ELSEVIER

Medical Image Analysis 1 (2001) 000–000

MEDICAL IMAGE ANALYSIS

www.elsevier.com/locate/media

Fast extraction of minimal paths in 3D images and applications to virtual endoscopy

Thomas Deschamps[a,b,*], Laurent D. Cohen[b]

[a]*Medical Imaging Systems Group, Philips Research France, PRF, 51 rue Carnot, B.P. 301, 92516 Suresnes Cedex, France*
[b]*CEREMADE UMR CNRS 7534, Université Paris IX Dauphine, Place du Marechal de Lattre de Tassigny, 75775 Paris Cedex 16, France*

Received 12 July 2000; received in revised form 12 March 2001; accepted 5 June 2001

Abstract

The aim of this article is to build trajectories for virtual endoscopy inside 3D medical images, using the most automatic way. Usually the construction of this trajectory is left to the clinician who must define some points on the path manually using three orthogonal views. But for a complex structure such as the colon, those views give little information on the shape of the object of interest. The path construction in 3D images becomes a very tedious task and precise a priori knowledge of the structure is needed to determine a suitable trajectory. We propose a more automatic path tracking method to overcome those drawbacks: we are able to build a path, given only one or two end points and the 3D image as inputs. This work is based on previous work by Cohen and Kimmel [Int. J. Comp. Vis. 24 (1) (1997) 57] for extracting paths in 2D images using *Fast Marching* algorithm.

Our original contribution is twofold. On the first hand, we present a general technical contribution which extends minimal paths to 3D images and gives new improvements of the approach that are relevant in 2D as well as in 3D to extract linear structures in images. It includes techniques to make the path extraction scheme faster and easier, by reducing the user interaction.

We also develop a new method to extract a centered path in tubular structures. Synthetic and real medical images are used to illustrate each contribution.

On the other hand, we show that our method can be efficiently applied to the problem of finding a centered path in tubular anatomical structures with minimum interactivity, and that this path can be used for virtual endoscopy. Results are shown in various anatomical regions (colon, brain vessels, arteries) with different 3D imaging protocols (CT, MR).

Keywords: Deformable models; Minimal paths; Level set methods; Medical image understanding; Eikonal equation; Fast marching; Virtual endoscopy

1. Introduction

Once a path is obtained in a CT or MR image, it can be used as input for virtual endoscopy inside an anatomical object. This process consists in creating perspective views of the inside of tubular structures of human anatomy along a user-defined path. Clinicians are then provided with an alternative to the uncomfortable and invasive diagnostic procedures of real endoscopy. Ordinarily, the examination of a patient pathology would require threading a camera inside his body. This new method skips the camera and can give views of regions of the body difficult or impossible to reach physically (e.g. brain vessels), the only requirement being X-ray exposure for CT and sometimes the injection of a contrast product (dye or air) in the anatomical objects, for better detection.

A major drawback in general remains when the user must define all path points manually. For a complex structure (small vessels, colon, ...) the required interactivity can be very tedious. If the path is not correctly build, it can cross an anatomical wall during the virtual fly-through. Path construction is thus a very critical task and precise anatomical knowledge of the structure is needed to set a suitable trajectory. Our work focuses on the automa-

*Corresponding author. Tel.: +33-147-283-539; fax: +33-147-283-505.

E-mail address: thomas.deschamps@philips.com (T. Deschamps).

1361-8415/01/$ – see front matter
PII: S1361-8415(01)00046-9

tion of the path construction, reducing the need for interaction and improving performance, in a robust way, given only one or two end points and the image as inputs.

We derived an automatic path tracking routine in 3D images by mapping this path tracking problem into a minimal path problem between two fixed end points. Defining a cost function inside an image, the minimal path becomes the path for which the integral of the cost between the two end points is minimal. This minimal path problem has been studied for ages by mathematicians, and has been solved numerically using graph theory and dynamic programming (Dijkstra, 1959). Cohen and Kimmel (1997) solved the minimal path problem in 2D with a front propagation equation between the two fixed end points, using the *Eikonal* equation (that physically models wave-light propagation), with a given initial front. Their approach has much in common with Dijkstra's but it has advantage of being consistent with the continuous formulation of the problem and it avoids metrication error. Therefore, the first step is to build an image-based measure that defines the minimality property in the studied image, and to introduce it in the *Eikonal* equation. The second step is to propagate the front on the entire image domain, starting from an initial front restricted to one of the fixed points.

The minimal path technique has many advantages. It needs a very simple initialization and leads to a global minimum of a snake-like energy, thus avoiding local minima. Moreover it is fast and accurate.

The propagation is done using techniques presented in (Adalsteinsson and Sethian, 1995; Sethian, 1996), and detailed in (Sethian, 1999): the authors proposed a method to propagate this front in a quick and efficient way. They first consider the initial front implicitly defined as the zero level set of a higher-dimension function, which evolves. This formulation, called level-sets method, allows to manage front propagation problems due to complex curves and topological changes. Then it uses an algorithm called *Fast Marching*, to quickly solve this new front propagation problem.

The original contribution of this work is twofold. First, we extend the minimal path technique developed in (Cohen and Kimmel, 1997) to 3D images. We also propose various improvements for this technique that are useful for image analysis in 2D as well as in 3D. It includes techniques to make the path extraction scheme faster and easier, by reducing the user interaction (partial and simultaneous propagation, one end point initialization). These improvements are very important when dealing with 3D images, where the data volume is huge and user interaction and visualization more difficult. We also develop a new method to extract a path centered in a tubular structure. This is a general technical contribution and it may be applied to other areas as well, either for medical imaging or other types of image analysis in 2D or in 3D, in order to extract linear structures (vessels, roads, . . .).

Secondly, we adapt this technique and the several improvements to the particular problem of tubular anatomical structure extraction.

This is applied to virtual endoscopy through 3D medical images.

We show that the level set method can be efficiently applied to the problem of finding a path in virtual endoscopy with minimum interactivity. A wide range of application areas are considered from colon to brain vessels. We also propose a range of choices for finding the right input measure to the minimal path tracking.

This paper is organized as follows. In Section 2, we summarize the method detailed in (Cohen and Kimmel, 1997) for 2D images, and we extend this method to 3D. In Section 3 we give details about our improvement made on the front propagation technique, including faster path extraction schemes, reduction of the user interaction. In Section 4 we explain how to extract centered paths in tubular structures. Finally in Section 5, we show how to apply our method to virtual endoscopy for several anatomical objects.

2. Finding minimal paths in 3D images

2.1. The Cohen-Kimmel method in 2D

2.1.1. Global minimum for active contours

We present in this section the basic ideas of the method introduced by Cohen and Kimmel (1997) to find the global minimum of the active contour energy using minimal paths. The energy to minimize is similar to classical deformable models (see (Kass et al., 1988)) where it combines smoothing terms and image features attraction term (Potential P):

$$E(C) = \int_{\Omega} \{w_1 \|C'(s)\|^2 + w_2 \|C''(s)\|^2 + P(C(s))\} \mathrm{d}s, \qquad (1)$$

where $C(s)$ represents a curve drawn on a 2D image, $\Omega = [0, L]$ is its domain of definition, and L is the length of the curve. The approach introduced in (Cohen and Kimmel, 1997) modifies this energy in order to reduce the user initialization to setting the two end points of the contour C. They introduced a model which improves energy minimization because the problem is transformed in a way to find the global minimum. It avoids the solution being sticked in local minima. Let us explain each step of this method.

