To be published in the proceedings of the
IEEE International Conference on
Computer Vision and Pattern Recognition (CVPR'96),
June 16-21, 1996, San Francisco, California, USA

Global Minimum for Active Contour Models: A Minimal Path Approach

Laurent D. COHEN
CEREMADE, URA CNRS 749
Université Paris 9-Dauphine
75775 Paris cedex 16, France
cohen@ceremade.dauphine.fr

Ron KIMMEL
Lawrence Berkeley Laboratory
University of California, Berkeley, USA
Mailstop 50A-2152, LBL UC Berkeley CA 94720
ron@csr.lbl.gov

Abstract

A new boundary detection approach for shape modeling is presented. It detects the global minimum of an active contour model's energy between two points. Initialization is made easier and the curve cannot be trapped at a local minimum by spurious edges. We modify the "snake" energy by including the internal regularization term in the external potential term. Our method is based on the interpretation of the snake as a path of minimal length in a Riemannian metric, or as a path of minimal cost. We then make use of a new efficient numerical method to find the shortest path which is the global minimum of the energy among all paths joining the two end points. The method is extended to closed contours, given only one point on the objects' boundary by using a topology–based saddle search routine. We show examples of our method applied to real aerial and medical images.
Keywords: Shape modeling, Deformable Models–Snakes, Path of minimal cost, Level Sets, Segmentation, Feature Extraction, Energy Minimization, Partial Differential Equations.

1 Introduction

An active contour model for boundary integration and features extraction, introduced in [1], has been considerably used and studied during the last decade. Most of the approaches that were introduced since then try to overcome the main drawbacks of this model: *initialization, minimization and topology changes.*

The model requires the user to input an initial curve close to the goal. Often, it has to be a very precise polygon approximation and it may be fastidious to use for an application dealing with a large number of images. Using the balloon model [2] allows a less demanding initialization since any initial closed curve inside an object may be used to obtain its complete boundary. The same property can be realized using the geometric model introduced in [3, 4] and recently improved in [5]. In [6], only two end points on the boundary are needed to follow the contour.

In this paper we present a new approach for finding the global minimum of energy minimizing curves given only one or two end points. Our goal is to help the user in solving the problem in hand by mapping it into a single minimum problem. The proposed method contributes to the improvement of the first two items above, *initialization* and *minimization* which are obviously related. Only end points are needed as an easy initialization and we are guaranteed that the global minimum is found between these points.

We modify the snake energy in a way that makes it 'intrinsic' or free of the parameterization. Most of the classical snake models are non-intrinsic models. Therefore, different parameterizations of the same (i.e. having the same geometric shape) initial curve, could lead to different solutions.

The modification we follow enables us to include the internal regularization term in the external potential term in a natural way, since the snake energy depends only on the location of the point and not on the geometry of the curve at this point. We use an evolution scheme that provides at each image pixel an output of the energy along the path of minimal integrated energy joining that pixel to the given start point. We use the Sethian fast marching method [7, 8, 9]. The search for a global minimum is then done efficiently. While this minimum is restricted to connect two given points, we also present a topology–based saddle search that helps in automatically closing contours by clicking on a single point along the boundary.

An upper bound of the curvature along the minimal path is introduced. This justify the fact that although our approach is a path integration, it also incorporate the regularization of the path like a "snake" model.

2 Deformable Contours

We refer to [10] for an overview of the different approaches for active contour models.

Since the introduction of "snakes" [1], deformable models have been extensively used to integrate boundaries and extract features from images. The deformable contour model is a mapping $\mathcal{C}(v) = (x(v), y(v))$. In some cases v is chosen to be the arc-length parameter, ranging on $\Omega = [0, L]$ where L is the length of the curve[1]. The energy of the model has

[1] We shall refer to arc-length parameter as s, to differ from an arbitrary parameter v

the following form:

$$E(\mathcal{C}) = \int_\Omega \frac{w_1}{2}\|\mathcal{C}_v(v)\|^2 + \frac{w_2}{2}\|\mathcal{C}_{vv}(v)\|^2 + P(\mathcal{C}(v))dv,$$

where $\mathcal{C}_v$ and $\mathcal{C}_{vv}$ are the first and second derivatives of $\mathcal{C}$ with respect to v, and P is the potential associated to the external forces. For the problem to be well-posed, the space of admissible deformations $\mathcal{A}$ is restricted by boundary conditions. These may be free boundaries, as in the original snakes [1], cyclic boundaries by using periodic closed curves [11], or fixed boundaries by giving $\mathcal{C}(0)$, $\mathcal{C}_v(0)$, $\mathcal{C}(1)$ and $\mathcal{C}_v(1)$ [2, 12].

A geometric approach for deformable models was recently introduced in [3, 4]. A level set approach for curve evolution [13, 14] is used to implement a planar curve evolution of the form:

$$\frac{\partial \mathcal{C}(s,\tau)}{\partial \tau} = P(\mathcal{C})(\mathcal{C}_{ss} + w\ \vec{n}), \qquad (1)$$

where s is the arc-length parameter of the curve $\mathcal{C}$ in this case. Therefore, $\mathcal{C}_{ss} \equiv \kappa\vec{n}$ is the curvature vector ($\vec{n}$ being the unit normal), and w is some predefined constant. This constant term is thus similar to the pressure force introduced for the balloon model [2].

