International Journal of Computer Vision, 13:2, 229–251 (1994)

Region-Based Strategies for Active Contour Models

RÉMI RONFARD*

remi@ina.fr

Centre d'Imagerie et Télédetection, Ecole des Mines de Paris, Sophia Antipolis,
06 700, Valbonne, France, and Télécom Paris, Laboratoire Image, 46, rue Barrault,
75 013 Paris, France

Received January, 1992. Revised February and July, 1993.

Abstract

The variational method has been introduced by Kass et al. (1987) in the field of object contour modeling, as an alternative to the more traditional edge detection–edge thinning–edge sorting sequence. Since the method is based on a pre-processing of the image to yield an edge map, it shares the limitations of the edge detectors it uses. In this paper, we propose a modified variational scheme for contour modeling, which uses no edge detection step, but local computations instead—only around contour neighborhoods—as well as an "anticipating" strategy that enhances the modeling activity of deformable contour curves. Many of the concepts used were originally introduced to study the local structure of discontinuity, in a theoretical and formal statement by Leclerc & Zucker (1987), but never in a practical situation such as this one. The first part of the paper introduces a region-based energy criterion for active contours, and gives an examination of its implications, as compared to the gradient edge map energy of snakes. Then, a simplified optimization scheme is presented, accounting for internal and external energy in separate steps. This leads to a complete treatment, which is described in the last sections of the paper (4 and 5). The optimization technique used here is mostly heuristic, and is thus presented without a formal proof, but is believed to fill a gap between snakes and other useful image representations, such as split-and-merge regions or mixed line-labels image fields.

1 Introduction

1.1 The Contour Modeling Problem in Image Analysis

This paper addresses the problem of automatically creating geometric models for the external boundaries of objects in a 2D image grid. We call the geometric representation a "contour" of the object, and reserve the term "boundary" for its pixel location in the image. A contour representation of objects can be useful for image understanding in 2D or 3D.

The contour modeling problem has traditionally received two opposed approaches: region-based approaches derive a contour representation from a segmentation of the image into well-defined regions, while edge-based methods use a continuous approximation of the original image function, so that boundary points can be characterized by a differential property (image gradient or curvature) and a contour representation be fitted to the boundary points. In their simpler versions, both methods use a point-wise criterium to decide if a given pixel belongs inside an object, outside all objects or at an object boundary. In the region-based approach, a pixel belongs to the boundary if it is in the object region and has neighbors in the background. In the edge-based approach, a pixel belongs to the boundary if it passes a numerical test (e.g. local maximum of the image gradient). In this early boundary detection step, such methods do not take into account the fact that those boundary points really constitute a closed geometric contour, with usually strong continuity and smoothness properties. The next step consists of an approximation method, which strives to find an optimal contour going through all boundary points, but has no interaction with the first step. The deficiency of those methods lies in the absence of top-down mechanisms, such

*Current address: Computervision, 14 Crosby Drive, Bedford, MA 02144, Phone: (617) 275-1800 that boundary detection could be guided by contour constraints.

1.2 Active Contour Models: Approaches and Previous Work

As an alternative to the traditional approach presented in the previous section, the methods of variational calculus (Prenter 1989) have been used by Kass, Witkins and Terzopoulos (1987) in the field of contour modeling. The resulting contour models were named "active contour models" or "snakes." The snakes method provides a way to constraint the points that are tested as boundaries, so that they constitute a parametric curve (or, more simply, an N-sided polygon). Starting from a user-defined curve, an energy minimization algorithm is used to deform the contour model until it fits objects boundaries. The method is gradient-based, and the criterium that characterizes boundary points is summed up over the whole contour to provide the goodness-of-fit measure (or external energy). A smoothness criterium (or internal energy) is also added to guarantee good convergence properties and robustness. Those internal energies provide a very nice framework for top-down processes as mentioned above, and can be theoretically founded on regularization theory.

Active contour models provide a very appealing and successful alternative to the more contrived sequence of boundary points detection and contour curve approximation. They have received much attention in the last few years, and have been improved significantly, notably by Fua & Leclerc (1990), Menet et al. (1990), Amini et al. (1990) and Cohen (1991). They have been applied to image understanding problems in 2D (Fua & Leclerc 1990) or 3D situations (Nitzberg & Mumford 1990), for tracking objects over time (Kass et al. 1987; Cohen 1991), or for infering 3D structure from the deformation of apparent contours in a sequence of images (Cipolla & Blake 1990). Our interest in this approach was motivated by the need for a fast, interactive tool to assist image interpretation and morphometry in scientific applications, such as medical imaging and remote sensing. Of particular interest to us was the ability to focus on a given object of interest, specified by the user, among large sets of data. However, active contours all use an edge-based definition for object boundaries, and we felt the need to extend them to region-based definitions, which are more appropriate for color or remote sensing image analysis problems, and also provide important clues even in more traditional image analysis problems.

1.3 Proposed Approach

In this paper, we propose algorithms and strategies that generalize the variational approach of active contour models to region-based image analysis. In Fig. 1, we illustrate the action of external forces acting on an energy-minimizing contour model. Figure 1A shows the effect of an edge-based energy criterium, as used in most active contour models. A now classic problem with this approach stems from the fact that the image and gradient functions are not very well-behaved (Cohen 1991; Leitner et al. 1991). Inside regions, both derivatives of either the image or the gradient function vanish, therefore providing no clue to the energy-minimizing process. Around boundaries, a more subtle situation arises, where one derivative (normal to the boundary curve) also vanishes. On a theoretical basis, one should resort to higher-order derivatives (e.g. image curvature) or piece-wise continuous image models (preferably with explicit image discontinuities) to correctly model the boundaries (Leitner et al. 1991). In practical terms, most recent active contour models turn the difficulty by pre-processing the edge-data, e.g. through the use of a distance function.

In Fig. 1B, a region-based criterium is illustrated. Instead of a point-wise edge criterium, we use statistical models of the object region (enclosed by the contour model) and background region. If a homogeneous region against a homogeneous background is anticipated, the forces are defined in the following way. All contour points with a neighborhood that fits the object model are pushed outside by centrifugal forces. Conversely, all contour points with a neighborhood that fits the background model are pulled inside by centripetal forces. Both situations are depicted in Fig. 1B, and it is easily seen that this conjectures an external force field $F(M)$, defined for all image points M on the contour curve, and aligned to the normal N to the contour curve (oriented from object to background in Fig. 1). The force magnitude should also be proportional to the difference of statistical fits to object and background. A convenient notation for this is the following:

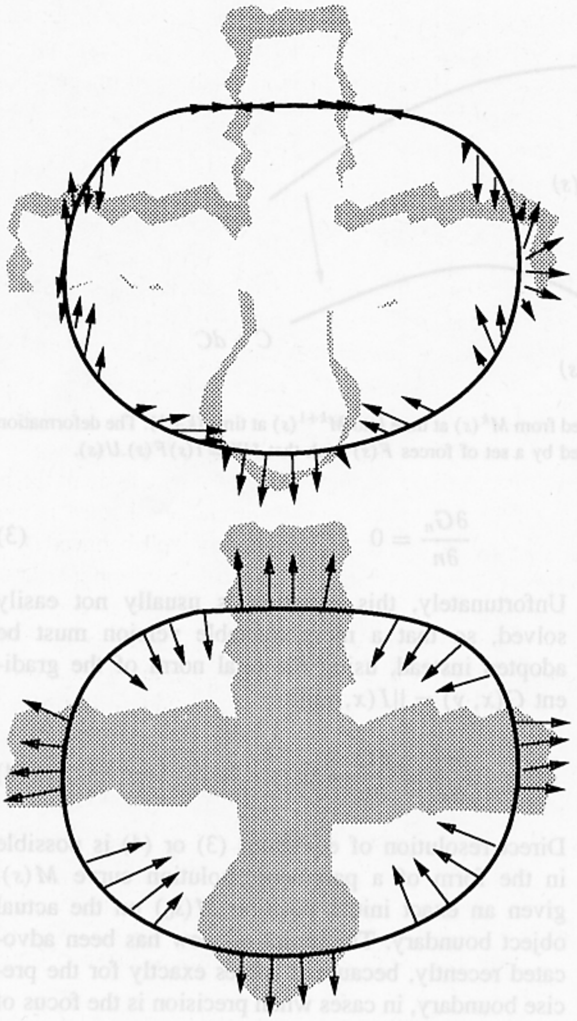

Fig. 1. Edge-based and region-based external forces acting on a contour model. (1A) Edge-based forces. An edge-map is estimated from the original image, and the x- and y-derivatives of the edge-map function are evaluated in every sampled point M. (1B) Region-based forces. Statistical models for the object and background regions are estimated from the original image values and current contour geometry. Statistical fits are evaluated in every sampled point M.

$$F(M) = [\,\text{object}(M) - \text{background}(M)\,]\, N(M) \quad (1)$$

Contrary to edge-based models such as snakes, this external energy cannot easily be derived from a potential energy, because it is only defined along the contour curve, and is therefore not a point-wise function in the image plane. This paper proposes an approach that closely follows the intuitive view of Fig. 1B, while providing a more formal definition of the energies and forces involved, as well as optimization strategies suited to experiment with them. The resulting algorithm, named "anticipating snake," eventually captures the essence of the original snake approach, but in a very different computational setting, owing much to region-contour interaction methods such as simulated annealing relaxation (Geman et al. 1990) and anisotropic diffusion methods

Thus, in the following section of the paper, we will introduce a contrast measure based on a region statistical image model, which will allow us to use region energies to drive a contour model, and an investigation of the local aspects of variational edge detection methods will be presented, showing the advantages and failings of both edge-oriented and region-oriented contour models. Then, the optimization procedure and heuristics necessary to fit contour curves to object boundaries following our local, region-based scheme will be presented in the third and fourth section of the paper. This will include an important discussion on scale-change issues, and it will be shown how region-based active models can be devised to use a scale heuristic (section 3) and a diffusion heuristic (section 4) while they are being optimized. The last section of the paper will allow us to present and discuss an implementation and some results of the method, with application to different kinds of images.

2 Edge and Region Based Contour Models

2.1 Minimum Principles for Contour Models

Using a minimum principle to define the loci of object contours in an image is appealing, because it corresponds to the intuition of the gestalt definition of shape, which is perceived as a stable, minimum configuration of sensed data. Accordingly, minimum principles have been used extensively in vision research to reconstruct shape from lower-level image data. Active contours are a special case of such reconstruction problems. In order to set those models in a general framework, we will refer to the notations of Fig. 2 throughout this paper, whether dealing with snakes or anticipating snakes. Our general formulation is as follows. Given an image and a hypothetical object contour $M(s)$, we define a set of interaction forces on every point in the contour, so that every deformation of the contour can be quantified with an energy transition δW, basically equal to the work of the interaction forces during deformation. Then, any

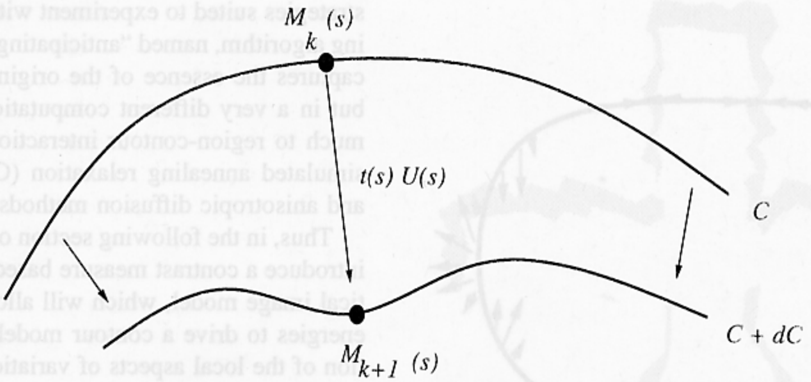

Fig. 2. Deformation of a contour model. Points on the contour are deformed from $M^k(s)$ at time k to $M^{k+1}(s)$ at time $(k+1)$. The deformation is in direction $U(s)$ and has amplitude $t(s)$. The contour model is defined by a set of forces $F(s)$ such that $\delta W = t(s)F(s).U(s)$.

particular contour will be a solution of the contour model if it is a local energy minimum, i.e.

$$\delta W \geq 0 \text{ for all deformations } \delta M(s) \qquad (2)$$

Inversely, when we set out to define a contour model to solve an application problem, we must insure that actual object boundaries are solutions of (2)—this is our first minimum principle. In addition, we should favor models for which all solutions of (2) are indeed object boundaries (no false local minima). This constitutes a second minimum principle. Devising an active contour model, or equivalently a set of interaction forces, meeting the requirements of those two minimum principles is a tremendous task in the general case. If the problem is somewhat restricted, e.g. by use of "proper" initializations from which solutions can be obtained, or with rigid constraints, then useful models can be devised. In the rest of this section, we will first review the method of snakes, which is one such restricted method of practical value, with external forces driven by the image gradient. From there, we will then introduce our region-based method of *anticipating snakes*.