2.1.2. Problem formulation. Most of the classical deformable contours have no constraint on the parameterization s, thus allowing different parameterization of the contour C to lead to different results. In (Cohen and Kimmel, 1997), contrary to the classical snake model (but similarly to geodesic active contours), s represents the arc-length

parameter, which means that $\|C'(s)\| = 1$, leading to a new energy form. Considering a simplified energy model without any second derivative term leads to the expression $E(C) = \int\{w\|C'\|^2 + P(C)\}\mathrm{d}s$. Assuming that $\|C'(s)\| = 1$ leads to the formulation

$$E(C) = \int_{\Omega} \{w + P(C(s))\}\mathrm{d}s. \tag{2}$$

We now have an expression in which the internal forces are included in the external potential. In (Cohen and Kimmel, 1997), the authors have related this problem with the recently introduced paradigm of the level-set formulation. In particular, its Euler equation is equivalent to the geodesic active contours (Caselles et al., 1997). The regularization of this model is now achieved by the constant $w > 0$. This term integrates as $\int_{\Omega} w\mathrm{d}s = w \times$ length(C) and allows us to control the smoothness of the contour (see (Cohen and Kimmel, 1997) for details). We remove the second order derivatives from the snake term, leading to a potential which only depends on the external forces, and on a regularization term w.

It makes thus the problem easier to solve, and it is used in minimal paths (Cohen and Kimmel, 1997), active contours using level sets (Malladi et al., 1995) and geodesic active contours as well (Caselles et al., 1997). In (Cohen and Kimmel, 1997) and in our Appendix A is also mentioned how the curvature of the minimal path is now controlled by the weight term w. This corresponds to a first order regularization term, and the paths show sometimes angles. A second order regularization term would give nicer paths, but this is difficult to include such a term in the approach.

Given a potential $P > 0$ that takes lower values near desired features, we are looking for paths along which the integral of $\tilde{P} = P + w$ is minimal. The surface of minimal action U is defined as the minimal energy integrated along a path between a starting point p_0 and any point p:

$$U(p) = \inf_{\mathscr{A}_{p_0,p}} E(C) = \inf_{\mathscr{A}_{p_0,p}} \left\{ \int_{\Omega} \tilde{P}(C(s))\mathrm{d}s \right\}, \tag{3}$$

where $\mathscr{A}_{p_0,p}$ is the set of all paths between p_0 and p. The minimal path between p_0 and any point p_1 in the image can be easily deduced from this action map. Assuming that potential P is always positive, the action map will have only one local minimum which is the starting point p_0, and the minimal path will be found by a simple back-propagation on the energy map. Thus, contour initialization is reduced to the selection of the two extremities of the path.

2.1.3. Fast marching resolution. In order to compute this map U, a front-propagation equation related to Eq. (3) is solved: $(\partial C/\partial t) = (1/\tilde{P})\vec{n}$. It evolves a front starting from an infinitesimal circle shape around p_0 until each point inside the image domain is assigned a value for U. The value of $U(p)$ is the time t at which the front passes over the point p.

The *Fast Marching* technique, introduced in (Adalsteinsson and Sethian, 1995; Sethian, 1996), and detailed in (Sethian, 1999), was used by Cohen and Kimmel (1996), noticing that the map U satisfies the Eikonal equation:

$$\|\nabla U\| = \tilde{P}, \tag{4}$$

Classic finite difference schemes for this equation tend to overshoot and are unstable. Sethian (1999) has proposed a method which relies on a one-sided derivative that looks in the up-wind direction of the moving front, and thereby avoids the over-shooting associated with finite differences:

$$\begin{aligned}&(\max\{u - U_{i-1,j}, u - U_{i+1,j}, 0\})^2 \\ &\quad + (\max\{u - U_{i,j-1}, u - U_{i,j+1}, 0\})^2 = \tilde{P}^2_{i,j},\end{aligned} \tag{5}$$

giving the correct viscosity-solution u for $U_{i,j}$. The improvement made by the *Fast Marching* is to introduce order in the selection of the grid points. This order is based on the fact that information is propagating *outward*, because action can only grow due to the quadratic Eq. (5). Therefore the solution of Eq. (5) depends only on neighbors which have smaller values than u.

The algorithm is detailed in 3D in the next section in Table 1. The *Fast Marching* technique selects at each iteration the *Trial* point with minimum action value. This technique of considering at each step only the necessary set of grid points was originally introduced for the construction of minimum length paths in a graph between two given nodes in (Dijkstra, 1959).

Thus it needs only one pass over the image. To perform efficiently these operations in minimum time, the *Trial* points are stored in a min-heap data-structure (see details in (Sethian, 1999)). Since the complexity of the operation of changing the value of one element of the heap is bounded by a worst-case bottom-to-top proceeding of the tree in $O(\log_2 N)$, the total work is about $O(N \log_2 N)$ for the *Fast Marching* on an N points grid. Finding the shortest path between any point p and the starting point p_0 is then simply done by back-propagation on the computed minimal action map. It consists in gradient descent on U starting from p until p_0 is reached, p_0 being its global minimum.

2.2. Extension to 3D minimal paths

We are interested in this paper in finding a minimal curve in a 3D image. The application that motivates this problem is detailed in Section 5. It can also have many other applications. Our approach is to extend the minimal path method of previous section to finding a path $C(s)$ in a 3D image minimizing the energy,

Table 1
Fast marching algorithm

- Definition:
- *Alive* is the set of all grid points at which the action value has been reached and will not be changed;
- *Trial* is the set of next grid points (6-connexity neighbors) to be examined and for which an estimate of U has been computed using Eq. (8);
- *Far* is the set of all other grid points, for which there is not yet an estimate for U;
- Initialization:
- *Alive* set is confined to the starting point p_0, with $U(p_0) = 0$;
- *Trial* is confined to the six neighbors p of p_0 with initial value $U(p) = \tilde{P}(p)$;
- *Far* is the set of all other grid points p with $U(p) = \infty$;
- Loop:
- Let $(i_{min}, j_{min}, k_{min})$ be the *Trial* point with the smallest action U;
- Move it from the *Trial* to the *Alive* set (i.e. $U_{i_{min},j_{min},k_{min}}$ is frozen);
- For each neighbor (i, j, k) (6-connexity in 3D) of $(i_{min}, j_{min}, k_{min})$:
  If (i, j, k) is *Far*, add it to the *Trial* set and compute U using Table 1;
  If (i, j, k) is *Trial*, recompute the action $U_{i,j,k}$, and update it.

$$\int_{\Omega} \tilde{P}(C(s))\mathrm{d}s, \tag{6}$$

where $\Omega = [0, L]$, L being the length of the curve. An important advantage of level-set methods is to naturally extend to 3D. We first extend the *Fast Marching* method to 3D to compute the minimal action U. We then introduce different improvements for finding the path of minimal action between two points in 3D. In the examples that illustrate the approach, we see various ways of defining the potential P.

Similarly to previous section, the minimal action U is defined as

$$U(p) = \inf_{\mathcal{A}_{p_0,p}} \left\{ \int_{\Omega} \tilde{P}(C(s))\mathrm{d}s \right\}, \tag{7}$$

where $\mathcal{A}_{p_0,p}$ is now the set of all 3D paths between p_0 and p. Given a start point p_0, in order to compute U we start from an initial infinitesimal front around p_0. The 2D scheme Eq. (5) is extended to 3D, leading to the scheme

$$\begin{aligned}&(\max\{u - U_{i-1,j,k}, u - U_{i+1,j,k}, 0\})^2 \\ &\quad + (\max\{u - U_{i,j-1,k}, u - U_{i,j+1,k}, 0\})^2 \\ &\quad + (\max\{u - U_{i,j,k-1}, u - U_{i,j,k+1}, 0\})^2 = \tilde{P}^2_{i,j,k},\end{aligned} \tag{8}$$

giving the correct viscosity-solution u for $U_{i,j,k}$. The algorithm which gives the order of selection of the points in the image is detailed in Table 1.