It was recently proven that introducing the 'gradient of potential' (∇P) term of the classical energy minimization snakes [1, 12] into the geometric snakes [3, 4] is based on geometrical as well as energy minimization reasoning, leading to the "geodesic active contour" [5].

These were shown to 'behave' better than both its 'ancestors' since they enjoy the advantages of both. It is shown in [5] that their curve evolution is a result of minimizing the functional $E(\mathcal{C}) = \int_\Omega P(\mathcal{C}(s))ds$, where s is the arclength (or $E(\mathcal{C}) = \int_\Omega P(\mathcal{C}(v))\|\mathcal{C}_v\|dv$, for the arbitrary parameter v).

Although our work is related to [5], it is a totally independent approach. We note that following the formulation of [5], the minimization of the classical energy (1) may be modified into the problem of finding *local geodesics* in a Riemannian metric computed from the image, where we propose to find the *minimal geodesics* in a similar Riemannian metric (see Equation (2) in the following section). Although it is shown in [5] that finding the solution of active contour models is closely related to finding geodesics, no method is proposed to find the minimal ones. In [5], like in most of the previous approaches, the algorithms search for some local minimum that is close to the initial guess, while we propose a method for finding the global minimum of the same energy between two points as a minimal path (minimal geodesic).

3 Paths of Minimal Action

Given some potential P that takes lower values near the edges or features, our goal is to find a single contour that best fits the boundary of a given object or a line of interest. This 'best fit' question leads to algorithms seeking for the minimal path, i.e. paths along which the integration over P is minimal. As mentioned earlier, snakes start from a path close to the solution and converge to a local minimum of the energy. Given only the end points, our goal is to find the minimal path between these points, thereby simplifying the initialization process and avoiding erroneous local minima. Motivated by the ideas put forward in [15, 16] we develop an efficient and consistent method to find the path of minimal cost between two points, using the surface of minimal action [16] and the fact that operating on a given potential (cost) function helps in finding the solution for our path of minimal action (also known as minimal geodesic, or path of minimal potential). Thereby, we are able to isolate the boundary of a given object in the image.

3.1 Problem Formulation

The minimization problem we are trying to solve is slightly different from the deformable models, though there is much in common. One may still differ between "internal" and "external" forces, yet now all terms are geometric which means a result of an intrinsic energy functional. Contrary to the classical snake energy, here s represents the arc-length parameter, i.e. $\|\mathcal{C}_s(s)\| = 1$. The reason we modified the energy is that we now have an expression in which the internal regularization energy is included in the potential term in a natural way. We can then solve the energy minimization in a similar way to that of finding the shortest path on a surface [15]. The fact that the energy integral is now intrinsic will also help us to explore the relation between the smoothness of the result and the potential. The energy $E(\mathcal{C})$ of the new model has the following form:

$$\int_\Omega w\|\mathcal{C}_s\|^2 + P(\mathcal{C})ds = wL + \int_\Omega P(\mathcal{C})ds = \int_\Omega \tilde{P}(\mathcal{C})ds \qquad (2)$$

where $\tilde{P}(p) = w + P(p)$ and the energy is minimized on $\mathcal{A}_{p_0,p_1}$, the space of all curves connecting two given points (restriction by boundary conditions): $\mathcal{C}(0) = p_0$ and $\mathcal{C}(L) = p_1$, where L is the length of the curve. Contrary to the classical snake energy, here s represents the arc-length parameter. This makes the energy depend only on the geometric curve and not on the parameterization. The regularization term multiplied by the constant w, now exactly measures the length of the curve. We note that a similar regularization effect may be also achieved by smoothing the potential P [17].

Having the above minimization problem in mind, we first search for the *surface of minimal action* U_0 starting at $p_0 = \mathcal{C}(0)$. At each point p of the image plane, the value of this surface U_0 corresponds to the minimal energy integrated along a path starting at p_0 and ending at p.

$$U_0(p) = \inf_{\mathcal{C}(L)=p}\left\{\int_{\mathcal{C}} \tilde{P}ds\right\} = \inf_{\mathcal{A}_{p_0,p}} E(\mathcal{C}), \qquad (3)$$

We next present an approach to determine the value of U_0 everywhere in the image domain.

3.2 Shortest Paths As a Set

Following [15], given the minimal action surfaces U_0 to p_0 and U_1 to p_1, then the minimal geodesic between p_0 and p_1 is exactly the set of coordinate points p_g that satisfy

$$U_0(p_g) + U_1(p_g) = \inf_{p \in \mathbb{R}^2} \{U_0(p) + U_1(p)\}. \quad (4)$$

Usually, the set of points p_g needs to be refined from a given "fat" set of points into a curve. In [16] a thinning algorithm was applied. The above is a global way for extracting the global minimum. In our experiments we have preferred to use a back propagation procedure that results in a single curve (see Section 3.5.2.)

