sup_s |\Psi(s) \Psi'(s)|$ . Let  $I_d$  and  $S_d$  be the integral and the sum at the denominators of (133) and (131), respectively. Using the Chasles's equality and the mean value theorem, one can check that

$$S_d - I_d = l \sum_{i=1}^n |\Psi(s_i)| - |\Psi(t_i)| \quad (136)$$

$$\leq n l \underbrace{(\sup_s |\Psi(s)| - \inf_s |\Psi(s)|)}_Q \quad (137)$$

where  $t_i$  belongs to  $]s_i, s_{i+1}[$ . Then, a condition stronger than (131) is

$$l^2 \leq \frac{12}{M} \frac{I_u - L P l}{I_d + L Q} \quad (138)$$

$$\iff \frac{M}{12} (I_d + L Q) l^2 + L P l - I_u \leq 0 \quad (139)$$

$$\iff l \leq l_{\max}^{\text{root}} \quad (140)$$

where  $l_{\max}^{\text{root}}$  is the largest root of the polynomial in (139). One can check that it exists and is positive.

Note that  $l_{\max}^{\text{root}}$  depends on  $\Gamma$  and  $\Psi$ , which implies that it should be computed at each iteration of the active contour evolution.

The reviewer's proof led to the following, much simpler upper bound  $1/\sqrt{M \sqrt{L}}$ .

<sup>8</sup>Remember that  $s_n$  is equal to  $L-l$  (see Section 1.4). Therefore, it is correct to sum for all  $i$  up to  $n$  (and not up to  $n-1$ ) to approximate the integral over  $\Gamma$ .

## E Some properties of closed, uniform cubic B-splines

Here, we present the properties of closed, uniform cubic B-splines that are used in Sections 3.2.4 and 4.3.2.

A closed, uniform cubic B-spline with  $n$  segments [3, 21, 32, 40] is uniquely defined by the coordinates  $(a_i, b_i)$  of  $n$  control points  $q_i$  and the following blending function  $B$  (see Fig. 9)

$$B(t) = \begin{cases} \frac{2}{3} - t^2 + \frac{|t|^3}{2} & \text{if } 0 \leq |t| < 1 \\ \frac{(2-|t|)^3}{6} & \text{if } 1 \leq |t| < 2 \\ 0 & \text{if } |t| \geq 2 \end{cases} \quad (141)$$

The equation of spline segment  $\gamma_i$  is

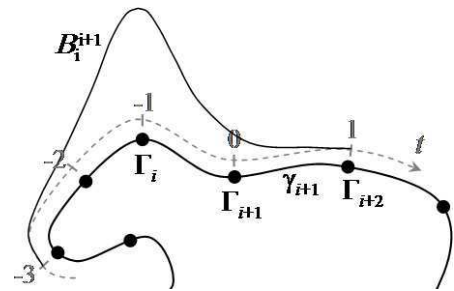
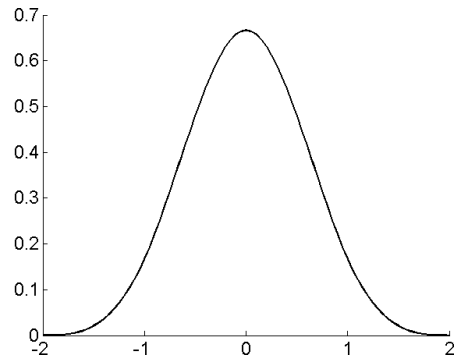


Figure 9: *Top*: the uniform cubic B-spline blending function  $B$ . It is equal to zero outside the interval  $[-2, 2]$ ; *Bottom*: function  $B_i^{i+1}$ , i.e., blending function  $B$  shifted to be “centered” on  $\Gamma_i$  and “seen” from segment  $\gamma_{i+1}$ .

$$\gamma_i(t) = \sum_{j=i-1}^{i+2} q_j B_j^i(t), \quad t \in [0, 1] \quad (142)$$

where  $B_j^i$  represents blending function  $B$  shifted to be “centered” on  $\Gamma_j$  and “seen” from segment  $\gamma_i$  (see Fig. 9). Mathematically,

$$B_j^i(t) = B(t - (j - i)). \quad (143)$$

Note that the following notation abuses are made

$$\begin{cases} q_0 = q_n \\ q_{n+1} = q_1 \\ q_{n+2} = q_2 \end{cases} \quad (144)$$

The extremities  $\gamma_i(0)$  and  $\gamma_i(1)$  of a segment are equal to sampling points  $\Gamma_i$  and  $\Gamma_{i+1}$ , respectively.

**Property 1** Although  $B_j^i(t)$  was defined for  $t \in [0, 1]$ , the domain of definition can be extended to  $[-n/2, n/2]$ , covering the entire curve. Note that if  $t$  is an integer in this extended interval, then it corresponds to  $\Gamma_{i+t}$ .