2.2 *Snakes: A Gradient-Based Contour Model*

2.2.1 Overview of the Snakes Method. Capturing the local structure of discontinuities is a difficult process, as illustrated by Leclerc & Zucker (1987), because many different situations can arise. One key definition for local edge modeling is that of the maximal step edge normal to the direction n at point (x, y), where G_n is the image gradient taken in the direction n (Haralick 1984):

$$\frac{\partial G_n}{\partial n} = 0 \qquad (3)$$

Unfortunately, this equation is usually not easily solved, so that a more tractable version must be adopted instead, using the total norm of the gradient $G(x, y) = ||I(x, y)||$:

$$\frac{\partial G}{\partial x} = \frac{\partial G}{\partial y} = 0 \qquad (4)$$

Direct resolution of equations (3) or (4) is possible in the form of a parametric solution curve $M(s)$, given an exact initial position $M(s_0)$ on the actual object boundary. This point of view has been advocated recently, because it solves exactly for the precise boundary, in cases when precision is the focus of image treatment—e.g. for medical applications (Cinquin et al. 1990). But in many other image vision problems, where robustness is the focus of interest, one must resort to regularization techniques to handle equations (3) or (4) efficiently.

The variational method has been introduced to cope with the difficult numerical problems encountered while trying to solve equations (3) or (4) on a local basis only. The interpretation of equations (3) or (4) becomes that of a maximal contrast condition i.e. minimization of gradient-based energies

$$E_{int} - G_n^2\left(M, \frac{\partial M}{\partial s}\right) \text{ or } E_{int} - G^2(M) \qquad (5)$$

relative to small variations or deformations of the parametric curve $M(s)$. The internal energy E_{int} which appears in this minimum principle has the interpretation of a regularizing term such as curvature or arc length. The original snakes algorithm (Kass et al. 1987) uses a linear combination

$$\alpha||\frac{\partial M}{\partial s}||^2 + \beta||\frac{\partial^2 M}{\partial s^2}||^2 \tag{6}$$

which can be expressed using a finite difference scheme in the form of a rigidity/elasticity matrix B_{ij} if the contour curve is sampled into N points, so that the function $M(s)$ can be approximated by the $2N$-dimensional vector $M_i = [(x_i, y_i)i = 1..N]$.

The minimum principle imposes that the energy integral over the whole solution curve $M(s)$ be minimal: as such, it has no immediate local interpretation, although the Euler-Lagrange method can be used to transform the global, compound energy minimization back into a differential equation in M, $\frac{\partial M}{\partial s}$ and $\frac{\partial^2 M}{\partial s^2}$ (Kass et al. 1987). Using equations (5) and (6), and the notations of Kass et al. (1987) and Szeliski & Terzopoulos (1989), a finite difference scheme can be used to express the total energy as a quadratic form of the (discretized) position vector M (see appendix):

$$E = \frac{1}{2}[{}^tMBM - {}^tM\nabla G^2] \tag{7}$$

A natural solution for variational contour optimization is the classical gradient descent strategy. Clearly, the gradient of expression (7) is $BM - \frac{1}{2}\nabla G^2$, so that a step from M^k to M^{k+1} can be taken at time T^k such that

$$M^{k+1} = M^k - t\left[BM^k - \frac{1}{2}\nabla G^2\right] \tag{8}$$

Equation (8) introduces a constant step-size t which controls the rate of deformation of the algorithm, and plays an important role in all active contour optimization schemes. A variation on this theme is suggested by (Kass et al. 1987), resulting in a two-step, semi-implicit scheme with faster convergence, because t can then be chosen arbitrarily large while equation (4) imposes the condition $t < \frac{1}{min(B)}$. This yields a powerful treatment of internal energies (as should be expected), but leaves many pratical issues unanswered as far as external energies are concerned (those aspects of the snakes method have been detailed and discussed in Fua & Leclerc (1990) and Amini et al. (1990).

2.3 Anticipating Snakes: A Region-Based Contour Model

2.3.1 Region Statistics and Image Models. Because the method of snakes relies entirely on its potential energy $E(x, y)$, its domain of application has remained limited. One limitation is that it is sometimes not possible to provide such a function, because of image non-stationnarity. Another limitation is that regional criteria such as color or texture cannot easily be integrated into a potential function (unless a segmentation of the image can be provided). Variational methods have been described, which incorporate gradient and region criteria into a single energy function (Mumford & Shah 1985; Grossberg 1987; Geman & Geman 1984; Marroquin et al. 1987; Geman et al. 1990; Shah 1990). Those methods differ fundamentally from snakes, because they attempt to model the image intensity function, as well as its object boundaries. Although their theoretical importance is enormous, none has led to significant practical solution, most notably because of mathematical difficulties documented in e.g. Mumford & Shah (1989). In this paper, we take a less rigorous approach, drawing mainly on heuristic solutions, in order to show that region criteria can indeed be used to guide a contour model, with a quality of results comparable to that of snakes or related models. In our anticipating snakes method, we replace energy criterions such as (5) by other photometry functions, taking into account the local partition of the image into an object region and a background region. An intuitive interpretation of the approach will first be presented, and then expanded to a complete mathematical treatment suitable for use in the rest of the paper.

The basic idea is that a given closed contour model $C = \{M(s), s \text{ in } [0, 1]\}$ partitions the image plane into an inside (or object) region and an outside (or background) region which, along with a statistical model, may be fitted to the image data. Thus, an alternative to gradient-based schemes will consists in choosing local changes from $M^k(s)$ to $M^{k+1}(s)$ in a given direction $U(s)$, with amplitude $t(s)$, such that the new partition improves the fit. An intuitive implementation of this approach would use separate statistical models $I_{object}(x, y)$ and $I_{background}(x, y)$. Those models would be tested against the given image data $I(x, y)$, based on a mean-square-error criterion:

$$MSE = \int_{object} ||I_{object}(x,y) - I(x,y)||^2 dxdy$$

$$+ \int_{background} ||I_{background}(x,y) - I(x,y)||^2 dxdy \quad (9)$$

While (9) readily takes on the interpretation of an energy, it still cannot satisfy our minimum principles, because *any* arbitrary contour can be a local minimum in homogeneous parts of the image. We therefore need a more constrained energy definition, so that energy minima occur only at actual image boundaries.

2.3.2 A Region-Based Energy Model for Active Contours. In this section, we will use an image contrast measure best known in the context of split-and-merge methods as the fusion energy of the regions (or their Ward distance) (Beaulieu & Goldberg 1989). Using the notations and conventions of Fig. 3, we define the region energy $W^{region}(R_k)$ of a given region R_k as in Leclerc & Zucker (1987), Beaulieu & Goldberg (1989), i.e. the image functions $I(x,y)$ (intensities, colors or textures) are approximated as a linear combination of basis functions $K_i(x,y)$, using a least squares scheme:

$$I(x,y) = \int_{i=1}^{n} a_{ik} K_i(x,y) + \Delta_k I(x,y) \quad (10)$$

The region energy is defined as the sum of squared errors $\Delta_k I(x,y)$, with subscript k to remind us that the error-of-fit function $\Delta_k I$ depends on the region described:

$$W^{region}(R_k) = [I R_k||\Delta_k I(x,y)||^2 dxdy] \quad (11)$$

This scheme can be devised to fit any possible set of basis functions $K_i(x,y)$, according to the same least squares rule. We then proceed to define the contour energy of a closed curve C, bounding an internal region R_{in} and an external region R_{out}, as depicted in Fig. 4A. We want to derive the contour energy from neighborhood region energies such as in (6). A simple solution to this classical problem is to introduce the union $R_{in} + R_{out}$. We can then write the classical Ward distance between regions R_{in} and R_{out}

$$D[R_{in}, R_{out}] = W^{region}(R_{in} + R_{out}) - W^{region}(R_{in}) - W^{region}(R_{out}) \quad (12)$$

This is defined by Beaulieu & Goldberg (1989) as the energy needed to merge the two regions R_{in} and R_{out} We are thus entitled to interpret it as the energy needed to disrupt the bounding contour C between the two regions, and we set $W^{contour}(C) = D[R_{in}, R_{out}]$. Such a contour energy can be interpreted as the amount of region energy absorbed (or explained) by the contour curve C at steady-state.

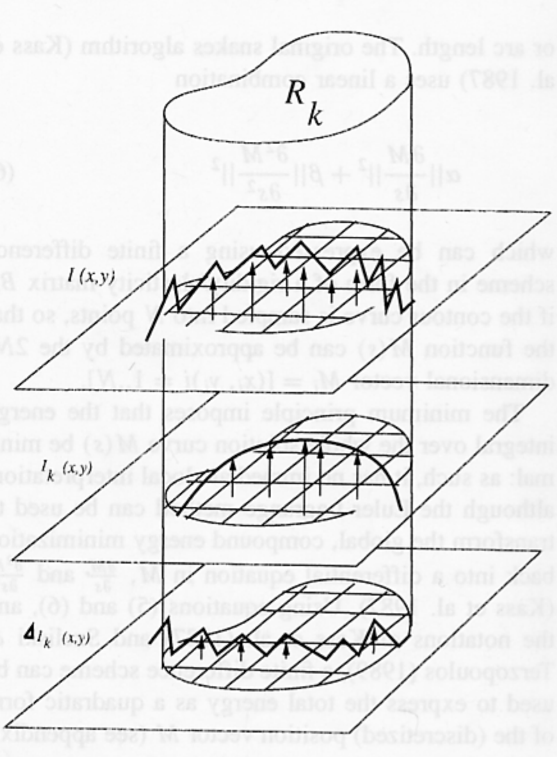

Fig. 3. Notations for image regions. Image intensity $I(x,y)$ is approximated by a region model $I_k(x,y)$ specific to region R_k. The error defines a new image function $\Delta I_k(x,y)$.