3.3 Minimal Action Level Sets Evolution

In what follows, we assume that $P \geq 0$. Applying the ideas of the previous section to minimize our energy (2), it is possible to formulate a partial differential evolution equation describing the set of equal energy contours $\mathcal{L}$ in 'time', where t is in fact the value of the energy. These are the level sets of U_0 defined by Equation (3). In the evolution equation t represents the height of the level set of U_0:

$$\frac{\partial \mathcal{L}(v,t)}{\partial t} = \frac{1}{\tilde{P}} \vec{n}(v,t), \quad (5)$$

where $\tilde{P} = P + w$ and $\vec{n}(v,t)$ is the normal to the closed curve $\mathcal{L}(.,t) : S^1 \rightarrow \mathbb{R}^2$. The motivation for this evolution is that we need to propagate with a velocity that is proportional to the inverse of the penalty. So that at 'low cost' area the velocity is high while at a 'high cost' area the velocity is low.

The curve $\mathcal{L}(v,t)$ corresponds here to the set of points p for which the minimal energy $U_0(p)$ is t:

$$\{\mathcal{L}(v,t), v \in S^1\} \; = \; \{p \in \mathbb{R}^2 \mid U_0(p) = t\}. \quad (6)$$

This evolution equation is initialized by a curve $\mathcal{L}(v,0)$ which is an infinitesimal circle around the point p_0. It corresponds to a null energy. It then evolves according to Eq. (5), similar to a balloon evolution [2] with an inflation force depending on the potential. Considering the (x,y,t)-space, the family of curves $\mathcal{L}(v,t)$ construct the level sets of the surface $U(x,y) : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ defined in (3). The t level set of U is exactly the curve $\mathcal{L}(.,t)$. Although a rigorous proof of this statement can be found in [18], it can be understood simply by the following geometric interpretation. Observe that adding to a path ending at a point of $\mathcal{L}(.,t)$ a small segment in the normal direction to $\mathcal{L}(.,t)$ and of length $\frac{1}{\tilde{P}}dt$ will add to the accumulated energy of (2) a contribution of $\tilde{P}\frac{1}{\tilde{P}}dt = dt$. This means that the new point is on the level $t + dt$, that is on the curve $\mathcal{L}(.,t+dt)$. Figure 4 presents such a surface U and its corresponding level sets.

It is possible to compute the surface U in several ways. We shall describe three of them that are consistent with the continuous case while implemented on a rectangular grid.

3.4 Numerical Implementation

The numerical schemes we propose are consistent with the continuous propagation rule. The consistency condition guarantees that the solution converges to the true one as the grid is refined. This is known **not** to be the case in general graph search algorithms that suffer from *digitization bias* due to the *metrication error* when implemented on a grid. This gives a clear advantage to our method over minimal path estimation using graph search. To evaluate and minimize the snake energy (1), the "internal" terms can be evaluated only by using the shape of the curve, leading to curve deformation and evolution schemes from an initial curve. Based on the new energy definition (2), we are able to compute the final path without evolving an initial contour, by using the surface of minimal action. To find the surface of minimal action, graph search and dynamic programming techniques were often used, considering the image pixels as vertices in a graph [17]. We review and compare these algorithms to our approach in [10]. These algorithms are indeed efficient, yet suffer from 'metrication errors'. Our philosophy here is different. We propose to deal with the continuous problem as long as possible. In that, we follow the numerical analysis community, by first analyzing the underlying problem in the continuous domain. Then, at the last stage which involves numerical implementation we will consider the image given as a grid of pixels, compute optimal paths and the surface of minimum action in a relatively efficient way, while at the same time enjoy the 'consistency' property of converging to the desired continuous solution as the grid is refined. The main reason is obviously accuracy which is important for example in medical applications.

3.4.1 Front Propagation Approach

According to this first continuous approach, the curve evolution $\mathcal{L}(t)$ of Equation (5) is reformulated into an evolution of an implicit representation of the curve defined by an evolving surface $\phi : \mathbb{R}^2 \times [0,T) \rightarrow \mathbb{R}$, where for each value of t, $\mathcal{L} = \phi^{-1}(0)$. This means that curve $\mathcal{L}(t)$ is the zero level set of $\phi(t) : \mathbb{R}^2 \rightarrow \mathbb{R}$. This Eulerian formulation for curve evolution was introduced by Osher and Sethian in [13, 14] to overcome numerical difficulties and handle topological changes. As initialization for $\mathcal{L}(0)$, we start with an infinitesimal circle around the start point p. The function ϕ is initialized at $t = 0$ to be negative in the interior and positive in the exterior of the curve $\mathcal{L}(0)$. This is obtained by setting one pixel to -1 and the rest to $+1$. The evolution rule of ϕ is then given by:

$$\frac{\partial \phi}{\partial t} = -\frac{1}{\tilde{P}} \|\nabla \phi\|. \quad (7)$$

For a fast implementation, of order $O(M\frac{\int ds}{\Delta t})$ where M is the number of points in a narrow band around the front and Δt is the time step of the scheme, of the above approach we refer to [19].