**Property 2** Sampling points and control points are linked together by the following equation

$$\Gamma_i = \frac{1}{6}(q_{i-1} + 4q_i + q_{i+1}) \quad (145)$$

**Property 3** Replacing  $B$  with its definition, the segment equation reads

$$\begin{aligned} \gamma_i(t) = & \frac{t^3}{6} (q_{i+2} - 3q_{i+1} + 3q_i - q_{i-1}) \\ & + \frac{t^2}{2} (q_{i+1} - 2q_i + q_{i-1}) \\ & + \frac{t}{2} (q_{i+1} - q_{i-1}) + \frac{1}{6} (q_{i+1} + 4q_i + q_{i-1}), \\ & t \in [0, 1] \end{aligned} \quad (146)$$

or, using matrices,

$$\gamma_i(t) = P(t) Q_i \quad (147)$$

where

$$\begin{cases} P(t) = \begin{pmatrix} \frac{t^3}{6} & -\frac{t^3}{2} + \frac{t^2}{2} + \frac{t}{2} + \frac{1}{6} \\ \frac{t^3}{2} - t^2 + \frac{2}{3} & -\frac{t^3}{6} + \frac{t^2}{2} - \frac{t}{2} + \frac{1}{6} \end{pmatrix} \\ Q_i = (q_{i+2} \quad q_{i+1} \quad q_i \quad q_{i-1})^T \end{cases} \quad (148)$$

and  $A^T$  is the transpose of matrix  $A$ . Let  $(e_1, e_2)$  be the canonical basis of  $\mathbb{R}^2$ . Note that

$$P(t) = \begin{pmatrix} \frac{\partial \gamma_i}{\partial a_{i+2}}(t) \cdot e_1 & \frac{\partial \gamma_i}{\partial a_{i+1}}(t) \cdot e_1 \\ \frac{\partial \gamma_i}{\partial a_i}(t) \cdot e_1 & \frac{\partial \gamma_i}{\partial a_{i-1}}(t) \cdot e_1 \end{pmatrix} \quad (149)$$

$$= \begin{pmatrix} \frac{\partial \gamma_i}{\partial b_{i+2}}(t) \cdot e_2 & \frac{\partial \gamma_i}{\partial b_{i+1}}(t) \cdot e_2 \\ \frac{\partial \gamma_i}{\partial b_i}(t) \cdot e_2 & \frac{\partial \gamma_i}{\partial b_{i-1}}(t) \cdot e_2 \end{pmatrix} \quad (150)$$

**Property 4** Let  $j$  be an integer in the interval  $[i - 1, i + 2]$ . Then, the derivative of (146) with respect to  $a_j$  is equal to

$$\frac{\partial \gamma_i}{\partial a_j}(t) = \alpha(t) e_1 \quad (151)$$

where  $\alpha$  is a polynom. If  $j$  is outside the interval  $[i - 1, i + 2]$ , then

$$\frac{\partial \gamma_i}{\partial a_j}(t) = 0 \quad (152)$$

since, in this case,  $\gamma_i$  is independent of  $a_j$ . In both cases,

$$\frac{\partial \gamma_i}{\partial a_j}(t) \cdot e_2 = 0 \quad (153)$$

Similarly,

$$\frac{\partial \gamma_i}{\partial b_j}(t) \cdot e_1 = 0 \quad (154)$$

**Property 5** Considering (151) and (152), it can be deduced that

$$\sum_{j=i-1}^{i+2} \alpha_j \frac{\partial \gamma_i}{\partial a_j}(t) = \sum_{j=1}^n \alpha_j \frac{\partial \gamma_i}{\partial a_j}(t) \quad (155)$$

for any set of  $\alpha_j$ 's, and similarly when taking the derivative with respect to  $b_j$ .

## References

- [1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188, 2002.
- [2] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson. Image segmentation using active contours: Calculus of variations or shape gradients? *SIAM Journal on Applied Mathematics*, 63:2128–2154, 2003.
- [3] P. Brigger, J. Hoeg, and M. Unser. B-spline snakes: A flexible tool for parametric contour detection. *IEEE Transactions on Image Processing*, 9:1484–1496, 2000.
- [4] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.
- [5] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22:61–79, 1997.
- [6] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10:266–277, 2001.
- [7] G. Charpiat, R. Keriven, J.-P. Pons, and O. Faugeras. Designing spatially coherent minimizing flows for variational problems based on active contours. In *International Conference on Computer Vision*, 2:1403–1408, 2005.