We now consider a deformed version $C + \delta C$ of the original contour, and proceed to compute the variation δW of the contour energy during deformation. To make further developments easier, we introduce two smaller regions δR_{in} and δR_{out}, corresponding to outwards and inwards deformations respectively, as in Fig. 4B. Note that the inside (respectively outside) region is now $R_{in} + \delta R_{out}$ (respectively $R_{out} + \delta R_{in}$) just before deformation, and $R_{in} + \delta R_{in}$ ($R_{out} + \delta R_{out}$) just after deformation. Since the union $R_{in} + R_{out} + \delta R_{in} + \delta R_{out}$ is not changed by the deformation, we only need consider the negative terms in (7). We thus write the contour energies as

$$W^{contour}(C) = W_0 - W^{region}(R_{in} + \delta R_{out}) - W^{region}(R_{out} + \delta R_{in})$$

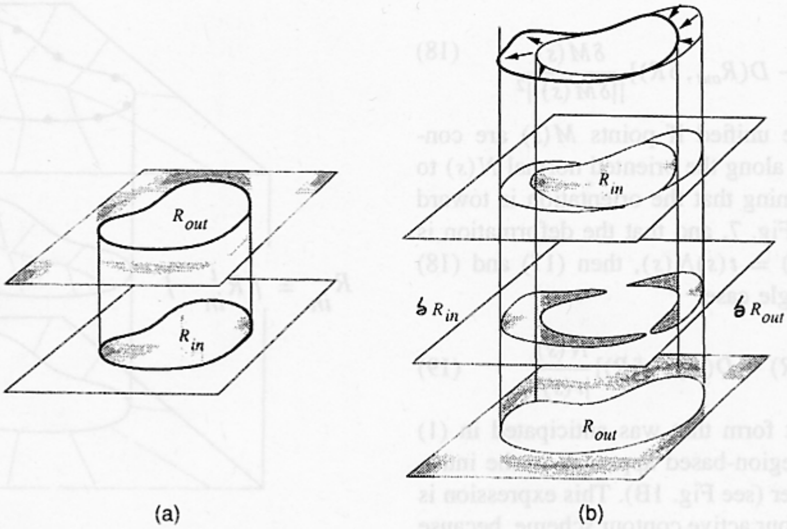

Fig. 4. Regions around a contour model. (4A) Inside and outside regions for a contour model at steady-state. (4B) Inside and outside regions during a deformation. R_{in} and R_{out} are defined as those regions which remain entirely inside or outside the curve during all deformations. The actual regions inside and outside the curve differ from those by δR_{in} and δR_{out}.

$$W^{contour}(C+\delta C) \;=\; W_0 - W^{region}(R_{in}+\delta R_{in}) - W^{region}(R_{out}+\delta R_{out}) \quad (13)$$

The energy for unions of disjoint regions can be further developed into a sum of the individual region energies and their Ward distance, thus

$$W^{region}(A+B) = W^{region}(A) + W^{region}(B) + D(A,B) \quad (14)$$

Using (14), we can transform (13) so that all individual region energies are cancelled out when we compute the difference $\delta W = W^{contour}(C+\delta C) - W^{contour}(C)$, and only Ward distances remain, thus

$$\delta W = [D(R_{in},\delta R_{in}) - D(R_{out},\delta R_{in})] + [D(R_{out},\delta R_{out}) - D(R_{in},\delta R_{out})] \quad (15)$$

When applied locally, this expression becomes even simpler, because local deformations will be either expanding ($\delta R_{out}=0$) or retracting ($\delta R_{in}=0$), so that only one of the two terms in (15) need to be evaluated locally. More specifically, we now study the case when an infinitesimal deformation $\delta M(s)$ is applied to a point $M(s)$ along the curve. The situation is depicted in Fig. 5 for the two possible cases of expansion and retraction. From (15) we can write the force $F(s)$ acting on the point $M(s)$, such that

$$\delta W = F(s).\delta M(s) \quad (16)$$

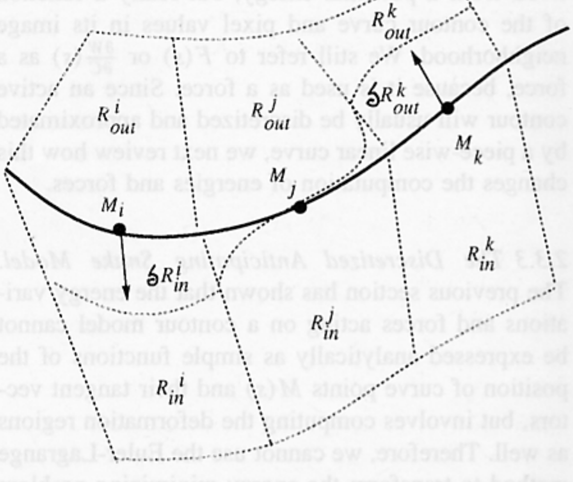

Fig. 5. Expansion and retraction of a contour model. Local interior and exterior regions are shown for each case.

In the case of a retraction, we find that

$$F(s) = [D(R_{out},\delta R) - D(R_{in},\delta R)]\frac{\delta M(s)}{||\delta M(s)||^2} \quad (17)$$

is a solution of (16), as can be easily checked out. Similarly, a solution of (16) for the case of an expansion is

$$F(s) = [D(R_{in}, \delta R) - D(R_{out}, \delta R)] \frac{\delta M(s)}{||\delta M(s)||^2} \quad (18)$$

Both cases can be unified if points $M(s)$ are constrained to deform along the oriented normal $N(s)$ to the contour. Assuming that the orientation is toward the outside, as in Fig. 7, and that the deformation is in the form $\delta M(s) = t(s)N(s)$, then (17) and (18) simplify to the single case

$$F(s) = [D(R_{in}, \delta R) - D(R_{out}, \delta R)] \frac{N(s)}{|t(s)|} \quad (19)$$

which is the exact form that was anticipated in (1) for the idealized region-based approach in the introduction to this paper (see Fig. 1B). This expression is very important for our active contour scheme, because it makes explicit the energy variations as a function of deformation, in a computationally attractive way. In this paper, we will use the notation $\frac{\partial W}{\partial C}$ to denote the pseudo-force defined by (19). This is to reminds us that the pseudo-force is not an external field derived from a potential energy, but really a function of the contour curve and pixel values in its image neighborhood. We still refer to $F(s)$ or $\frac{\partial W}{\partial C}(s)$ as a force, because it is used as a force. Since an active contour will usually be discretized and approximated by a piece-wise linear curve, we next review how this changes the computation of energies and forces.

2.3.3 The Discretized Anticipating Snake Model. The previous section has shown that the energy variations and forces acting on a contour model cannot be expressed analytically as simple functions of the position of curve points $M(s)$ and their tangent vectors, but involves computing the deformation regions as well. Therefore, we cannot use the Euler-Lagrange method to transform the energy-minimizing problem into a differential equation. This, of course, is due to the fact that the region derivatives (or Ward distances) depend on the image data in a more intricate fashion than in the case of snakes. Accordingly, it is preferable to use a discretized version of the contour, and express the local forces in this framework, much in the same way as Kass et al. (1987) derived their simpler equations using a finite difference approximation.

Our next step will be to partition the object and background neighborhoods of the contour into small local neighborhoods R_i, following a discretization of the contour curve itself, as in Fig. 6. For the purpose of prototyping our approach, we used a neighborhood structure described in Fig. 7 and section 3 below. We now write the energy as a sum of contributions from every contour point:

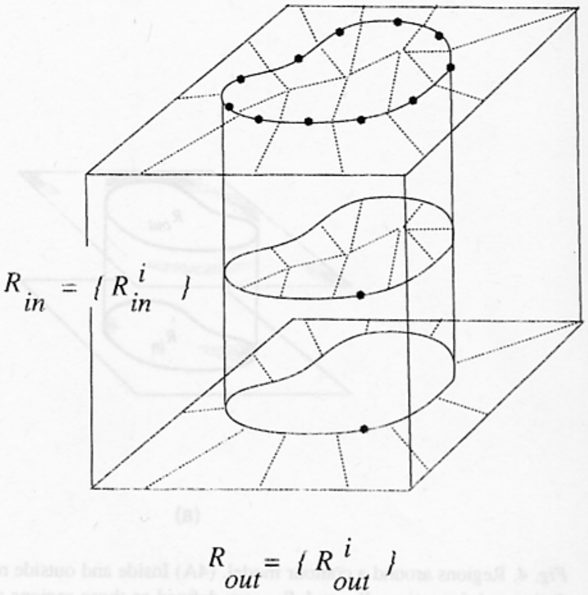

Fig. 6. Discrete regions of deformation. All deformations of point $M(i)$ are confined to its inside and outside regions.

$$W^{contour} = W_0 + \sum_i W^i_{INT}(M^i) - \sum_i [W^{region}(R^i_{in}) + W^{region}(R^i_{out})] \quad (20)$$

where $W^i_{INT}(M^i)$ is an internal, elastic energy, as defined in Kass et al. (1987). This approximates the real energy field whenever local neighborhoods partition the entire image plane, as in Fig. 6. Smoothness and rigidity terms can be transformed into finite differences as in (3), (16), Kass et al. (1987). External forces can still be computed as in (19), but using only terms belonging to one particular point M^i, so that the local dissipated energy during a deformation δM^i amounts to:

$$\left[\frac{\partial W^i}{\partial C} + \sum_{j \neq i} B_{ij} M^j\right] \delta M^i \quad (21)$$

Practical implementations of (21) can vary greatly, e.g. region statistics can be modeled separately for each point M^i, hardware can be used to produce the goodness-of-fit functions, or the image function itself can be approximated continuously and calculations

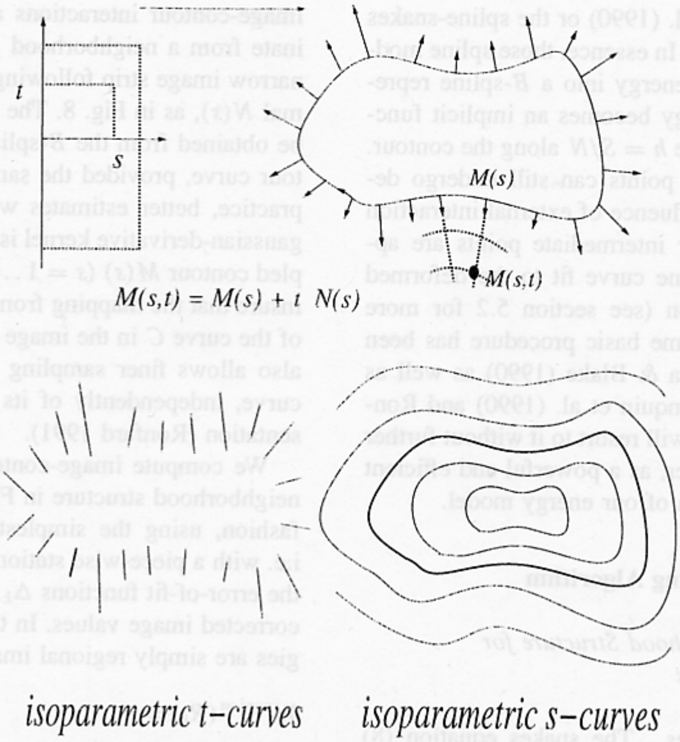

Fig. 7. Local coordinates. Choosing the normal direction for all deformations defines a local system of coordinates (s, t). Curves of constant t are the directions of deformation. Curves of constant s are iso-deformation curves. The real deformation will be $M(s) + t(s)N(s)$.

performed analytically. In this paper, we present the first (and simpler) solution.