3.4.2 Shape from Shading Approach

The second approach is based on a shape from shading method introduced by Rouy and Tourin [20, 21] and searches for the surface U itself instead of tracking its level sets. In this case the surface may be found according to the following minimization procedure:

Given $\mathcal{U} = 0$ at the start point as boundary condition,

$$\frac{\partial \mathcal{U}}{\partial \tau} = \tilde{P} - \|\nabla \mathcal{U}\|, \tag{8}$$

where the solution U is the steady state of $\mathcal{U}(p, \tau)$ when τ is large. The limit value $U = \mathcal{U}_\infty$ is such that

$$\|\nabla U\| = \tilde{P}, \tag{9}$$

with obviously $U = 0$ at the start point. We can again give a geometric interpretation that relates (5) to (9). The gradient of U is normal to its level sets $\mathcal{L}(t)$, and the gradient norm is thus the value of the spatial directional derivative in the normal direction. As U increases by dt, the normal displacement of the level set $\mathcal{L}(t)$ is $\frac{dt}{\tilde{P}}$ from (5). So the derivative $\frac{\partial U}{\partial \vec{n}} = \langle \nabla U, \vec{n} \rangle = \|\nabla U\|$ is equal to $dt / \frac{dt}{\tilde{P}} = \tilde{P}$. Here, boundary conditions are given in the form of fixing the point $\mathcal{C}(0) = p_0$, i.e. $\mathcal{U}(p_0, \tau) = 0$ for all τ. Authors of [20] also presented a direct numerical approach to solve (9) and gave a convergence proof to that minimization procedure in the viscosity solutions framework. We shall discuss this method and its discretization in more details in the following section. The method we recommend is presented in the following section. It is in some sense a hybrid of both methods just described.

3.4.3 Sethian Fast Marching Method

In his recent report [7], Sethian presents a fast and efficient method for solving Eq. (9). It is based on a clever way for propagating the information on the grid. Motivated by the two methods above and by the narrow band approach [19], his method uses the proposed numerical scheme in [13, 20]. However, by marching in an ordered way, the problem is solved after a fixed number of steps, and by that contradicting Remark 5 in [20]. We recommend this method for any real time application.

Given the potential values $P_{i,j} = P(i\Delta x, j\Delta y)$ on a grid (e.g. the pixel grid), the numerical method approximating $U_{i,j}$ in Eq. (9) is given by

$$\begin{aligned} &\left(\max\{u - U_{i-1,j}, u - U_{i+1,j}, 0\}\right)^2 + \\ &\quad \left(\max\{u - U_{i,j-1}, u - U_{i,j+1}, 0\}\right)^2 \quad = P_{i,j}^2, \end{aligned} \tag{10}$$

where, for simplicity, we assume $\Delta x = \Delta y = 1$. In [20] the numerical viscosity solution was obtained by solving the above equation at each grid point, selecting for $U_{i,j}$ the largest u that satisfies Eq. (10). The grid points were selected in an arbitrary way, and thus it was claimed that convergence is obtained after infinite number of such iterations. Where each iteration involves an arbitrarily selection of a grid point (i, j), and updating the value of $U_{i,j}$ at that point.

The 'fast marching level set method' introduces order in the selection of the grid points. It is based on the fact that information is propagating form the source point 'outwards'. Following [7], the method goes as follows:

Initialization:

- For each point in the grid, let $U_{i,j} = \infty$ (large positive value). Label all points as *far*.
- Set the start point $(i, j) = p$ to be zero: $U_p = 0$, and label it *trial*.

Marching Forward Loop:

- Let (i_{min}, j_{min}) be the *trial* point with the smallest U value.
- Label the point (i_{min}, j_{min}) as *alive*, and remove it from the *trial* list.
- For each of the 4 neighboring grid points (k, l) of (i_{min}, j_{min}):
  - If (k, l) is labeled *far*, then label it *trial*.
  - If (k, l) is not *alive*, then compute $U_{k,l}$ according to Eq. (10), selecting the largest solution to the quadratic equation, which is the only valid solution. *i.e.* solve

$$\begin{aligned} &\left(\max\{u - \min\{U_{k-1,l}, U_{k+1,l}\}, 0\}\right)^2 + \\ &\quad \left(\max\{u - \min\{U_{k,l-1}, U_{k,l+1}\}, 0\}\right)^2 \quad = P_{k,l}^2, \end{aligned}$$

  and let $U_{k,l} = u$.

For efficiency, the *trial* list is kept as min heap structure. We refer to [7, 8, 9, 22] for further details on the above algorithm, as well as a proof of correct construction. Using a min-heap structure for the *trial* list, the algorithm computational complexity is $O(NlogN)$ where N is the number of grid points. It has similar complexity to that of graph search based algorithms like the A^* or Dijkstra. For example on a SPARC 1000, it took a second to compute the U surface of a 256×256 image. This is a first order numerical scheme. As an example for accuracy we should note that the Euclidean distance ($P_{i,j} = 1$) from a straight line is accurate with sub pixel accuracy (error = 0). In general, the consistency condition guarantees that as the grid is refined, the solution converges to the true continuous one.