Considering the neighbors of grid point (i, j, k) in 6-connexity, we study the solution of the Eq. (8). We note $\{A_1, A_2\}$, $\{B_1, B_2\}$ and $\{C_1, C_2\}$ the three couples of opposite neighbors such that we get the ordering $U_{A_1} \leqslant U_{A_2}$, $U_{B_1} \leqslant U_{B_2}$, $U_{C_1} \leqslant U_{C_2}$ and $U_{A_1} \leqslant U_{B_1} \leqslant U_{C_1}$. To solve the equation, three different cases are to be examined sequentially in Table 2. We thus extend the *Fast Marching* method, introduced in (Adalsteinsson and Sethian, 1995), and used by Cohen and Kimmel (1997) to our 3D problem.

3. Several minimal path extraction techniques

In this section, different minimal path extraction procedures are detailed. We present new back-propagation techniques for speeding up extraction, a one end-point path extraction method to reduce the need for interaction, and in the next section, a centering path extraction method adapted to the problem of tubular structures in images. The methods presented in this section are valid in 2D as well as in 3D and this is an important contribution that can be useful for image analysis in general, for example in radar applications (Barbaresco and Monnier, 2000), in road detection (Merlet et al., 1993), or in finding shortest paths on surfaces (Kimmel et al., 1995).

Examples in 2D are used to make the following ideas

Table 2
Solving locally the upwind scheme

Algorithm for 3D Up-Wind Scheme

1 Considering that we have $u \geqslant U_{C_1} \geqslant U_{B_1} \geqslant U_{A_1}$, the equation derived is
$$(u - U_{A_1})^2 + (u - U_{B_1})^2 + (u - U_{C_1})^2 = \tilde{P}^2. \tag{9}$$
Computing the discriminant Δ_1 of Eq. (9) we have two possibilities
- If $\Delta_1 \geqslant 0$, u should be the largest solution of Eq. (9);
  - If the hypothesis $u > U_{C_1}$ is wrong, go to 2;
  - If this value is larger than U_{C_1}, go to 4;
- If $\Delta_1 < 0$, at least one of the neighbors A_1, B_1 or C_1 has an action too large to influence the solution. It means that the hypothesis $u > U_{C_1}$ is false. Go to 2;

2 Considering that we have $u \geqslant U_{B_1} \geqslant U_{A_1}$ and $u < U_{C_1}$, the new equation derived is
$$(u - U_{A_1})^2 + (u - U_{B_1})^2 = P^2. \tag{10}$$
Computing the discriminant Δ_2 of Eq. (10) we have two possibilities
- If $\Delta_2 \geqslant 0$, u should be the largest solution of Eq. (10);
  - If the hypothesis $u > U_{B_1}$ is wrong, go to 3;
  - If this value is larger than U_{B_1}, go to 4;
- If $\Delta_2 < 0$, B_1 has an action too large to influence the solution. It means that $u > U_{B_1}$ is false. Go to 3;

3 Considering that we have $u < U_{B_1}$ and $u \geqslant U_{A_1}$, we finally have $u = U_{A_1} + P$. Go to 4;

4 Return u.

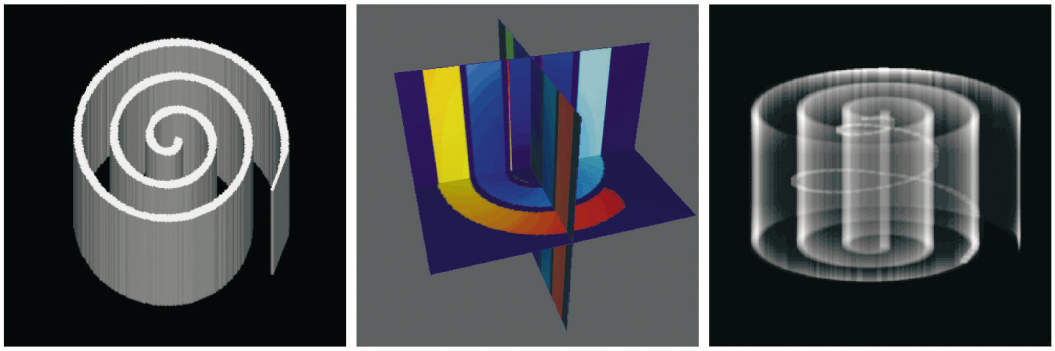


Fig. 1. Examples on synthetic potentials.

easier to understand. We also illustrate the ideas of this section on two synthetic examples of 3D front propagation in Figs. 1 and 3. Examples of minimal paths in 3D real images are presented for the application described in Section 5.

The minimal action map U computed according to the discretization scheme of Eq. (7) is similar to convex, in the sense that its only local minimum is the global minimum found at the front propagation start point p_0 where $U(p_0) = 0$. The gradient of U is orthogonal to the propagating fronts since these are its level sets. Therefore, the minimal action path between any point p and the start point p_0 is found by sliding back the map U until it converges to p_0. It can be done with a simple steepest gradient descent, with a predefined descent step, on the minimal action map U, choosing $p_{n+1} = p_n - \text{step} \times \nabla U(p_n)$. More precise gradient descent methods like Runge-Kutta midpoint algorithm or Heun's method can be used for this path extraction. A simpler descent can be choosing $p_{n+1} = \min_{\{\text{neighbors of } p_n\}} U(p)$, but it gives an approximated path in the L_1 metric. Such a descent has no more the property of being consistent. As an example, see in Fig. 3 the computed minimal action map for a 3D potential defined by $P(i, j, k) = 1 \ \forall (i, j, k)$.

See in Fig. 1-middle the action map corresponding to a binarized potential defined by high values in a spiral rendered in Fig. 1-left. The path found between a point in the center of the spiral and another point outside is shown in Fig. 1-right by transparency.

3.1. Partial front propagation

An important issue concerning the back-propagation technique is to constrain the computations to the necessary set of pixels for one path construction. Finding several paths inside an image from the same seed point is an interesting task, but in the case we have two fixed extremities as input for the path construction, it is not necessary to propagate the front on all the image domain, thus saving computing time. In Fig. 2 is shown a test on an angiographic image of brain vessels. We can see that there is no need to propagate further the points examined in Fig. 2-right, the path found being exactly the same as in Fig. 2-middle where front propagation is done on all the image domain. We used a potential $P(\mathbf{x}) = |\nabla G_\sigma \overset{*}{} I(\mathbf{x})| + w$, where I is the original image (512^2 pixels, displayed in Fig. 2-left), G_σ a Gaussian filter of variance $\sigma = 2$, and $w = 1$ the weight of the model. In Fig. 2-right, the partial front propagation has visited less than half the image. This ratio depends mainly on the length of the path tracked.