2.4 *Internal Energy and Spline Models*

Following Kass et al. (1987), we have written explicitly the internal energy terms in (7) and (21). We are now going to cancel those terms out, and use simpler kinds of contour models with external forces only. The reason is the following. In the regularization approach to snakes, a key issue lies in the choice of a common scale unit between internal and external energy terms (Ronfard 1990). Internal energies were first introduced by Kass et al. (1987), drawing on the theory of approximating spline functions (Laurent 1972), which have classically handled the difficulty by use of cross-correlation methods (Sharahray & Anderson 1989). No such solution extends to the case of equations (17) or (22), because of the strongly non-linear image forces (Ronfard 1990). This is therefore a very difficult problem, and one that this work does not try to address.

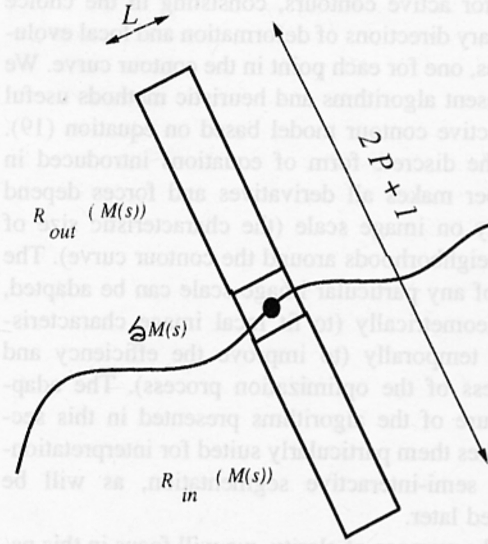

Fig. 8. A simple local structure for variational edge detection. We approximate local coordinates (s, t) with a discrete set $s = 1 \dots NL$, $t = -P \dots P$, i.e. a local image grid (s, t).

In the following, we will use instead a simplified version of contour models, similar to the *B*-snakes described in Menet et al. (1990) or the spline-snakes in Cinquin et al. (1990). In essence, those spline models imbed the internal energy into a B-spline representation. Internal energy becomes an implicit function of the sampling rate $h = S/N$ along the contour. Control points or knot points can still undergo deformation, under the influence of external interaction forces alone, and other intermediate points are approximated as a B-spline curve fit to the deformed control or knot polygon (see section 5.2 for more details on this). The same basic procedure has been reported also by Cipolla & Blake (1990) as well as Menet et al. (1990), Cinquin et al. (1990) and Ronfard (1990), so that we will resort to it without further justification in this paper, as a powerful and efficient short-cut for illustration of our energy model.

3 Local Depth-Adapting Algorithm

3.1 A Simple Neighborhood Structure for Anticipating Snakes

3.1.1 Energy and forces. The snakes equation (8) establishes a global evolution rule for all sampled points on an active contour model. The first part of this paper has been devoted to present a different approach for active contours, consisting in the choice of arbitrary directions of deformation and local evolution rules, one for each point in the contour curve. We now present algorithms and heuristic methods useful for an active contour model based on equation (19). Using the discrete form of equations introduced in this paper makes all derivatives and forces depend explicitly on image scale (the characteristic size of image neighborhoods around the contour curve). The choice of any particular image scale can be adapted, either geometrically (to fit local image characteristics) or temporally (to improve the efficiency and robustness of the optimization process). The adaptive nature of the algorithms presented in this section makes them particularly suited for interpretation-guided, semi-interactive segmentation, as will be illustrated later.

For the purpose of clarity, we will focus in this paper on the very simple neighborhood structure shown in Fig. 7 and Fig. 8, which can be easily and efficiently implemented. It consists in separate, non-overlapping L-pixel-wide bands centered on sampled points $M(s)$. These structures are $(2P+1)$ pixel-deep and directed along the normal to the curve $N(s)$. All image-contour interactions at point $M(s)$ thus originate from a neighborhood $M(s) + t(s)N(s)$, in the narrow image strip following the estimated local normal $N(s)$, as in Fig. 8. The normal vector $N(s)$ may be obtained from the B-spline expansion of the contour curve, provided the sampling is not too fine. In practice, better estimates will be obtained if a 1-D gaussian-derivative kernel is convolved with the sampled contour $M(s)$ $(s = 1 \ldots N)$. This is important to insure that the mapping from (s, t) to a neighborhood of the curve C in the image plane is one-to-one. This also allows finer sampling intervals on the contour curve, independently of its original B-spline representation (Ronfard 1991).

We compute image-contour interactions from the neighborhood structure in Fig. 8 in a straightforward fashion, using the simplest form of equation (10), i.e. with a piece-wise stationary image model, so that the error-of-fit functions $\Delta_k I$ are simply the average-corrected image values. In this case, the region energies are simply regional image variances

$$W^{region}(R_k) = \int_{R_k} ||I(x, y) - \langle I\rangle_k||^2 dxdy \qquad (22)$$

and the contour energy is just the classical (stationary) Ward distance between regions R_{in} and R_{out}, i.e.

$$W^{contour}(C) = D[R_{in}, R_{out}] \qquad (23)$$

The general expression for the image forces acting on the contour curve during a step-wise deformation $t(s) \rightarrow t(s) + \Delta t(s)$ then becomes

$$\frac{\partial W}{\partial C} \approx \int_0^1 (D_{in}[M(s)] - D_{out}[M(s)]) \frac{ds}{|\Delta t(s)|} \qquad (24)$$

where $D_{in}(M(s))(resp.D_{out}(M(s)))$ denotes the Ward distance between $R_{in}(M(s))(resp.R_{out}(M(s)))$ and the L-pixel deep region $\delta M(s)$ around $M(s)$, as shown in Fig. 8. This is the simplest possible implementation of equation (21), and it has the interesting properties that the local deformation state of the contour curve is completely described by the parameter $t(s)$, and image forces are local to the neighborhood structure $M(s) + t(s)N(s)$, as prescribed.

3.1.2 Computational Framework. We can now present a computionally simple variation scheme, based on the above discussion, as a candidate to solve the active contour optimization problem. For each iteration step of the procedure, we have the following sequence:

- Compute normals $N(s)$ for all sampling points $M(s)$, $s = 1 \dots N$, and build the image statistics in the neighborhood image regions.
- Starting from undeformed points $M(s) = A(s, 0)$ take as many steps as possible in either one of the deformation directions $M(s) + t(s)N(s)$ or $M(s) - tN(s)$, $t(s) = 0 \dots P$, as long as the power of the external forces remain positive

$$D_{in}[M(s)] - D_{out}[M(s)])\Delta t(s) > 0 \quad (25)$$

- Compute a new contour from all deformed sampled points, using them as control points for an regular cubic B-spline curve. Such a curve will *not* interpolate sampled points, but merely approximate them with the best-looking piece-wise cubic curve, in a certain sense (Laurent 1972). This is sufficient for our purpose, because we re-sample contour points in every iteration.

In order to proceed successfully with such a simple scheme, several issues have to be dealt with. First, a number of parameters remain unspecified, i.e. the maximum depth P allowed for step-wise deformations in a single iteration, the sampling rate N of points along the curve, the gaussian variance σ used to estimate smooth normals along the curve. Along with neighborhood width L, those parameters determine all scale choices for our model (see Fig. 9). In contrast to the original snakes (see Fig. 10), those parameters are all local to the iterative scheme, and can therefore be adapted more easily than a filter size for edge detection. Indeed, experience has shown that all scale parameters should be modified during the optimization process. Thus, heuristic methods for determining N and P in each step of the procedure will be presented here (we introduce no such refinement for σ, which we in fact consider as a function of N and P), while the last section of the paper will be devoted to the width scale parameter L.

A different line of problems arise from the choice of the Ward distance. We use the sign of $\Delta t(s)$ $(D_{in}[M(s)] - D_{out}[M(s)])$ in order to determine in which direction the curve should be deformed. It would be comforting to know that this expression vanishes on boundary points. This is not the case, however, and section 3.3 discusses transformed distance functions with a better behaviour in that respect.

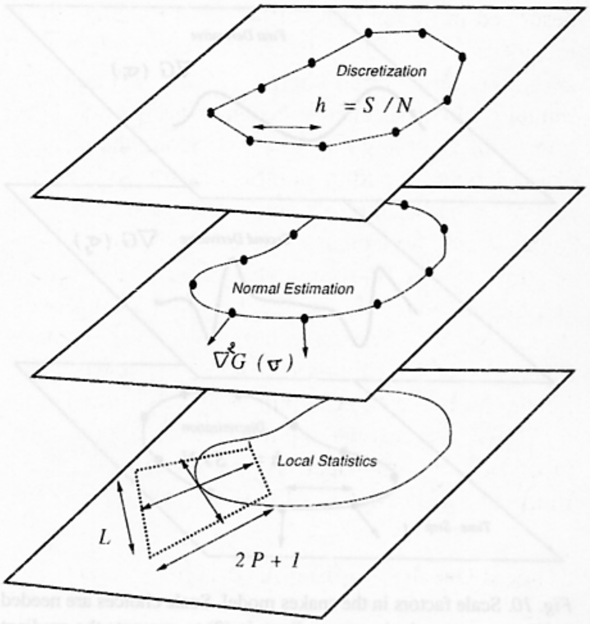

Fig. 9. Scale factors in the anticipating snakes model. Scale choices are needed in (1) to discretize the contour; in (2) to estimate smoothed normals along the contour; in (3) to decide image neighborhood size (depth $2P+1$ and width L).

3.2 Local Depth-Adapting Strategy

3.2.1 Scale-Space Strategies. The point in this section is to adapt neighborhood size P (see Fig. 8 and Fig. 9) to the scale at which the object-to-background contrast is maximal. A compromise must therefore be found between smaller neighborhoods (for which the norms have no statistical significance) and larger neighborhoods (where more than two regions are in the scope of $M(s) + t(s)N(s)$). The second term of the alternative represents the most difficult situation, because all the equations used to model edges assume two regions (possibly identical) only. This makes the traditional coarse to fine, scale-space approach advocated by Kass et al. (1987) very dangerous, since the local forces acting from the image at coarser scales may become unfounded and misleading, and no verification can be made at finer scales.

This failing is illustrated by the dominant role played in such cases by the automatic internal forces of snakes or other related models (elastic forces, pressure). In such cases, active contours are driven away from their target object boundaries, without being given any chance to recover them (since only finer scales will subsequently be examined, to improve—not correct—the first coarser optimization steps). Fol-

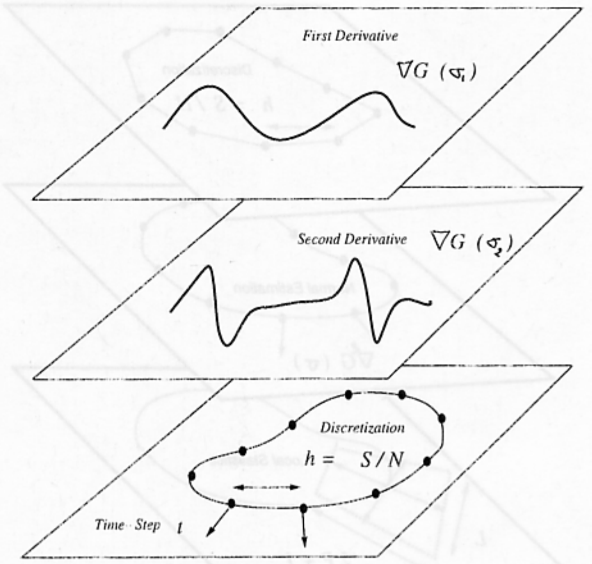

Fig. 10. Scale factors in the snakes model. Scale choices are needed in (1) to compute the image gradient; in (2) to compute the gradient derivatives in x and y; in (3) to discretize the contour and choose uniform step t.

lowing Leclerc & Zucker (1987), we have investigated the opposite, fine to coarse strategy. This approach starts with very small neighborhood structures ($P = 1, 2, 3 \ldots$) even though a high level of noise may be present at such scale. Noise in earlier stages of optimization can be dealt with because the contour curve is regularized, and the steps taken in each iteration are very small ($t_{max} < P$). Then P is increased while optimizing, until all neighborhoods become statistically significant. The optimal deformation state is then easily obtained in a few iterations.