3.5 Global Snake Minimization

3.5.1 Shortest path between p_0 and p_1

Using the approach described in Section 3.2, the shortest path between a start point p_0 and a destination point p_1, according to the energy minimization is the set of points $p_m = (x_m, y_m)$ that satisfy:

$$U_0(x_m, y_m) + U_1(x_m, y_m) = \inf_{(x,y)} \{U_0(x, y) + U_1(x, y)\}, \tag{11}$$

where U_0 and U_1 correspond to the minimal action obtained in the previous section with paths starting at p_0 and p_1 respectively. A natural combination is to use the above method in order to locate the minimal set, and then let the model defined in [5] take over and refine the result. However, we recommend an easier way to compute the path by back propagation.

3.5.2 Back propagation from p_1

In order to determine the minimal path between p_0 and p_1, we need only to calculate U_0 and then slide back on the surface U_0 from $(p_1, U_0(p_1))$ to $(p_0, 0)$. The surface of minimal action U_0 has a convex like behavior in the sense that starting from any point $(q, U_0(q))$ on the surface, and following the gradient descent direction, we will always converge to p_0. It means that U_0 has only one local minimum that is of course the global minimum and is reached at p_0 with value zero. We show in Figure 4 an example of 3D representation of the $U_0(x, y)$ surface and a level set image of the same U_0. Given the point p_1, the path of minimal action connecting p_0 (the minimal point in U_0, $U(p_0) = 0$) and p_1 is the curve $\tilde{\mathcal{C}}(\sigma)$ starting at $\tilde{\mathcal{C}}(0) = p_1$ and following the opposite gradient direction on U_0:

$$\frac{\partial \tilde{\mathcal{C}}}{\partial \sigma} = -\nabla U_0, \tag{12}$$

Then the solution $\mathcal{C}(s)$ is obtained by arclength parameterization of $\tilde{\mathcal{C}}(-\sigma)$ with $\mathcal{C}(0) = p_0$ and $\mathcal{C}(L) = p_1$. The minimal path can be obtained this way since ∇U is tangent to the geodesic. This is a consequence of the results in [23] that show that the light rays (geodesics, constant parameter curves) are orthogonal to the wave fronts (equal cost contours). The gradient of U is also orthogonal to the wave fronts since these are its level sets.

The back propagation procedure is a simple steepest gradient descent. It is possible to make a simple implementation on a rectangular grid: given a point $q = (i, j)$, the next point in the chain connecting q to p is selected to be the grid neighbor (k, l) for which $U(k, l)$ is the minimal, and so forth. Of course, a better tracking can be obtained using a more precise estimation of the gradient of U. In our examples we have chosen the discrete steepest descent just described, because of its simplicity, and the fact that it is used only for presentation purpose. We back track the path of minimal action connecting the two points, which is the global minimum of the snake energy defined in Eq. (2). Being a local operation, back propagation suffers from angular error accumulation. In [22], a more sophisticated back propagation technique developed for other purposes is introduced, it is used in the examples of Figure 4.

4 Regularization properties

We showed in [10] how the constant w and the potential P in Eq. (2) control the smoothness of the solution. We introduced quantitative results in the form of geometric bounds on the curvature of the final contour. Given a potential $P > 0$ and let $\tilde{P} = w + P$, the curvature magnitude $|\kappa| = \|\mathcal{C}_{ss}\|$ along the geodesics minimizing (2), $\int_\Omega P(\mathcal{C}(s))ds$, where s is the arclength parameter, or $\int P(\mathcal{C}(v))\|\mathcal{C}_v\|dv$ for an arbitrary parameter v, is bounded by:

$$|\kappa| \leq \frac{\sup_{\mathcal{D}}\{\|\nabla P\|\}}{w}. \tag{13}$$

The conclusion is that increasing the constant w added to P, increases the denominator without affecting $\sup_{\mathcal{D}}\{\|\nabla P\|\}$. This gives a justification for referring to w as a regularization parameter.

5 Closed Boundary Extraction

It is often needed to detect a closed contour. Our previous approach of finding a minimal path between two given end points, detects the two paths that complete a closed contour only if both ways correspond to a global minimum. In the general case of selecting the second point, it is clear that although both ways are local minima, only one is a global minimum. Assuming only one start point p_0 is given on the closed contour, let us compute the minimal action U from this start point. We should then find a second point p_1 that is located on the unknown contour and from where the two half paths have the same energy. This means we have to find a point p_1 from which there is more than one curve connecting it to the source p_0. As can be justified in [10], these special points are the saddles of U.

The saddle points may serve as clues in closing contours of objects that are contained within the image domain. When the user searches for a closed contour from p_0, an automatic search for saddle points on U is performed. Back propagating from a saddle point p_1 to both directions will connect the saddle to the source point p_0 by two curves. Thereby, a closed contour is formed representing the complete boundary of an object.

To detect such a saddle point, we can compute the gradient $|\nabla U|$ and the Gaussian curvature $(\kappa_1\kappa_2)$, and check for $|\nabla U| < \epsilon$ and $\kappa_1\kappa_2 < 0$. Another possibility to isolate the saddle points on U is to use a simple test to determine the number of *level crossings*. Consider a small radius circle centered at a candidate point q and embedded in the horizontal plane $(x, y, U(q))$. Denote the number of level crossings to be the number of points this circle intersects with the surface $(x, y, U(x, y))$. It can be shown that this number at a saddle point is equal to four, while for most surface points it is two, and at maximum and minimum points there are no level crossings. In our implementation of the number of level crossings, for each point (i, j) in the pixels grid, we simply count the number of sign changes in $U(k, l) - U(i, j)$ while traveling around the 8 neighbors (k, l) of the point.