3.2. Simultaneous partial front propagation

The idea is to propagate simultaneously a front from

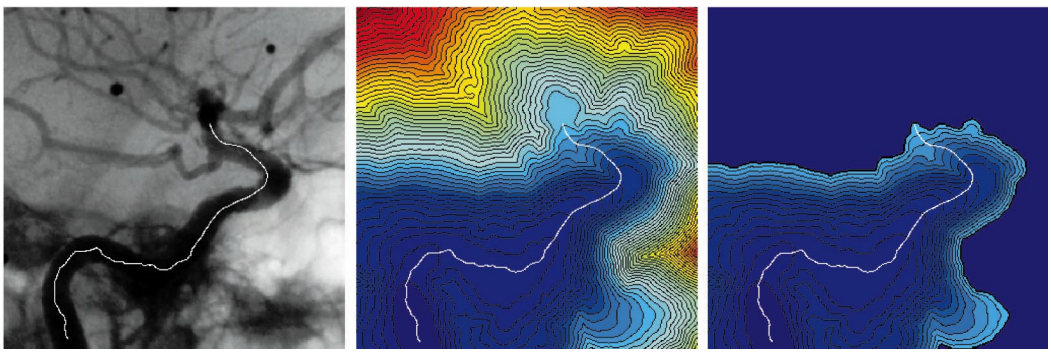


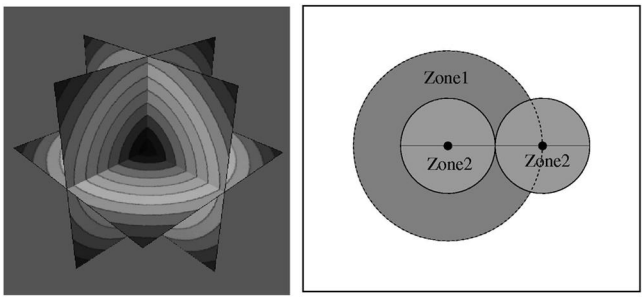
Fig. 2. Comparing complete front propagation with partial front propagation method on a digital subtracted angiography (DSA) image.

each end point p_0 and p_1. Let us consider the first grid point p where those front collide. Since during propagation the action can only grow, propagation can be stopped at this step. Adjoining the two paths, respectively between p_0 and p, and p_1 and p, gives an approximation of the exact minimal action path between p_0 and p_1. Since p is a grid point, the exact minimal path might not go through it, but in its neighborhood. Basically, it exists a real point p^*, which nearest neighbor on the Cartesian grid is p which belongs to the minimal path. Therefore, the approximation done is sub-pixel and there is no need to propagate further. This *colliding fronts* method is described in Table 3.

It has two interesting benefits for front propagation:

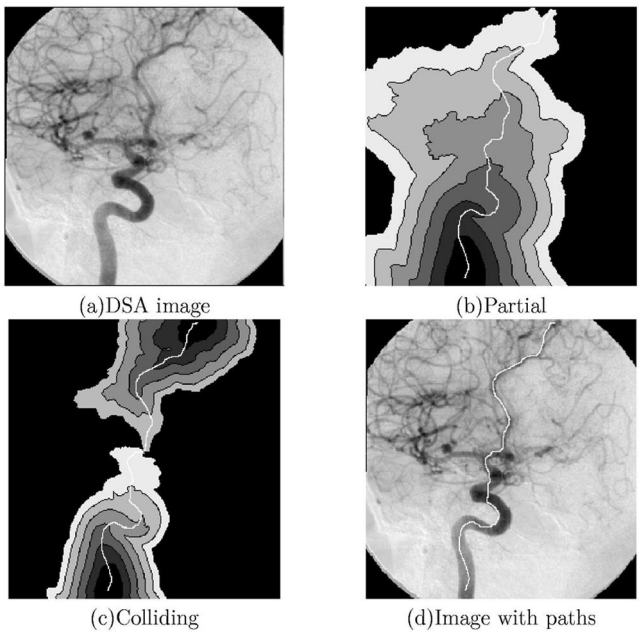
- It allows a parallel implementation of the algorithm, dedicating a processor to each propagation;
- It decreases the number of pixels examined during a partial propagation. With a potential defined by $P = 1$, the action map is the Euclidean distance.
  - In 2D (Fig. 3-right), this number is divided by $(2R)^2/2 \times R^2 = 2$;
  - In 3D (Fig. 3-left), this number is divided by $(2R)^3/2 \times R^3 = 4$.

In Fig. 4 is displayed a test on a digital subtracted angiography (DSA) of brain vessels. The potential used is $P(\boldsymbol{x}) = |I(\boldsymbol{x}) - C| + w$, where I is the original image (256×256 pixels, displayed in Fig. 4(a)), C a constant term (mean value of the start and end points gray levels), and $w = 10$ the weight of the model. In Fig. 4(b), the partial front propagation has visited up to 60% of the image. With a colliding fronts method, only 30% of the image is visited (see Fig. 4(c)), and the difference between both paths found is sub-pixel (see Fig. 4(d) where the paths superimposed on the data do not differ).



3D action with $P = 1$ Simultaneous and partial fronts in 2D

Fig. 3. 2D and 3D front propagation examples.



(a)DSA image (b)Partial (c)Colliding (d)Image with paths

Fig. 4. Comparing the partial front propagation with the colliding fronts method on a DSA image.

3.3. One end point propagation

We have shown the ability of the front propagation techniques to compute the minimal path between two fixed points. In some cases, only one point should be necessary, or the needed user interaction for setting a second point is too tedious in a 3D image. Here we derive a method that builds a path given only one end point and a maximum path length.

As we explain below, we can compute simultaneously at each point the energy U of the minimal path and its length. We choose as end point the first point for which the length of the minimal path has reached a given value. Since the front propagates faster along lower values of Potential, interesting paths are longer for a given value of U.

The technique is similar to that of Section 3.1, but the new condition will be to stop propagation when the first path corresponding to a chosen Euclidean distance is extracted. Since the front propagates in a tubular structure, all the points for which the path length criterion is reached earlier in the process are located in the same area, far from the start point. Therefore the first point for which the length is reached is located in this area and is a valuable choice as endpoint.

Table 3
Minimal path as intersection of two action maps

Algorithm
· Compute the minimal action maps U_0 and U_1 to respectively p_0 and p_1 until they have an *Alive* point p_2 in common;
· Compute the minimal path between p_0 and p_2 by back-propagation on U_0 from p_2;
· Compute the minimal path between p_1 and p by back-propagation on U_1 from p_2;
· Join the two paths found.

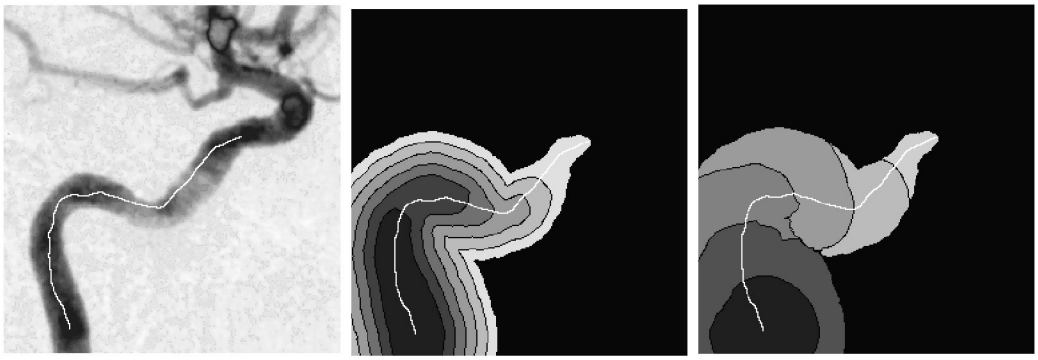


Fig. 5. Computing the Euclidean path length simultaneously.