3.2.2 Depth-Adapting Algorithm. This strategy captures the intuitive idea behind active contour models, that optimization should use local image analysis whenever available (in close neighborhoods of object boundaries) and internal cohesion forces when the image profiles are locally flat. The fine to coarse approach also offers an efficient heuristic for increasing the dependency of the variational procedure on image characteristics, since object boundaries are only a finite distance away from the initial contour position, and will be reached in a finite number of steps—without oscillations. This of course assumes a control mechanism for stopping at interfering objects appearing in the background, as the scope of the neighborhood structures extends. All of the above observations can now be summed up in the following depth-adapting algorithm:

[1] Start with an initial B-spline curve, sampled uniformly after arc-length s. Then choose an initial depth parameter $P = P_0$ (preferably $d < P_0 < D$ where d is the min-distance from the initializer to the target object and D is the min-distance between objects in the image).

[2] Iterate the basic steps : fit all control points in the B-spline basis, compute normals $N(s)$ at control points $M(s)$, extract $(2P + 1)$ image pixels in the direction of $N(s)$ for every s, using a fast line drawing algorithm such as Bressenham's, move all control points step-wise from $M(s)$ to $M(s) + tN(s)$, $-P < t < P$, according to the sign of the dissipated energy. This is simply the work of the external pseudo-force from equation (19).

[3] Increment P, and compare energy levels obtained in step [2] with those of the increased neighborhoods, allowing only lower energies at coarser scales. All points with increasing energies at this stage should become inactive or attached.

[4] Iterate steps [2] and [3] until no more control points are active.

An application of this algorithm to a brain tomography scan is presented in Fig. 11. The initial curve is an interactively designed B-spline, sampled every five pixels in a 256 × 256 image. Ten iterations were performed before convergence, with increased depth ranging from five to fifteen pixels on each side of the curve. Since all objects in this image are isolated and offer a roughly constant contrast to their background, the algorithm performs well in this case.

3.3 Controlling Stability

When computing the difference of the Ward distances $D_{in}[M(s)]$ and $D_{out}[M(s)]$, we simply check for its sign, in order to determine in which direction the curve should be deformed. This raises a difficult problem for stability issues, since deformations will always occur, regardless of the magnitude of the energy term. We would like to have a threshold value here to tell us when the external forces acting on the curve are so small as too be negligible. This is not possible within the Ward distance formulation, however, because it does not capture the intuition that external forces smoothly vanish around real object

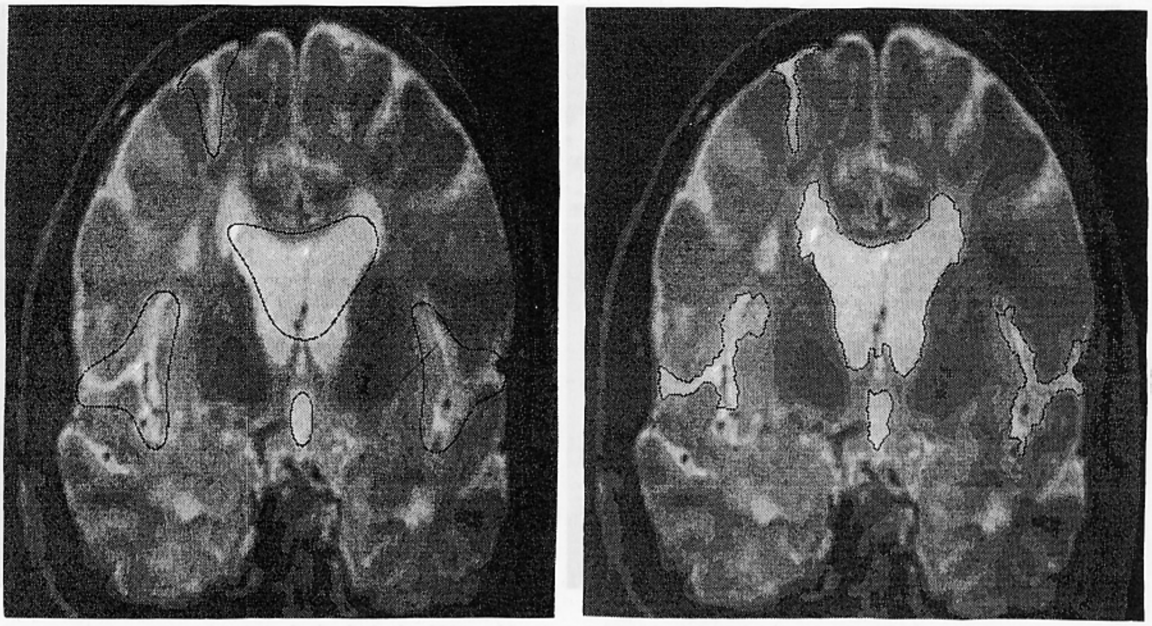

Fig. 11. Application of depth-adapting algorithm to medical imaging segmentation.

boundaries. Instead, it is easily seen that the force magnitude $D_{in} - D_{out}$ may be as great as the Ward distance $[R_{in}, R_{out}]$ in the vicinity of a highly contrasted edge, whereas regions without edges can have very low $D_{in} - D_{out}$ altogether. In such cases, the threshold value would have all optimal points move in and out around an object boundary, while those points that are far from optimum remain in a fixed (and erroneous) position. In order to avoid this undesirable behaviour, we can use the sign and magnitude of other functions of the involved Ward distances. For example, we can use

$$\frac{D_{in}[M(s)] - D_{out}[M(s)]}{D[R_{in} R_{out}]} \tag{26}$$

which appears as a formal derivative of the logarithm of the Ward distance, and enhances the forces in regions where the overall contrast is low, so that points in those areas can be moved away. Another possibility is

$$[D_{in}[M(s)] - D_{out}[M(s)]]e^{-kD[R_{in} R_{out}]} \tag{27}$$

which can be interpreted (loosely) as the derivative of the exponential function of the Ward distance. This also has the desirable effect of boosting the forces in regions away from real boundaries, while having them vanish at higher-contrast edges. Since the exponential in (27) is better-behaved than the ratio in (26), we have used (27) in our implementation.

The use of the exponential further brings a nice statistical interpretation, since $e^{-kD[R_{in} R_{out}]}$ is the Boltzmann distribution for energies $D[R_{in} R_{out}]$, meaning that it is (up to some factor) the probability of the image intensity at M, given its neighbor pixels and the hypothesis of the contour curve C passing through M. Let's denote this conditional probability $P(I(M)|C \text{ in } M)$. It is classically related to the probability of the curve C passing in M, conditionally to the observed intensity $I(M)$, by the Bayes rule which we write here:

$$P(C \text{ in } M|I(M)) = \frac{P(I(M)|C \text{ in } M)P(C \text{ in } M)}{P(I(M))} \tag{28}$$

so that, taking both prior probabilities $P(C \text{ in } M)$ and $P(I(M))$ to be uniform, our evolution rules can be shown to solve for the most probable contour curve position, given the image intensity function $I(M)$ (Ronfard 1991).

Expressions (26) and (27) are very useful to enhance the performances of our variational scheme. We thus substitute one of the expressions (26) or (27) into the computation of image interaction forces. This alteration of our variational definition for optimal im-

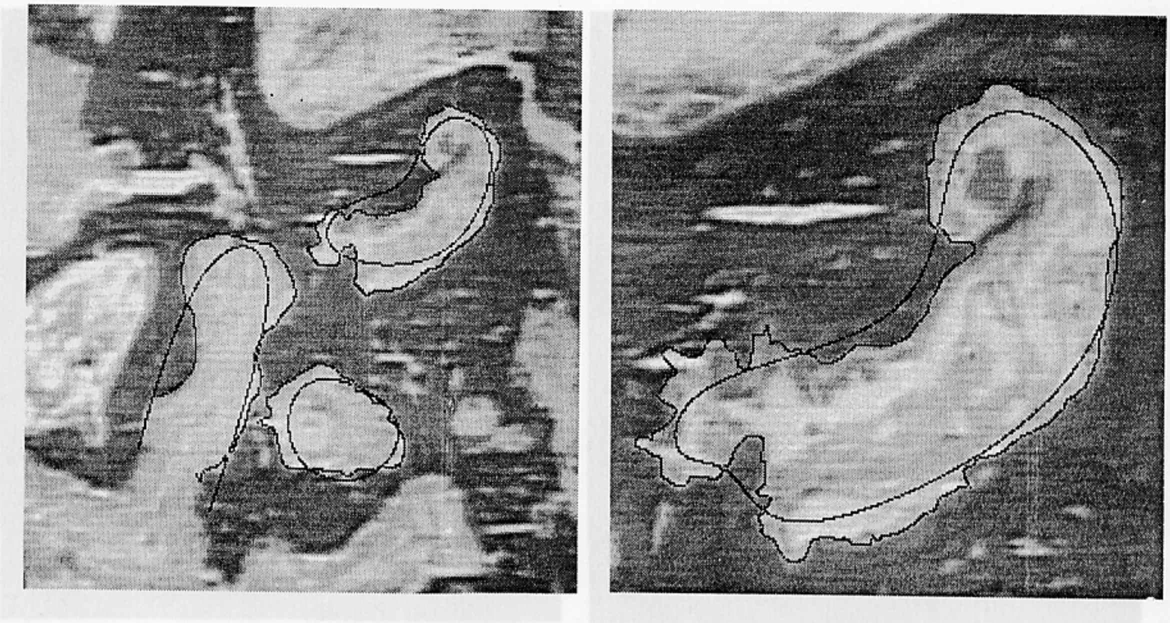

Fig. 12. Application of depth-adapting algorithm to histological morphometry.

age contours makes it remarkably similar to probabilist, non-determinist schemes such as described in Geman et al. (1990), where the exponentials of a quadratic contrast function play a central role.

Our second example (presented in Fig. 12) is a case where direct application of the depth-adapting algorithm fails if we do not use (27), because of the texture in the lighter background. Those bone biopsy images were digitized and contours extracted at interactive rate from a photo-microscope. Because the internal structure in the background has a lower contrast, using (27) results in a correct segmentation in very few iterations. Sampled points on the bone boundary are easily stabilized, since their energy transitions are scaled down by the exponential. Sampled points falling in the background are then more easily moved to the boundary, as the search depth is increased to reach higher contrasted edges.

4 Adaptive Diffusion Algorithm

4.1 *Failings and Extensions of the Depth-Adapting Algorithm*

Contours obtained with the algorithm presented in section 3 are usually comparable in precision and quality to other snakes (Kass et al. 1987), balloons (Cohen 1991) or model-driven detectors (Fua & Leclerc 1990), although it is difficult to substantiate such an affirmation. One critical element in our approach is the choice of the initial depth and discretization scales. Also, a snake can be used to fill-in missing data from the edge-map, or even occluded contours from 3D-objects, through its internal energy. Our method does not have this feature. More importantly, it usually fails in cases when too many object boundaries are present (experiments not presented), because each local part of the curve sticks to its own optimum, with very little global interaction between different sampling points, except in the spline fitting step.