Although there are only few saddle points in U (see Figure 5 for example), finding the level crossing for every point q in the domain is not enough. It is necessary

to filter out the insignificant saddles that have a relatively large value of P, i.e. those that are not close to an edge, or a too large energy U.

This usually reduces the number of candidates to a relatively small number (only two remain after simple filtering of the saddles in Figure 5). Selecting the right regularization constant w will obviously filter out most of the saddles that are formed due to noise, yet will obviously introduce further constraints on w. According to our experience, selecting the right w for a smoothing effect reduces the number of saddles to the only interesting ones.

6 Examples and Results

We demonstrate the performance of the proposed algorithm (using the minimal action algorithm described in Section 3.4.3) by applying it to several real images. The images were scaled to 128×128 pixels, and the gray levels for P were normalized between 0 and 1. Parameter w is usually of the order of 0.1.

6.1 Open contour

In the first example, we are interested in a road detection between two points in the image of Figure 2. Road areas are lighter and correspond to higher gray levels. The potential function P was thus selected to be the opposite of the gray level image itself: $P = 1-I$. Minimizing this potential along a curve yields a path that follows the middle of the road. In the example of Figure 1, we show how a bad initialization for classical snakes leads to a wrong local minimum and in fact requires a very accurate initial guess, as in the bottom example, to guarantee convergence to the desired solution. It is shown that given two end points, the proposed procedure detects the path of minimal action along the right road. This example illustrates the efficiency of our approach compared to classical snakes. Given a start point p_0 on the bottom left, the image of minimal action $U(x, y)$ from this point is shown in Figure 2. Observe the way the level curves propagate faster along the road. Note, that using a completely different approach based on classical snakes, the authors of [6] have also found a way to solve efficiently the snake problem between two end points. Although their method behaves better than classical snakes, it does not ensure to converge to the global minimum and may be trapped in a bad local minimum solution as we illustrate in [10]. The interactive tool for outlining roads in aerial or medical images presented in [6] could also make use of our method between fewer constraint points or key-points to solve some cases in which there are many erroneous local minima. Our approach can be used for the minimization of many paths emerging from the same point in one single calculation of the minimal action. Figure 3 shows an application of this operation for the road image. Given a start point in the upper left area, the path achieving the global minimum of the energy is found between this point and four other given points to determine the roads graph in our previous image. In a second example, we show an application to the detection of blood vessels in a medical angiographic image of the eye fundus. Here also, the potential is obtained from

Figure 1: Local Minimum. The initial data is shown on the left and the result on the right. Results of two different initializations of the classical snakes.

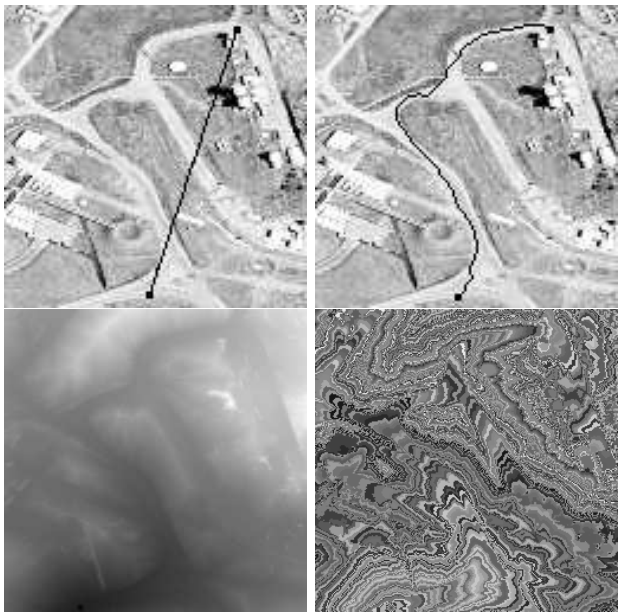

Figure 2: Top : On the right, minimal path between two points given on the left. Bottom: Minimal action U from bottom left start point. On the left, black corresponds to lower values of U, on the right a random look up table is used to render the level curves of U.

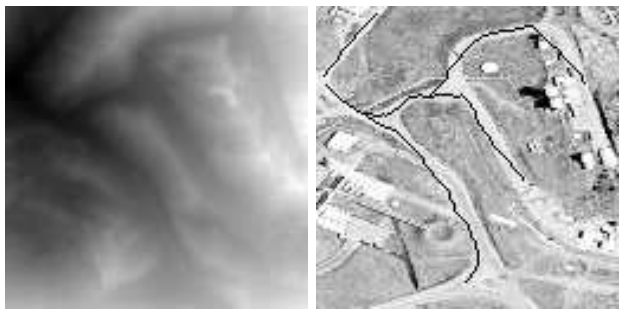

Figure 3: Many paths are obtained simultaneously connecting the start point on the upper left to 4 other points. The minimal action is shown on the left.

the image itself to detect higher gray levels. These results make use of high order ODE integrators for the back propagation as described in [22].