An example of this path length condition is shown on Fig. 5 which is a DSA image of brain vessels. Propagating a front with potential $P = 1$ computes the Euclidean distance to the start point. This is obvious from definition (3), and we can see its illustration with Fig. 3-left. Therefore, we use simultaneously an image-based potential P_1, for building the minimal path and a potential $P_2 = 1$ for computing the path length.

While we are propagating the front corresponding to P_1 on the image domain, at each point p examined we compute both minimal actions for P_1 (shown in Fig. 5-middle) and for P_2 (shown in Fig. 5-right). This means Eq. (5) or Eq. (8) is solved for P_2 using the same points that are used in the scheme for P_1 in Table 2. In this case the action corresponding to P_2 is an approximate **Euclidean length of the minimal path** between p and p_0. Although this length is an approximation, it is still a good estimation since it makes use of the same *E*ikonal equation scheme. The main advantage of doing so is that it does not add much computation time to the algorithm.

Note that this Euclidean path length is discontinuous and must be smoothed in order to be used in a robust manner.

4. The path centering method

The path is the set of locations that minimize the integral of the potential in Eq. (2). If the potential is constant in some areas, it will lead to the shortest Euclidean path. The same thing happens when the potential does not vary much inside a tubular shape. The minimal path extracted is often tangential to the edges, as shown in Fig. 6, and would not be tuned for a problem which may require a centered path, like finding the optimal trajectory for virtual endoscopy.

The general framework for obtaining a centered path is the following

- Segmentation: the first goal is to obtain the edges of the tubular region;
- Centered path: once we have this segmented region, we want to find a path that is as much centered as possible in it. In order to attract the minimal path to the center of the region, we use a distance map from the segmented edges.

In the following we are going to present our method, initially presented in (Deschamps et al., 1999) and (Deschamps and Cohen, 2000), detailing each step and making comparisons with other existing techniques.

4.1. Segmentation step

In order to find the tubular structure, several approaches can be used. We can use a balloon model (Cohen, 1991) with a classical snake approach that inflates inside the object, starting with the given end point. Or we can segment the object using its correspondent level-sets implementation, as in (Malladi et al., 1995) and like the bubbles in (Tek and Kimia, 1995). In fact, this kind of region growing method can also be implemented using the *Fast Marching* algorithm. This fast approximation has already been used for segmentation in (Malladi and

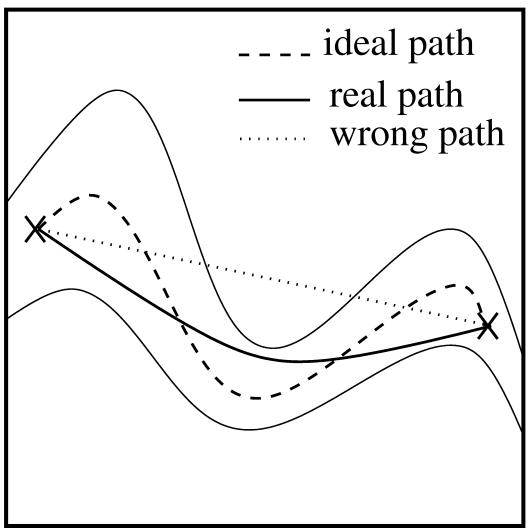


Fig. 6. Problem of path centering.

Sethian, 1998). This allows us to include the segmentation step in the same framework as our minimal path finding: having searched for the minimal action path between two given points, using a partial front propagation (see Section 3.1), the algorithm provides different sets of points:

- the points whose action is set and labeled *Alive*;
- the points not examined during the propagation and labeled *Far*;
- the points at the interface between *Alive* and *Far* points, whose actions are not set, and labeled *Trial*.

This last category, the border of the visited points, is a contour in 2D and a surface in 3D which defines a connected set of pixels or voxels. If the potential is a lot higher along edges than it is inside the shape, the edges will act as an obstacle to the propagation of the front. Therefore, the front propagation can be used as a segmentation procedure, recovering the object shapes. In this case the *Trial* points define a surface which can be described as a rough segmentation. Once the front has reached the endpoint, we use the front itself to define the edges.

4.2. Centering the path

Having obtained this interface of *Trial* points, we now want the information of distance to the edges. This information can be either used for a skeletonization, computing the medial-axis transform, or used as a new snake energy, that constrains the path in the center of the tubular shape.

In order to compute this distance, we can use a second front propagation procedure. The edges ares stored in the min-heap data-structure (see (Sethian, 1999) for details), and this is a very fast re-initialization process to compute this distance. The potential and the initial action for this second front propagation are defined as follows:

$P(i, j) = 1 \quad \forall (i, j)$ inside the shape,
$P(i, j) = \infty \quad \forall (i, j)$ outside the shape,
$U(i, j) = 0 \quad \forall (i, j) \in \{Trial\}$ points of Section 4.1,
$U(i, j) = \infty$ elsewhere.

Starting the front propagation from all the points stored in the min-heap data-structure, we compute the distance map, said $\mathscr{E}$, very quickly, visiting only the pixels inside the tubular object.

Our distance map $\mathscr{E}$ is used to create a second potential P_1. Choosing a value d to be the minimum acceptable distance to the walls, we propose the following potential:

$$P_1(\boldsymbol{x}) = \max(d - \mathscr{E}(\boldsymbol{x}); 0)^{\gamma}. \tag{11}$$

We use it as a potential for a new front propagation approach: P_1 weights the points in order to propagate faster a new front in the center of the desired regions. This final propagation produces a path centered inside the tubular structure in a very fast process.

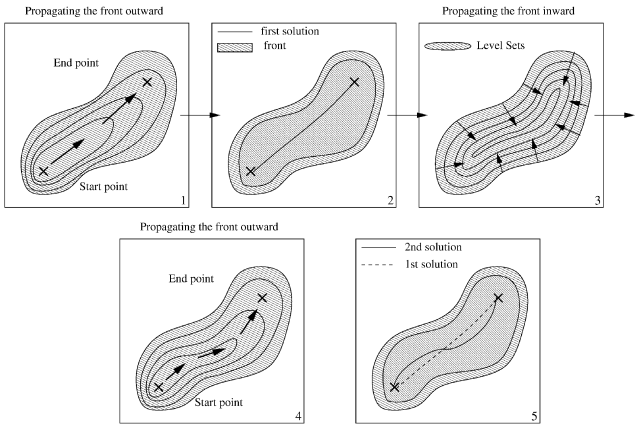


Fig. 7. Centering the path inside the object.

4.3. Description of the method

The complete method is described in Fig. 7.

- Segmentation: the first step is to compute the weighted distance map given the start and end points. It is obtained by front propagation from the start to the end point. Notice also that the end point can be determined automatically by a length criterion as in Section 3.3.
- Segmentation: the second step is to consider the set of points which have same minimal action as the endpoint. For this, we store the front position (set of trial points) at the end of the first step.
- Centering Potential: the third step is to compute the distance map $\mathscr{E}$ to the boundary front inside the tubular region. For this we propagate inward the front with a uniform potential $P = 1$. This gives the higher values towards the center of the object.
- Centered path: the fourth step is to find the minimal path between start and end points relatively to the distance potential P_1 defined in (11) computed from the previous step. This is obtained by applying again the minimal path technique. The front is now pushed to propagate faster in the center of the object.
- Centered path: the final step is to make back-propagation from the end point using the last minimal action map.