This section introduces a different algorithm, based on a diffusion method, rather than a purely local scale-adapting method. More precisely, we show how the framework that we have established enables us to control some sort of consistency between locally optimized neighborhoods $M(s) + t(s)N(s)$ along the curve parameter s, so that alien boundary parts can be dismissed while consistent parts are inforced. This is intended as a solution to some of the initialization problems, as well as an interesting methodological shift from a purely local scheme, to a more global approach. While snakes and most other active models have used internal energy to provide global control, we choose to propagate the external energy, in a sense that will be made clear later.

4.2 A Diffusion Heuristic

The depth-adapting algorithm that we have presented and illustrated in our previous sections can be described as a set of automata, placed at regular intervals along the contour model, which detect possible object boundaries in a given direction, and move to those anticipated boundaries. Those moves determine the global deformation of the contour model. Each move is based on the Ward distance between the automaton and its local estimates of intensities or colors inside and outside the contour curve. We have made no attempt so far toward a cooperation between adjacent automata, and this of course results in a rather poor modeling power, since the global behaviour of our contour model only depends on (1) the nearest edge element to each automaton in the direction of its normal, and (2) the shape of the contour curve, because of the spline-fitting step which tends to pull its points towards its centers of curvatures. Without an intuition of what object shapes should be, this does not provide a robust detection method, except in regions where only one object is present.

In order to obtain a more satisfying model, especially in cases where several objects are present around the initial contour model, we are faced with the alternative, either to allow multiple contour models, i.e. deformable models consisting of several closed contours (one for each object), or to enhance a single contour model, so that it can discriminate between different object boundaries, and converge toward a single object. We will not discuss further the first approach, because it seems difficult to restrict it to the simpler cases of contour models having disjoint interiors, without introducing nested structures (Koenderink & van Doorn 1979) or overlapping contours (Nitzberg & Mumford 1990). In both those cases, contour models become much more complex. In the second approach, we only need to enforce new constraints on the contour model, e.g. impose that the regions bounded by the contour curve be of homogeneous intensity or color, at least in the neighborhood of the contour model. This will cause the contour model to favour consistent boundaries, and selectively converge towards a single object. We now have to find an efficient way of enforcing such constraints.

This will be easier to illustrate in terms of a decision network. Instead of a set of independent automata, we now want a network of such automata, so that each one can base its dynamics on estimates of intensities or colors provided by its nearest neighbors as well. As an example, Fig. 13 shows how local estimates $\langle I\rangle_{in}$ and $\langle I\rangle_{out}$ can be propagated along the contour curve, and new estimated values $\langle I^*\rangle_{in}$ and $\langle I^*\rangle_{out}$ can be formed from this mechanism. Since each automaton computes its own estimates, it is a simple matter to connect them so that they share those values. In this section of the paper, we will show how such a simple diffusion mechanism can be used to enhance the modeling power of our anticipating snake model.

In order to compute the second-order estimates $\langle I^*\rangle_{in}$ and $\langle I^*\rangle_{out}$, we use combinations of the first-order estimated image intensities $\langle I\rangle_{in}$ and $\langle I\rangle_{out}$, typically using a one-dimensional gaussian weight function $g(j)$ as shown in Fig. 13 and Fig. 14

$$\langle I^*\rangle^i_{in} = \sum_{j=-L}^{L} g(j)\langle I\rangle^{i+j}_{in}$$

and

$$\langle I^*\rangle^i_{out} = \sum_{j=-L}^{L} g(j)\langle I\rangle^{i+j}_{out} \qquad (29)$$

In (29), indices i and j are in the range of the discrete arc length s across C. This yields a representation of the neighborhood structure extended in width as two layers of inter-connected cells, as depicted in Fig. 14 below, while retaining the local decision structure described above. The input cells are image pixel values, extracted in the estimated normal direction $N(s)$. The first layer of cells then computes local estimates of the image values in half-neighborhoods, while the second layer averages those values along the curve parameter. Finally, output cells perform nonlinear transforms and local comparisons to determine the directions of deformation, much in the same way as before.

4.3 Adaptive Diffusion

With this new definition for our local neighborhood structure, we have introduced a new parameter L, which controls the width of the propagation process, i.e. typically the number of inter-connected points on the contour curve, or more intuitively, the width of the second-order neighborhood structure itself. Just as in our previous sections, we need to decide for a control strategy for this new parameter. This will be

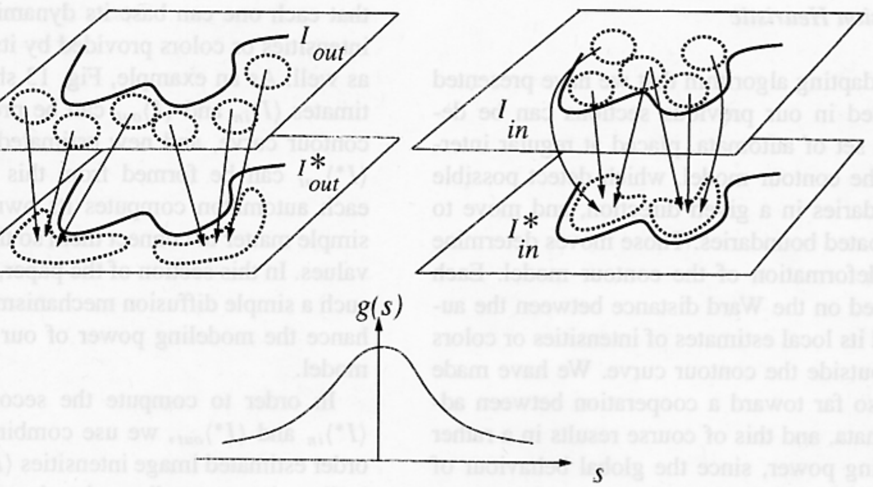

Fig. 13. Extending image neighborhood width. The simultaneous diffusion of intensity on both sides of the contour extends neighborhoods in width.

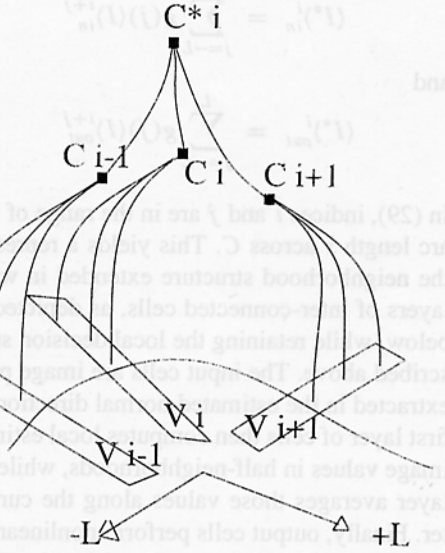

Fig. 14. Interconnection of neighborhood structures. Local color values are weighted sums of the estimated colors for each neighborhood.

based on the following experimental observations:

- substituting $\langle I^* \rangle_{in}$ and $\langle I^* \rangle_{out}$ to the original values in equations (23) and (25) improves the convergence qualities of the anticipating snakes algorithm around object boundaries (where first-encountered boundaries act as seeds for width propagation of the deformations),
- away from object boundaries, extending the width of image neighborhoods yields confusions and unpredictable results, mainly because the differences between inside and outside neighborhoods are small and obscured by random, tangent heterogeneities.

We thus have to introduce a more adaptive propagation scheme, so that we can use the seed values at object boundaries to extend in the width dimension, while ignoring this dimension altogether when no such information is available. Such adaptive treatments are reminiscent of biologically motivated computational models, such as retinex theory (Land 1977) or boundary contour–feature contour interactions (Grossberg 1987).

The retinex scheme computes perceived lightness or color using path integrals away from boundaries—a scheme later shown to be equivalent to a diffusion equation with boundary conditions (Blake & Brelstaff 1987). This original idea was recently re-discovered and extended as anisotropic diffusion i.e. diffusion of image intensity values, with coefficients depending on local image gradients—a scheme which allows filling-in to take place away from object boundaries, which act as diffusion barriers. It should be noted that our own extending neighborhood structures must follow the same kind of rule, but in a different perspective, since all unidentified neighborhoods should progressively be filled-in by image values diffusing from distinct object and background seed half-neighborhoods (the contour curve itself acting as a diffusion barrier).

Interaction between extending neighborhood structures (allowing image intensities to fill-in) and optimizing contour curves (which extend and maintain optimal local contrasts) can best be described in the light of the neural theory of interacting features and boundaries, as defined in Grossberg (1987). S. Grossberg's model of visual perception introduces two competing forces (completion of boundaries and featural filling-in) which are both necessary and sufficient to explain, in his view, most pre-attentive object perception data. Feature contours are generated by orientation-insensitive, direction-of-contrast sensitive cells, while boundary contours are detected by direction-of-contrast insensitive, orientation-sensitive cells.

This is an important distinction, which we are going to use a lot in this section. The process of computing a Ward distance along the normal to the contour curve, as in our previous section, is orientation-sensitive, but not sensitive to direction-of-contrast, i.e. it responses to all discontinuities tangent to the contour curve, regardless of the absolute intensities or colors inside and outside the contour. It would be useful to devise direction-of-contrast sensitive detectors as well, in order to guide the deformable contour towards a continuous boundary, i.e. with a constant or slowly varying direction-of-contrast. Of course, this is difficult, unless we are given the object and background colors, or we can anticipate those colors, based on larger, extended neighborhoods.

In order to achieve this goal in the framework of our anticipating snake algorithm, we propose to use an intensity or color diffusion scheme with non-constant coefficients, each coefficient $h(s)$ at arc-length s being a monotonous increasing function of the image contrast along the normal $N(s)$ at s. This is a quite different assumption compared to anisotropic diffusion, because the diffusion along our curve parameter s is controlled by the contrast in the orthogonal direction t. Therefore, the more contrast we find between the inside and outside colors at s, the more we diffuse those colors. When the diffused colors are used by each automaton to compute its motion, we get a process which has become sensitive in the direction-of-contrast. For highly contrasted points, this does not change the computation of energy and forces much, and those points will still maximize their own local contrast, regardless of direction-of-contrast. On the other hand, for lower contrast points, local color values are outweighted by the diffused values from highly contrasted points, which impose a given direction of contrast. Therefore those points will maximize their Ward distance to the diffused colors "from within" and "from without," as if we had set the functions object() and background() in equation (1) for those points, or equivalently the intensity or color charateristics of regions R_{in} and R_{out} in equation (19). This clearly results in a direction-of-contrast process for those points.

Drawing upon this basic idea, we therefore have implemented a simple diffusion process with a diffusion coefficient $h(s)$ which is a function of arc length. This consists of the same two-layer structure as Fig. 14, but with an additional term $h(s)$. We defer the precise definition of $h(s)$ to section 4.5 and simply introduce it here with the following diffusion equation:

$$\langle I^*\rangle^i = \langle I\rangle^i + \sum_{j=-L}^{L} g(j)h_{i+j}(\langle I\rangle^{i+j} - \langle I\rangle^i) \tag{30}$$

The interpretation of $h(s)$ at $s = i + j$ in this equation is that of a local contrast measure at point $M(s)$. Using $h(s)$ results in a selective diffusion from highly contrasted areas to undifferentiated regions, as in Fig. 15. More precisely, equation (30) results from the following line of reasoning: $\langle I^*\rangle^i$ is expected to take values ranging between the local depth value $\langle I\rangle^i$ and the sûrrounding value I^i_{surr}:

$$I^i_{surr} = \frac{\sum_{j=-L}^{L} g(j)h({}_{i+j})\langle I\rangle^{i+j}}{\sum_{j=-L}^{L} g_j h_{i+j}} \tag{31}$$

Therefore we have $\langle I^*\rangle^i = (1-H)\langle I\rangle^i + HI^i_{surr}$. This simplifies to equation (30) if H is chosen to be the averaged contrast $\sum_{j=-L}^{L} g({}_j)h({}_{i+j})$ around M_i. Substituting $\langle I^*\rangle^i$ from equation (30) into the anticipating snake algorithm, and increasing the neighborhood width L at every iteration, a very efficient and stable scheme was obtained, as illustrated in Fig. 16 and Fig. 17. The critical—and somehow more technical—point in this version of the algorithm lies in the estimation of the $h(s)$ coefficients—which play a similar role here as the line-process energies in Geman et al. (1990).