6.2 Closed contour

In this third example, we want to extract the left ventricle in an MR image of the heart area. The potential is a function of the distance to the closest edge in a Canny edge detection image (see Figure 5). Since we are after a closed contour, we use the saddle points classification in closing the boundaries of a single object in the heart image (see Figure 5). Given a single point, saddle point classification is used to find the second end point. The closed contour is formed of the two minimal paths joining the start and end points.

7 Concluding Remarks

In this paper we presented a method for integrating objects boundaries by searching for the path of minimal action connecting two points. The search for the global minimum makes sense only after the two end points are determined, and the 'action' or 'potential' is generated from the image data. The proposed approach makes snake initialization an easier task that requires only one or two end points and overcomes one of the fundamental problems of the active contour model, that is being trapped by an insignificant local minimum. Applying the proposed procedure to real images gave very promising results that were compared to the results obtained by other approaches that search for local solutions. The result of the proposed procedure may be considered either as the solution or as initial condition for classical snake models, or even more naturally for geodesic active contours for further refinement. In the later case, refinement to the proper solution should be almost immediate.

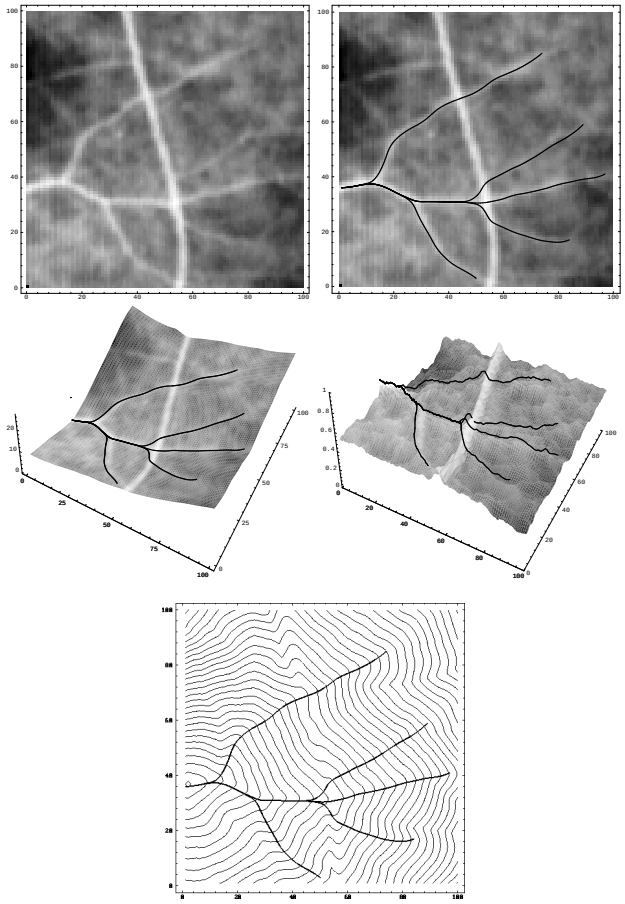

Figure 4: Finding vessels in medical angiographic image of the eye fundus: Top: original image and multiple path detection to the start point on the left of the image. Middle: the paths are superimposed on the surfaces obtained with elevation U on the left and I on the right. Bottom shows the level sets of U.

Acknowledgments.

We thank Prof. James A. Sethian, Dr. David Adalsteinsson, and Dr. Ravikanth Malladi for intriguing discussions on the fast marching method, and the authors of [6] for providing the roads image. This work is supported in part by the Applied Mathematics Subprogram of the Office of Energy Research under DE-AC03-76SFOOO98, and ONR grant under NOOO14-96-1-0381. We also thank the French-Israeli Arc-en-Ciel-Keshet program that helped us initialize this work.

References

[1] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1988.

[2] Laurent D. Cohen. On active contour models and balloons. *CVGIP:IU*, 53(2):211–218, March 1991.

[3] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.

[4] R. Malladi, J. A. Sethian, and B. C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. on PAMI*, february 1995.