An interesting improvement is that the value of the weight w can be automatically set to a very low value:

- During the first propagation the regularity of the path is not important, and w can be very small;
- During the second propagation, $P' = P + w = 1$;
- During the final propagation the potential based on the distance to the object walls is synthetic and leads to smooth paths even if $w \ll 1$.

As an illustration, a test is proceeded on a DSA image of the brain vessels shown in Fig. 8-left. In Fig. 8-center is shown the result obtained using a potential based on the image, where the shortest path is tangential to edges. But

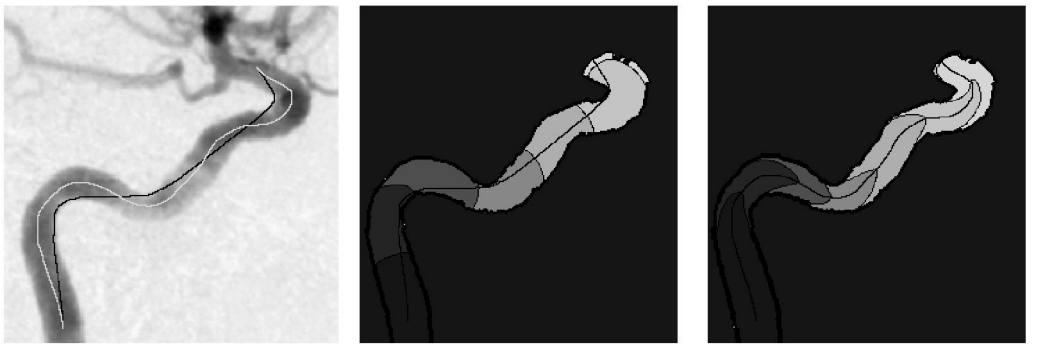


Fig. 8. Comparing classic and centered paths.

the front propagates only along the vessel direction, and is rapidly stopped transversally, allowing to compute the distance to the walls. Defining a new potential according to Eq. (11) based on this distance map, the second front propagates faster in the center of the vessel. Due to the shape of the iso-action lines of the centered minimal action shown in Fig. 8-right, the path avoids the edges and remains in the center of the vessel. We will present results on real 3D data in Section 5.2 applied to the problem of virtual endoscopy (see Fig. 16).

4.4. Comparison with other work

Another method to obtain a centered path would be to make a classical snake minimization on the centering potential P_1, starting from the path obtained previously, like it is done in the thesis of (Cuisenaire, 1999), a nice application indicated by one of our reviewers. But too much smoothing may lead to a wrong path. For example, in the case of thin tubular structures, smoothing the path may lead it outside the tubular structure. Also, the unpublished work presented by Cuisenaire (1999) details an algorithm which is applied to a tubular object which is already manually segmented by the user, whereas our method comprises both steps of segmentation and centering.

Another category of very similar centered line extraction technique is skeletonization, and particularly the definition of the medial axis function of Blum (1967) which treats all boundary pixels as point sources of a wave front. Considering that the *Fast Marching* computes the Euclidean distance to an arbitrary set of points using a potential $P = 1$, it can also be used for skeletonization.

However, the purpose of our application is to have a smooth line which always stays inside the tubular object and which is far from the edges. This is motivated by the application to virtual endoscopy (see the next section).

If one wishes to achieve this task with a skeletonization, like in (Yeorong et al., 1999), he will need and rely on the results of post-processing techniques in order to obtain a unique and smooth path inside this segmented object. Smoothing and removing undesirable small parts of the skeleton can be done using techniques shown in (Tek and Kimia, 2001). The main advantage of our approach is that it gives only one smooth and centered path in a unique and fast process. Therefore, it cannot be replaced by a simple medial-axis transform.

In (Paik et al., 1998), the authors extract first the surface of the colon, then compute a minimal path on this surface and move this initial path to the center of the object by applying a thinning algorithm to the object segmented and projecting the path on the resulting surface. The algorithm developed by Kimmel et al. (1998) can be applied to their methods since it computes the minimal path on a surface defined by a manifold. Although it seems to produce a smooth centered line, the thinning algorithm is computationally inefficient, compared to the speed of our algorithm that needs less than a minute on a classical inexpensive computer (300 MHz CPU).

In the different techniques quoted, the main difference with our method lies in the fact that the object is manually segmented by the user. Our method comprises steps of segmentation and path extraction, and achieves them in a very fast way. More than a robust and fast method, we have developed a tool that is used for segmentation, minimal path tracking, and even potential definition. The main advantage of our approach is that it comprises all those steps and gives only one smooth and centered path in a unique and fast process.

5. Application to virtual endoscopy

In previous sections we have developed a series of issues in front propagation techniques. We study now the particular case of virtual endoscopy, where fast extraction of centered paths in 3D images with minimum user interactivity is required.

5.1. Targets for virtual endoscopy

Visualization of volumetric medical image data plays a crucial part for diagnosis and therapy planning. The better the anatomy and the pathology are understood, the more efficiently one can operate with low risk. Different possibilities exist for visualizing 3D data: three 2D orthogonal views (see Fig. 9), maximum intensity projection (MIP, and its variants), surface and volume rendering. In particular, virtual endoscopy allows by means of surface/volume rendering techniques to visually inspect regions of the body that are dangerous and/or impossible to reach physically with a camera (e.g. behind an airway stenosis or obstruction, or too small). An extensive definition virtual endoscopy can be found in (Jolesz et al., 1997).

Virtual endoscopy techniques can be divided into two groups of methods that can collaborate:

- techniques which deal with simulation of a real endoscope motion; In this case, virtual endoscopy is very interactive, simulating the motion of a camera inside the body, based on an extracted anatomical object that is modeled using rigid body dynamics; a good example of this simulation is presented in (Hong et al., 1997).
- techniques which focus on the observation of the interior of anatomical objects by extracting trajectories inside them, see (Yeorong et al., 1999) for an example.

In this article we have decided to focus on the second kind of techniques. However, the minimal path techniques can also be useful for the first kind of methods: Kimmel and Sethian have applied the Fast-Marching algorithm for a robotic application in (Sethian, 1999), for the motion of an object with a certain shape and orientation in an image with obstacles. They have discretized the *E*ikonal equation in a space that describes the object position and orientation and added a dimension to the problem that could lead to huge computing costs for an interactive 3D application.

For the second kinds of virtual endoscopic technique, the system is composed of two parts:

1. A Path construction part, which provides the successive locations of the fly-through in the tubular structure of interest (see Fig. 10-left);
2. Three dimensional interior viewing along the endoscopic path. Those views are adjoined creating an animation which simulates a virtual fly-through through them (see Fig. 10-right).

Original CT slice + path Endoscopic view

Fig. 10. Interior view of a colon, reconstructed from a defined path.

A major drawback in general remains when the path construction is left to the user who manually has to ''guide'' the virtual endoscope/camera. The required interactivity can be very tedious for complex structures such as the colon for example (see Fig. 11). Actually, on most clinical platforms the user must define all path points manually, using for example three 2D orthogonal views, as shown in Fig. 9, leading to problems as the following:

- Since the anatomical objects have often complex shapes, they tend to pass in and out of the three orthogonal planes. Consequently the right location is accomplished by successively entering the projection of the desired point in each of the three planes;
- The path is approximated between the user defined points by lines or Bezier splines. If the number of points is not sufficient, it can easily cross an anatomical wall.

Path construction in 3D images is thus a very critical task and precise anatomical knowledge of the structure is needed to set a suitable trajectory, with the minimum required interactivity.