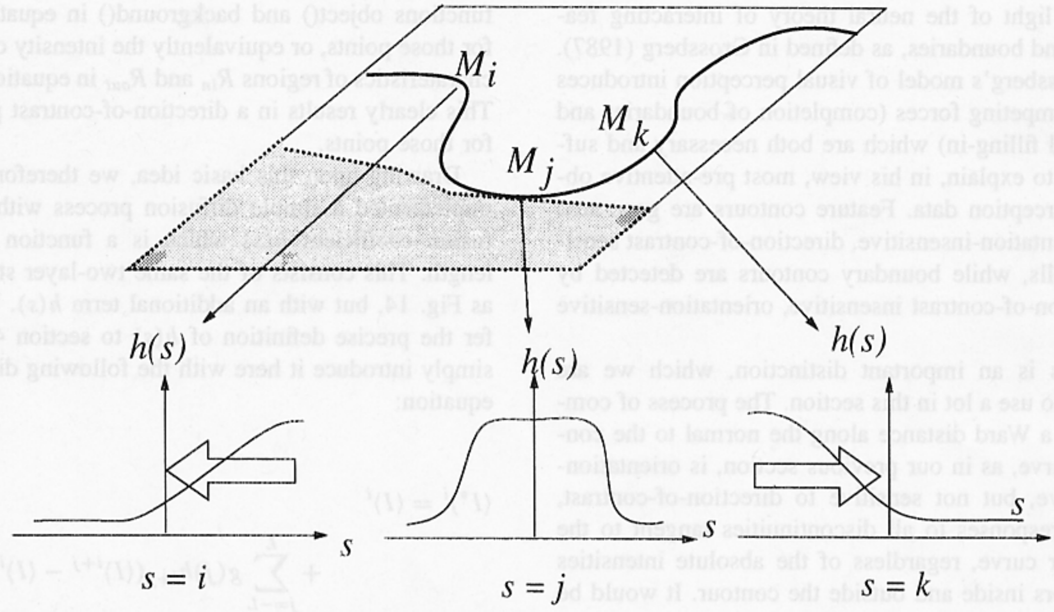

Fig. 15. Diffusion coefficient influence. Diffusion coefficients are related to image contrast across the contour curve. This has the effect of diffusing contrasted colors into undifferentiated regions.

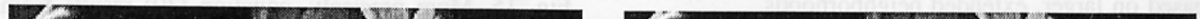

Fig. 16. Application of diffusion algorithm to real-scene image.

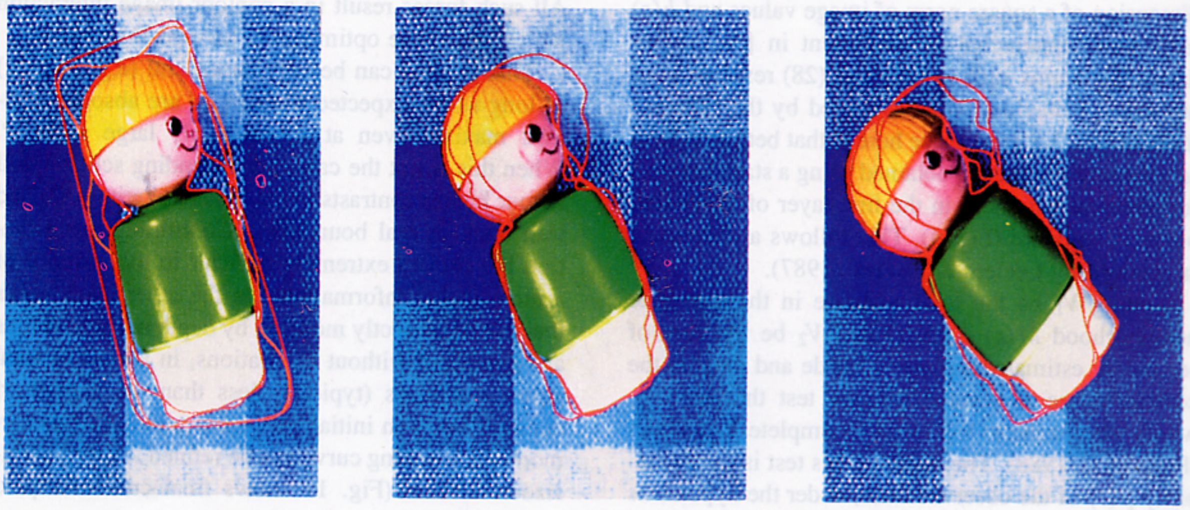

Fig. 17. Steps in diffusion algorithm and object tracking application.

4.4 Adaptive Diffusion Algorithm

In order to illustrate the use of our diffusion heuristic, we have implemented the following extension of our anticipating snakes algorithm.

[1] Starting from an initial contour curve, we perform a first series of iterations following the depth procedure of our previous section, until a limit depth value P is obtained.

[2] We use variances from all half-neighborhoods to estimate the contrasts $h(s)$ and build the two layers of neighborhoods described above.

[3] We compute fusion forces as in equation (15), using the second layer outputs $\langle I^* \rangle^i$ for all neighborhoods, and apply the depth deformation rules from $M(s)$ to $M(s) + t(s)N(s)$.

[4] We increment L and iterate [2] and [3], until an equilibrium is reached.

It should be noted that, since expression (30) acts as a smoothing, regularizing term, no explicit stopping criterion is needed (contrary to the depth-adapting algorithm). In fact, it is possible to rely (as in the original snakes approach) on a much simpler rule, i.e. we end the process when no more deformations occur. Figure 16 presents an example of initial and stabilized contour curves in a color image (shown here in grey levels). The Ward distance is derived from a visual metric in the vector-space of color triplets. All objects have been optimized separately. The contours shown are sampled every five pixels. The depth parameter was varied from 10 to 20 pixels in the first ten iterations. Then the width parameter was increased from 5 to 15 points in the last ten iterations (i.e. every sampled point received information from up to 30 neighbors). The displayed results use as many as 80 to 100 sampling points here. It should be noted that the final scale is large enough for different objects to overlap, meaning that we really obtain a solution which is an optimum among all possible present object boundaries. Figure 17 shows further interesting application results, in the case of a sequence of images. The images in the sequence have been chosen so that the average deplacement corresponds to the depth parameter used, of approximately 20 pixels. Again, this shows a good modeling power, discriminating easily between object boundaries and the surrounding texture. Intermediate states of the contour model are shown in those three images, illustrating only the adaptive-diffusion part of the algorithm (Fig. 17). Except for the first frame, where we have provided an initial curve manually, contours obtained in a given frame were used directly to initialize the next frame (although no real-time implementation has been attempted).

4.5 Computing the Diffusion Coefficients

We cannot use the Ward distance in order to compute the diffusion coefficients, because $D(I_1, I_2)$ has the dimension of a square norm of image values and $h(s)$ must be a dimensionless coefficient in [0,1]. Nonlinear transforms such as (27) and (28) result in poor numerical behaviour when averaged by the gaussian function g_j. Our experience here is that better contrast estimates can in fact be obtained using a statistical test for identical variances in the first layer of the neighborhood structure (Fig. 4). This follows a theoretical suggestion by Leclerc & Zucker (1987).

Letting V_1 be the total variance in the complete neighborhood $M(s) + tN(s)$ and V_2 be the sum of separately estimated variances inside and outside the contour curve at point $M(s)$, we test the ratio $\frac{V_1}{V_2}$, which we assume to follow an incomplete β-function (Press et al. 1988). The result of this test is the probability $p(s)$ of the observed ratio, under the hypothesis that V_1 and V_2 were drawn from a homogeneous population. The local contrast $h(s)$ can thus be estimated as $1-p(s)$. The test Fisher-Snedecor has already been used to study the local structure of discontinuities and detect edges (Leclerc & Zucker 1987) and it proves both consistent and reasonably efficient in our case as well. It extends naturally to more complex image cases than illustrated here, e.g. color or multi-spectral imagery (Ronfard 1991).

5 Results and Discussion

5.1 Scale Changes: Comparison of Depth-Adapting and Diffusion Algorithms

A remarkable feature of the two algorithms presented here is their capacity to change scales during optimization. Our treatment of scale changes in depth closely follows Leclerc & Zucker (1987) but in a more favourable context, because of error-correcting iterations and regularization. As a consequence, a more ambitious treatment of scale changes was possible, as captured by our width extending scheme. However, extensions in depth or in width both have merits and failings, which will now be illustrated and discussed.

Extending in depth has been illustrated in Fig. 1 and Fig. 8. The intuitive nature of the extension process is that all pixels in a neighborhood of the contour curve interact with their projections on the curve, the forces being repulsive, proportional to the similarity of pixel values on and around the curve, and limited in range to the depth parameter P (Fig. 8) All such forces result in a contour-image interaction which drives the optimization process throughout.

This scheme can be very successful (e.g. Fig. 11) as long as the expected boundaries are absolute contrast maxima even at a reasonably large scale P. When this is not the case, our extending scheme will favour higher contrasts in the distance (within P) and lose track of real boundaries. Controlling this situation has proved extremely difficult in the absence of a more global information. On the other hand, cases that can be correctly modeled by depth neighborhoods are optimized without oscillations, in a near-optimal number of steps (typically less than the Haussdorf distance between initial curve and solution). Furthermore, the resulting curve can be refined to quasi pixel-size resolution (Fig. 11 shows final curve sampled every 3 pixels).

Thus, our depth-adapting strategy is best used when image contrast is high and varies smoothly along the anticipated boundary. Otherwise, smoothing image values around optimizing contour is necessary. Adaptive diffusion provides important clues on how to solve those cases. The process has been represented in Fig. 13 as distributed forces acting on the contour curve from neighborhood pixels on the curve.

The diffusion process that we use to compute local pixel colors presents major advantages over both traditional snake methods and our local depth-adapting algorithm, when an overall direction of contrast is perceptible, because it is able to correct itself locally, using neighborhood information without assumptions on the geometry of expected object boundary (as in Fig. 16 and Fig. 17). On the other hand, it is more difficult to control, and not as precise as the simpler depth-adapting algorithm, because it changes image values and therefore leads to smoother contour solutions than the real image boundaries. An implementation using more elaborate neighborhood structures (such as a triangulation of the image) would be very useful for further study of the adaptive diffusion concept.

5.2 Merits and Failings of the Anticipating Snakes Method

The framework presented in this paper addresses only the case of feature contours (step-edges), and cannot easily be made to recognize "roof-edges" or boundaries between objects of similar colors. In the case of step-edges, our diffusion algorithm retains most of the available image contrast information, while allowing non-stationnarity (as opposed to pure region segmentation based on the same energy function). The use of an energy measure derived from local comparisons of quadratic error-of-fit functions was borrowed from split-and-merge methods, for which it is well known that smooth, regular contours are usually difficult to obtain. Our anticipating snakes approach could provide a criterion for those cases, making contour extraction much more reliable, while retaining the robustness qualities of region analysis. It also supports scale changes more easily than other related schemes, which is an important factor for stability of our results.