[5] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *ICCV'95*, pages 694–699, Cambridge, USA, June 1995. to appear in *IJCV*.
[6] W. Neuenschwander, P. Fua, G. Szekely, and O. Kubler. Making snakes converge from minimal initialization. In *ICPR'94*, Jerusalem, Israel, 1994.
[7] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proc. Nat. Acad. Sci.*, 93(4), 1996.
[8] J. A. Sethian. A review of the theory, algorithms, and applications of level set methods for propagating interfaces. *Acta Numerica*, 1995.
[9] D. Adalsteinsson, R. Kimmel, R. Malladi, and J. A. Sethian. Fast marching method for computing solutions to static Hamilton-Jacobi equations. Technical report, LBL UC Berkeley, CA 94720, 1996, submitted for publication.
[10] Laurent D. Cohen and Ron Kimmel. Edge Integration Using Minimal Geodesics. TR 9504, Ceremade, January 1995.
[11] Demetri Terzopoulos. On matching deformable models to images: Direct and iterative solutions. In *Topical meeting on machine vision, Technical Digest Series*, volume 12, pages 160–167. Optical Society of America, 1987.
[12] Laurent D. Cohen and Isaac Cohen. Finite element methods for active contour models and balloons for 2-D and 3-D images. *PAMI*-15(11), November 1993.
[13] S. J. Osher and J. A. Sethian. Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.
[14] J. A. Sethian. A review of recent numerical algorithms for hypersurfaces moving with curvature dependent flows. *J. Differential Geometry*, 31:131–161, 1989.
[15] R. Kimmel, A. Amir, and A. Bruckstein. Finding shortest paths on surfaces using level sets propagation. *IEEE PAMI*-17(6):635–640, June 1995.
[16] R. Kimmel, N. Kiryati, and A. M. Bruckstein. Distance maps and weighted distance transforms. *Journal of Mathematical Imaging and Vision*, March 1996.
[17] M.A. Fischler, J.M. Tenenbaum, and H.C. Wolf. Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique. *CGIP*, 15:201–223, 1981.
[18] A. M. Bruckstein. On shape from shading. *Computer Vision, Graphics, and Image Processing*, 44:139–154, 1988.
[19] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *J. of Comp. Phys.*, 118:269–277, 1995.
[20] E. Rouy and A. Tourin. A viscosity solutions approach to shape-from-shading. *SIAM. J. Numer. Analy.*, 29:867–884, 1992.
[21] P. Dupuis and J. Oliensis. An optimal control formulation and related numerical methods for a problem in shape reconstruction. *Annals of Applied Probability*, 4(2):287–346, 1994.
[22] R. Kimmel and J. A. Sethian. Fast marching methods for computing distance maps and shortest paths. TR, LBL, Univ. of California, Berkeley, February 1996, submitted for publication.
[23] R. Bellman and R. Kalaba. *Dynamic Programming and modern control theory*. London mathematical society monographs, London, 1965.

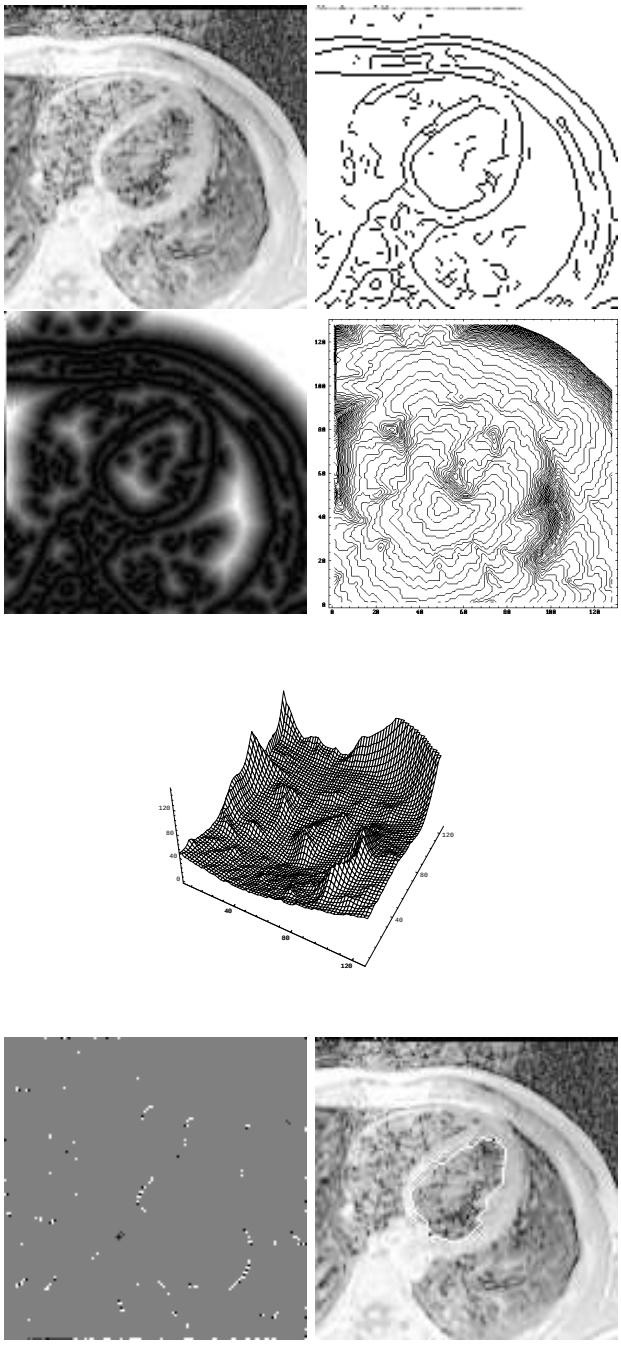

Figure 5: MRI heart image: Top: Original image on the left, edge image on the right; Middle: distance map on the left, minimal action U with its level sets on the right and represented as a graph surface below; Bottom: saddle point image on the left (after filtering, only two of the white points remain); after filtering the white pixels, the selected saddle point is used to find the two half contours on the right.