- [8] C. Chesnaud, P. Réfrégier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:1145–1156, 1999.
- [9] D. Cremers and C. Schnörr. Motion competition: Variational integration of motion segmentation and shape regularization. In *DAGM-Symposium*, LNCS-2449:472–480, 2002.
- [10] D. Cremers and S. Soatto. Variational space-time motion segmentation. In *International Conference on Computer Vision*, 2:886–893, 2003.
- [11] 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*, 20:643–659, 2001.
- [12] E. Debreuve, M. Barlaud, J.-P. Marmorat, and G. Aubert. Active contour segmentation with a parametric shape prior: link with the shape gradient. In *International Conference on Image Processing*, 1653–1656, 2006.
- [13] M. C. Delfour and J.-P. Zolésio. *Shapes and geometries: Analysis, differential calculus and optimization*. Advances in Design and Control. Society for Industrial and Applied Mathematics, Philadelphia, 2001.
- [14] H. Delingette and J. Montagnat. Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding*, 83:140–171, 2001.
- [15] G. Doğan, P. Morin, R. H. Nochetto, and M. Verani. Finite element methods for shape optimization and applications. *Preprint*, 2005.
- [16] M. Gastaud, M. Barlaud, and G. Aubert. Combining shape prior and statistical features for active contour segmentation. *IEEE Transactions on Circuits Systems and Video Technology*, 14:726–734, 2004.
- [17] O. Germain and P. Réfrégier. Optimal snake-based segmentation of a random luminance target on a spatially disjoint background. *Optical Letters*, 21:1845–1847, 1996.
- [18] E. Haug and K. K. Choi. *Methods of engineering mathematics*. Prentice Hall, Englewood Cliffs, 1993.
- [19] M. Hintermüller and W. Ring. A second order shape optimization approach for image segmentation. *SIAM Journal on Applied Mathematics*, 64:442–467, 2003.
- [20] M. Isard and A. Blake. CONDENSATION - Conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29:5–28, 1998.
- [21] M. Jacob, T. Blu, and M. Unser. A unifying approach and interface for spline-based snakes. In *International Symposium on Medical Imaging: Image Processing, Proc. SPIE*, 4322:340–347, 2001.
- [22] M. Jacob, T. Blu, and M. Unser. Efficient energies and algorithms for parametric snakes. *IEEE Transactions on Image Processing*, 13:1231–1244, 2004.
- [23] S. Jehan-Besson, M. Barlaud, and G. Aubert. DREAM<sup>2</sup>S: Deformable regions driven by an eulerian accurate minimization method for image and video segmentation. *International Journal of Computer Vision*, 53:45–70, 2003.
- [24] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–332, 1988.
- [25] H. Li, A. Elmoataz, J. Fadili, and S. Ruan. Dual front evolution model and its application in medical imaging. In *Medical Image Computing and Computer-Assisted Intervention*, LNCS-3216:103–110, 2004.
- [26] H. Li and A. Yezzi. Local or global minima: flexible dual front active contours. In *Computer Vision for Biomedical Image Applications*, LNCS-3765:356–366, 2005.
- [27] S. Lobregt and M. Viergever. A discrete dynamic contour model. *IEEE Transactions on Medical Imaging*, 14:12–23, 1995.
- [28] J.-P. Marmorat, 2005. Private communication on shape gradient in the context of a parametric active contour.
- [29] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–685, 1989.
- [30] N. C. Overgaard and J. E. Solem. An analysis of variational alignment of curves in images. In *International Conference on Scale Space and PDE methods in Computer Vision*, LNCS-3459:480–491, 2005.
- [31] N. Paragios and R. Deriche. Geodesic active regions for motion estimation and tracking. In *International Conference on Computer Vision*, 1:688–694, 1999.

- [32] F. Precioso, M. Barlaud, T. Blu, and M. Unser. Robust real-time segmentation of images and videos using a smooth-spline snake-based algorithm. *IEEE Transactions on Image Processing*, 14:910–924, 2005.
- [33] D. Robinson and P. Milanfar. Fast local and global projection-based methods for affine motion estimation. *Journal of Mathematical Imaging Vision*, 18:35–54, 2003.
- [34] T. Roy, E. Debreuve, M. Barlaud, and G. Aubert. Segmentation of a vector field: Dominant parameter and shape optimization. *Journal of Mathematical Imaging Vision*, 24:259–276, 2006.
- [35] C. Schnörr. Computation of discontinuous optical flow by domain decomposition and shape optimization. *International Journal of Computer Vision*, 8:153–165, 1992.
- [36] J. Sokolowski and J.-P. Zolésio. *Introduction to shape optimization: Shape sensitivity analysis*. Springer, 1992.
- [37] J. E. Solem and N. C. Overgaard. A geometric formulation of gradient descent for variational problems with moving surfaces. In *International Conference on Scale Space and PDE methods in Computer Vision*, LNCS-3459:419–430, 2005.
- [38] B. Taton and J.-O. Lachaud. Deformable model with non-Euclidean metrics. In *European Conference on Computer Vision*, LNCS-2352:438–453, 2002.
- [39] G. Unal, A. Yezzi, and H. Krim. Information-theoretic active polygons for unsupervised texture segmentation. *International Journal of Computer Vision*, 62:199–220, 2005.
- [40] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part I - Theory. *IEEE Transactions on Signal Processing*, 41:821–833, 1993.
- [41] A. J. Yezzi, A. Tsai, and A. Willsky. A statistical approach to snakes for bimodal and trimodal imagery. In *International Conference on Computer Vision*, 2:898–903, 1999.
- [42] S. Zhu and K.-K. Ma. A new diamond search algorithm for fast block matching motion estimation. *IEEE Transactions on Image Processing*, 9:287–290, 2000.