Numerous techniques try to automate this path construc-

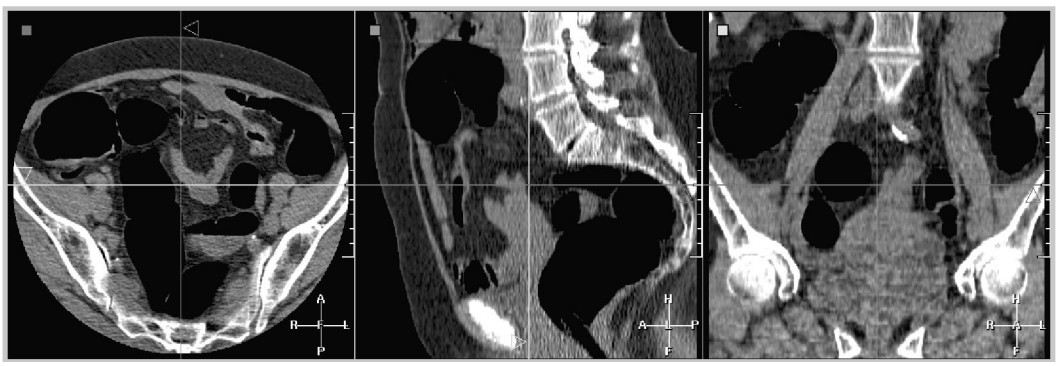


Fig. 9. Three orthogonal views of a volumetric CT data set of the colon.

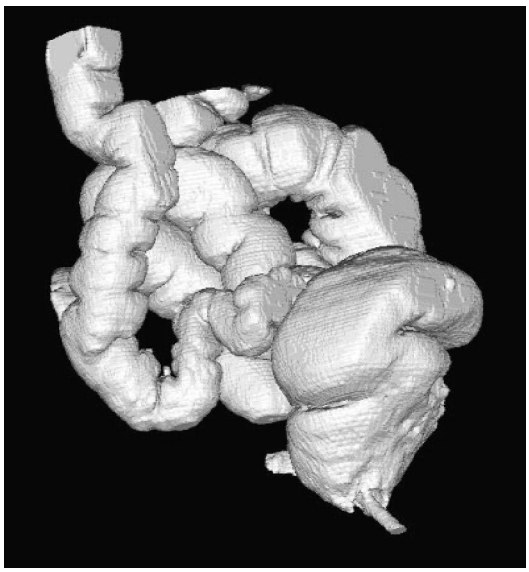


Fig. 11. The complex shape of the colon.

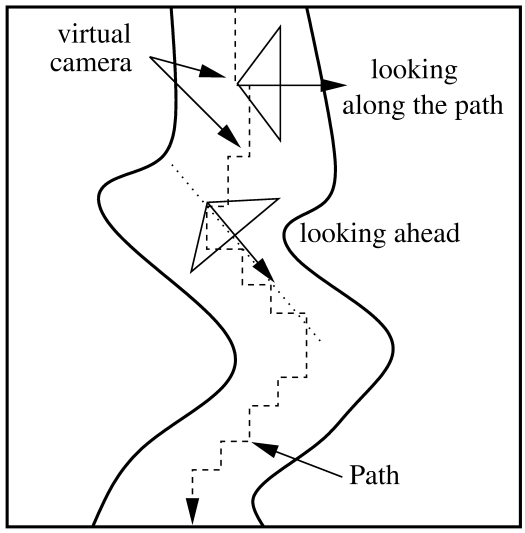


Fig. 12. Orientation of the virtual camera.

tion process. Most of them use a skeletonization technique, like in (Yeorong et al., 1999), in order to extract a centerline in the dataset. But extracting the skeletons of an anatomical shape requires first to segment it. And the skeleton often consists in lots of discontinuous trajectories, and post-processing, as done in (Tek and Kimia, 2001) is necessary to isolate and smooth the final path. The front propagation techniques studied in this paper in contrast to other methods does not require any pre- or post-processing as explained in Section 4.

It is sometimes necessary to smooth the path extracted by the front propagation. The point of view in the volume rendering of the tubular structure is very important, because it constrains the result of the examination. Thus, during the virtual fly through, the point of view of the camera must change smoothly. Traditionally, the position of the virtual camera frame at a particular path point is orthogonal to the path. If the path is not smooth, the point of view of the virtual camera will change in an abrupt manner. There are two ways to achieve this regularization:

- by modifying the view angle of the virtual camera, being no more orthogonal to the path, but looking in the direction of a path point which is located far from its current position (see Fig. 12), or using a running average of the local direction of the camera;
- by increasing the weight w in Eq. (2) since it has a smoothing effect on the minimal path (see Appendix A for details). We preferred to use this technique in the following examples, since it is efficient and very simple to add.

We first apply the minimal path construction to the case of virtual endoscopy in the colon in Section 5.2, then we extend this technique to other anatomical shapes in Section 5.3.

5.2. Building a potential for virtual colonoscopy

All tests are performed on a volumetric CT scan of size $512 \times 512 \times 140$ voxels, shown in Fig. 9. The grey level range is between 0 and 1500. The target is to build a potential P with the 3D data set allowing paths to stay inside the anatomical shapes where end points are located. We thus define the potential by a general model $\tilde{P}(x) = |I(x) - I_{\text{mean}}|^{\alpha} + w$.

First, the potential must be lower inside the colon in order to propagate the front faster, and to avoid problems with crossing the edges of the anatomical object. In a colon CT scan, an average position I_{mean} of the colon grey level in the histogram can be defined (see Fig. 13) as a peak in the histogram where $I_{\text{mean}} = 200$. Secondly, if the path to be extracted is very long, the situation can lead to pathological cases, and the front can go through potential *walls*. This is frequent for large objects that have complex shapes and very thin edges, as colon. Then, edges should be enhanced to enable long trajectories, with a non-linear function. We thus take $\alpha = 2$ in order to enhance the dynamic of the image with a quadratic function.

However, this potential does not produce paths relevant for virtual endoscopy. Indeed, paths should remain not only in the anatomical object of interest but as far as

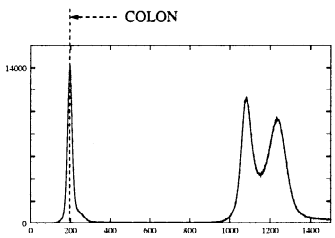


Fig. 13. Localization of the colon in the histogram.

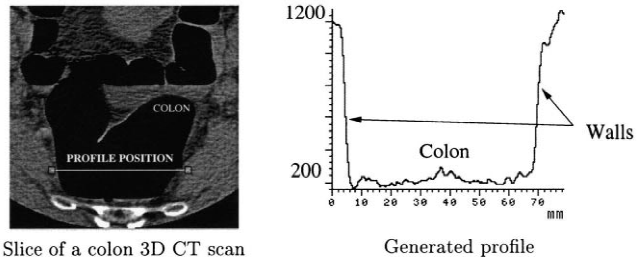


Fig. 14. Profile of the colon volume.

possible from its edges. In order to achieve this target, we use the centering potential method as detailed in Section 4. We first need to obtain a *shape* information. In fact, a CT scan of the colon contains already a shape information sufficient to constrain a front propagation. In Fig. 14-left is shown a slice of a colon volumetric data set. Fig. 14-right shows the grey level profile along the line drawn in Fig. 14-left. Air fills the colon and is represented in our CT image by a grey level around 200 (see Fig. 14-right), while edges are defined by a grey intensity around 1200. Then, using the potential $\tilde{P}(x) = |I(x) - I_{\text{mean}}|^{\alpha} + w$, the front obtained through *Fast Marching* is stopped by the anatomical shapes, as seen in Fig. 15. It illustrates the fact that the *Fast Marching* can act also as a segmentation tool, as noted in Section 4.