Figures 11 and 12 present results obtained with the depth algorithm, on tomography scan and microscopy images. Figure 16 shows results of the diffusion algorithm on a fairly complex video image. An application of the diffusion algorithm to object tracking in an image sequence is shown in (Fig. 17). In simpler cases such as Fig. 11 or Fig. 12, convergence is obtained in ten to fifteen iterations, with an initial depth of about five pixels, and the algorithm runs in time linear with the number of sampled points. If the diffusion heuristic is used in those images, the number of iterations is reduced, but the running time increases because of the more drastic computations there. However, all examples shown were obtained in times compatible with interactive use (typically under 2 seconds on a 12 MHz microcomputer for 30 sampled points).

The combination of contour-oriented control structures and region-oriented energy measures is powerful, yet many problems will remain unsolved as long as contours are optimized one at a time. There are cases when our anticipating snakes split into separate curves, and it would be interesting to proceed with such twin processes (as a matter of fact, such situations are detected, and corrected by simply deleting all loops but the largest one). Multiple-contour optimization could also prove valuable in order to generate initial curves at different positions in the image—sharing neighborhood values when necessary, thus cooperating into a global segmentation procedure such as Geman et al. (1990), but with an explicit contour shape representation, as in Mumford & Shaw (1985; 1989).

Other difficult issues in the approach that we present here include the choice of an optimal sampling rate, efficient control of the diffusion heuristic after several iterations, and the trade-off between local depth-adapting and diffusion strategies in the course of optimization. Such issues should be discussed in the light of more specific, domain-dependent image analysis application of the method.

6 Conclusion

This paper was motivated by the need to explore variational conditions and algorithms resting on local region analysis, instead of pre-processed edge maps, to handle cases when such maps are not available or too costly. Based on such local region criteria, we have presented a coherent treatment of image-contour forces and variations, as well as strategies for application in a practical active contour system, which compares well with other existing systems, especially in the case of over-segmented images. This is remarkable since we essentially took a very simplistic approach to optimization, control and convergence issues. We argue that it is due to the fact that the region-based energy used here is a better representation for optimal shapes than are image gradients in those cases. Our energy definitions have been presented with statistical and perceptual interpretations, and can be suggested—along with their variational procedures and heuristics—to enforce explicit shape representations in other areas of image analysis i.e. simulated annealing image segmentation and retinex/diffusion schemes.

In the more specific domain of active contour models, our first contribution has been to introduce a general formulation for region-based models, using local error-of-fit functions to build an energy criterium suitable for optimization. This formulation has been illustrated using piece-wise constant image models only, and should benefit from higher-order models when available. Our second contribution has been to make use of adaptive neighborhood structures and diffusion processes, in order to obtain a more robust active contour modeling scheme in the case of very busy images with many objects. The same heuristics could easily be transposed to the case of edge-based contour models as well, hopefully with the same benefits.

This work was performed while the author was a research assistant with Ecole des Mines de Paris, as well as being hosted by Laboratoire Image, Telecom Paris. The author would like to acknowledge the help and support of J.M. Monget, M. Albuisson and

L. Wald, at Ecole des Mines, H. Maitre, F. Schmidt and M. Sigelle at Telecom, as well as valuable critical examination and discussion on earlier versions of this work by Ph. Cinquin.

References

A. A. Amini, T. E. Weymouth & R. C. Jain, Using dynamic programming for solving variational problems in vision, IEEE trans. Pattern Analysis &Machine Intelligence, vol. 12, no. 9, 1990.

J. M. Beaulieu & M. Goldberg, Hierarchy in picture image segmentation—a step-wise optimization approach, IEEE trans. Pattern Analysis & Machine Intelligence, vol. 11, no. 2, pp. 150–163, 1989.

M. O. Berger, Snake growing, Lecture Notes in Computer Sciences, vol. 427, edited by O. Faugeras, pp. 571–572, Springer-Verlag, 1990.

A. Blake & G. Brelstaff, Computing lightness, Pattern,Recognition Letters, vol. 5, pp. 129–138, 1987.

P. Cinquin, F. Leitner, I. Marque & S. Lavallée, 2D and 3D segmentation methods based on differential equations and spline-snakes, Proc. Conference on Curves & Surfaces, Chamonix, France, 1990.

R. Cipolla & A. Blake, The dynamic analysis of apparent contours, 3rd International Conference on Computer Vision, Osaka, 1990.

Cohen, On active contour models & balloons, Computer Vision, Graphics & Image Processing, Image Understanding, vol. 53, n. 2, pp. 211–218, March 1991.

P. Fua & Y. Leclerc, Model driven edge detection, Machine Vision & Applications, vol. 3, pp. 45–56, 1990.

M. Gage, On an area-preserving evolution equation for plane curves, Contemporary Mathematics, vol. 51, pp. 51–62, 1986.

S. Geman & D. Geman, Stochastic relaxation, Gibbs distributions and the restoration of images, IEEE trans. Pattern Analysis & Machine Intelligence, vol. 6, no. 6, 1984.

S. Geman, D. Geman, C. Graffigne & P. Dong : Boundary detection by constrained optimization, IEEE trans. Pattern Analysis & Machine Intelligence, vol. 12, no. 7, pp. 609–627, 1990.

S. Grossberg, Cortical dynamics of 3-dimensional form, color and brightness perception—monocular theory, Perception & Psychophysics, vol. 41, no. 2, pp. 87–116, 1987.

R. M. Haralick, Digital step-edges from zero-crossings of second directional derivatives, IEEE trans. Pattern Analysis & Machine Intelligence, vol. 1, pp. 58–68, 1984.

M. Kass, A. Witkins & D. Terzopoulos, Snakes—active contour models, International Journal of Computer Vision, vol. 1, no. 4, pp. 321–330, 1987.

B. B. Kimia, A. Tannenbaum & S. W. Zucker, Toward a computational theory of shape : an overview, Lecture Notes in Computer Sciences, vol. 427, pp. 402–407, Springer-Verlag, 1990.

J. J. Koenderink & A. J. van Doorn, The Structure of Two-dimensional Scalar Fields with Applications to Vision, Biological Cybernetics, vol. 33, pp. 151–158, 1979.

E. H. Land, The retinex theory of color vision, Scientific American, vol. 237, no. 6, pp. 108–129, 1977.

P. J. Laurent, Approximation et optimisation, Hermann, Paris, 1972.

Y. G. Leclerc & S. W. Zucker, The local structure of image discontinuities in one dimension, IEEE trans. Pattern Analysis & Machine Intelligence, vol. 9, no. 3, pp. 341–355, 1987.

F. Leitner, F. Berthommier, T. Coll, I. Marque, D. Francillard, Ph. Cinquin & J. Demongeot, Neural networks, differential systems and segmentation of medical images, in From pixels to features II, ed. by H. Burkhardt, Y. Neuvo & J. C. Simon, Elsevier Science Publishers, pp. 253–274, 1991.

Marroquin, Mitter & Poggio, Probabilistic solution of ill-posed problems in computional vision. 1987

S. Menet, P. Saint-Marc & G. Medioni, Active contour models: Overview, Implementation and applications, proc. IEEE Conference on Systems, Man & Cybernetics, Los Angeles, California, USA, 1990.

D. Mumford & J. Shah, Boundary detection by minimizing functionals, in Proc. International Conference on Computer Vision and Pattern Recognition, pp. 22–26, San Franciso, 1985.

D. Mumford & J. Shah, Optimal approximations by piece-wise smooth functions and associated variational problems, Communications on Pure and Applied Mathematics, vol. 42, pp. 577–685, 1989.

M. Nitzberg & D. Mumford, The 3.1-D sketch, 3rd International Conference on Computer Vision, Osaka, 1990.

P. M. Prenter, Splines and Variational Methods, Wiley Classics Library, 1975, reprinted in 1989.

W. H. Press, B. P. Flannery, S. A. Teukolsky & W. T. Vetterling, Numerical Recipes in C, Cambridge University Press, 1988.

R. Ronfard, Contours & boundaries in color images, proc. of SPIE Conference on Visual Communication & Image Processing, Lausanne, 1990.

R. Ronfard, Principles for variational edge detection in multi-spectral and color images, PhD thesis, Ecole des Mines de Paris, Feb. 1991.

J. Shah, Parameter estimation, multi-scale representation and algorithms for energy-minimizing segmentations, in Proc. 10th International Conference on Pattern Recognition, pp. 815–819, Atlantic City, 1990.

B. Sharahray & D. J. Anderson, Optimal estimation of contour properties by cross-validated regularization, IEEE trans. on Pattern Analysis & Machine Intelligence, vol. 11, n. 8, Jan. 1989.

M. Sigelle & R. Ronfard, "Relaxation of previously classified images by a Markov Field technique and its relationship with statistical physics," proc. of the 7th Scandinavian Conference on Image Analysis, IAPR, Aalborg, Denmark, August 1991.

R. Szeliski & D. Terzopoulos, From splines to fractals, Computer Graphics, vol. 23, no. 3, pp. 51–60, 1989.

Appendix A Variational Principles and Equations

We assume that the exact energy expression is :

$$E = \int_0^1 \left(\alpha || \frac{\partial M}{\partial s} ||^2 + \beta || \frac{\partial^2 M}{\partial s^2} ||^2 - G^2(M) \right) ds$$

and that it can be approximated by a sum of local energy fields $E = \sum_{i=1}^n E_i$:

$$E = \alpha \sum_{i=1}^n || \frac{\partial M_i}{\partial s} ||^2 + \beta \sum_{i=1}^n || \frac{\partial^2 M_i}{\partial s^2} ||^2 - \sum_{i=1}^n G^2(M_i)$$

All terms in this expression can be approximated by finite differences:

$$
\begin{aligned}
\|\frac{\partial M_i}{\partial s}\|^2 &= (x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 \\
\|\frac{\partial^2 M_i}{\partial s^2}\|^2 &= (x_{i+1} - 2x_i + x_{i-1})^2 \\
&\quad + (y_{i+1} - 2y_i + y_{i-1})^2
\end{aligned}
$$

If we now differentiate, we obtain

$$\frac{\partial E}{\partial x_i} = \sum_{j=1}^{n} B_{ij} x_j - \frac{\partial G^2}{\partial x_i}$$

and

$$\frac{\partial E}{\partial y_i} = \sum_{j=1}^{n} B_{ij} y_j - \frac{\partial G^2}{\partial y_i}$$

in which the coefficients in B are simple sums of α and β.

If we assume $E(M)$ to be a quadratic function, we can write (using Euler formula for homogeneous functions):

$$2E = \sum_{i=1}^{n} \frac{\partial E}{\partial x_i} x_i + \sum_{i=1}^{n} \frac{\partial E}{\partial y_i} y_i$$

which easily develops into

$$E = \frac{1}{2}\left[{}^t XBX -{}^t X\frac{\partial G^2}{\partial X} +{}^t YBY -{}^t Y\frac{\partial G^2}{\partial Y}\right]$$

Equation (7) in the paper is but a short-hand notation for this latter expression, in which we recognize potential energy terms ${}^t XBX +{}^t YBY$ as well as artificial, quasi-static Gibbs function coefficients ${}^t X\frac{\partial G^2}{\partial X} +{}^t Y\frac{\partial G^2}{\partial Y}$.

When we use our region-based formalism, the latter, image-driven energy terms appear only as variations ${}^t\delta X\frac{\partial G^2}{\partial X} +{}^t\delta Y\frac{\partial G^2}{\partial Y}$ which are interpreted (more meaningfully in our opinion) as the amount of dissipated energy due to the image force vector ∇G^2.