In Fig. 16 we show the result of applying this new method to colonoscopy. The edges are obtained via a first propagation: in Fig. 15 we can see the evolution of the *narrow band* during propagation. It gives a rough segmentation of the colon and provides a good information

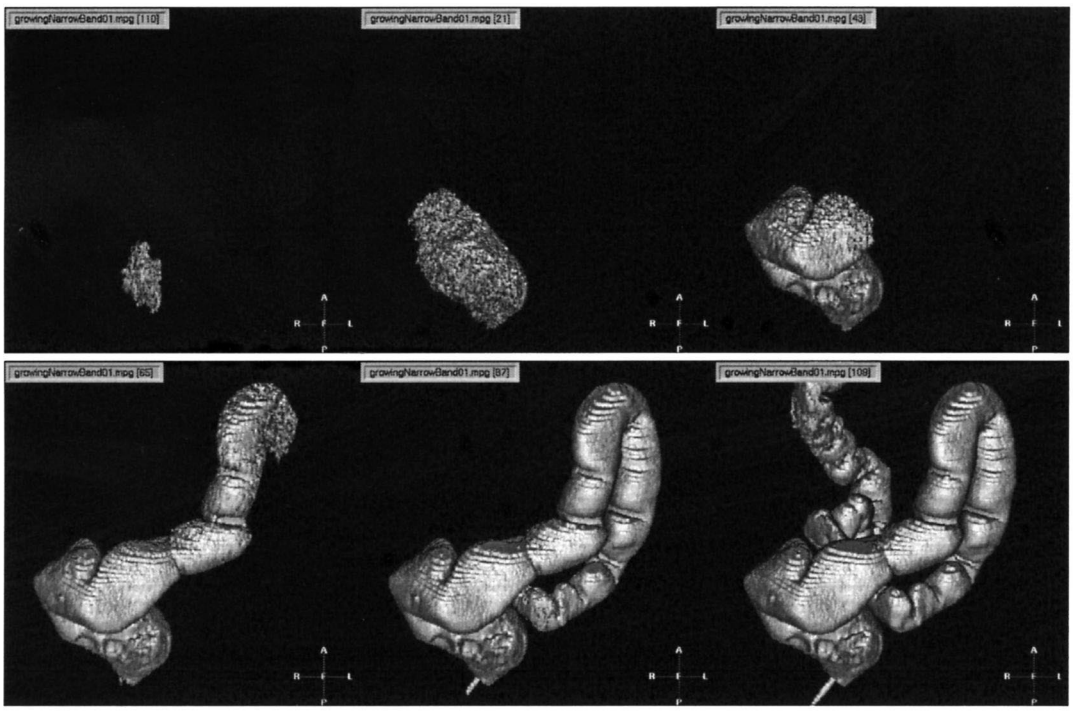


Fig. 15. Propagating inside the colon volume.

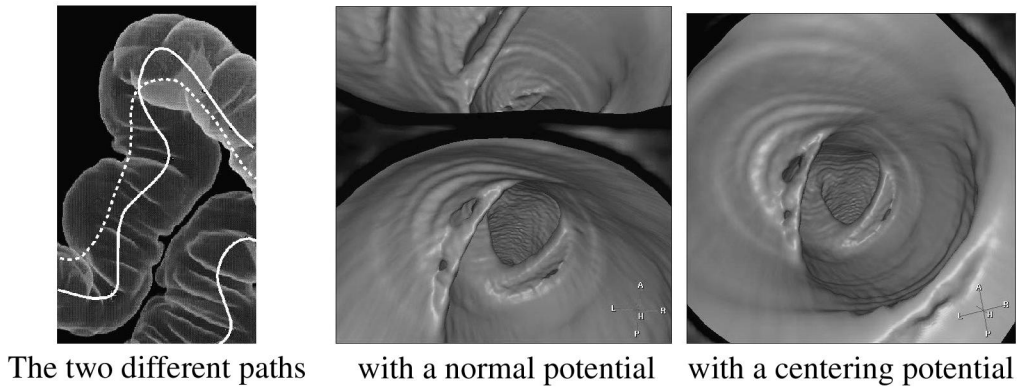


Fig. 16. Centering the path in the colon.

and a fast re-initialization technique to compute the distance to the edges.

Using this distance map as a potential (from Eq. (11)) that indicates the distance to the walls, we can correct the initial path as shown in Fig. 16-left: the new path remains more in the middle of the colon. And the value of the parameter d can be derived from anatomical characteristics. If we know approximately the section of the colon along the path we can easily choose a value to stay in the center of the tubular structure.

The two different Figs. 16-middle and 16-right display the view of the interior of the colon from both paths shown in Fig. 16-left. With the initial potential, the path is near the wall, and we see the u-turn, whereas with the new path, the view is centered into the colon, giving a more correct view of the inside of the colon. The new centered path is smooth because this final propagation is done on a synthetic potential (the distance to the walls) where noise has been removed.

Therefore, the two end points can be connected correctly, giving a path staying inside the anatomical object. But for virtual colonoscopy, it is often not necessary to set the two end points within the anatomical object. The colon being a closed object with two extremities, we can use the Euclidean path length stopping criterion as explained in Section 3.3. Fig. 15 shows the front propagation in the *Fast Marching* technique with a starting point belonging to the colon and an Euclidean path length criterion of 500 mm. The image resolution is 1 mm for x and y axes and 4 mm for the z axis. Fig. 17 shows the minimal path obtained. Fig. 21 shows rendered views from a few points along the path.

Interaction is limited to setting the start point for front propagation and choosing the minimal path length (as explained in Section 3.3). It takes 30 seconds of computing time for building the complete path on an Ultra 30 (with 300 MHz CPU and 1 Go RAM), comprising steps of segmentation of the colon and calculation of the distance to the walls in order to center the path as detailed in Section 4. The complete virtual fly-through renderings (300 images) are computed in approximately 10 minutes (the rendering is a tool included in the EasyVision workstation developed by Philips Medical Systems).

5.3. Results on Other Anatomical Objects

5.3.1. Trachea CT scan

Extracting paths inside the trachea is the same problem as in the colon. The dataset used is shown in Fig. 18 by means of three orthogonal slices of the volume displayed together with a path extracted. Air fills the object and give a shape information all along from throat to lungs. Therefore, the anatomical object having a very simple shape, the path construction with one or two fixed points is easier than in the colon case. One example path tracks the trachea, using a nonlinear function of the image grey levels ($\tilde{P}(x) = |I(x) - 200|^2 + 1$). Two views of an extracted path in 3D are displayed in Fig. 18 together with 3 orthogonal slices of the dataset. An endoscopic view along the path is displayed in Fig. 21.

5.3.2. Brain magnetic resonance angiography (MRA) image

Tests were performed on brain vessels in a MRA scan. Three orthogonal slices of this dataset are shown in Fig. 19 together with a path extracted.

The problem is different, because there is only signal from the dye in the cerebral blood vessels. All other structures have been removed. The main difficulty here lies in the variations of the dye intensity. The example path tracks the superior sagittal venous canal, using a nonlinear function of the image grey levels ($\tilde{P}(x) = |I(x) - 100|^2 +$

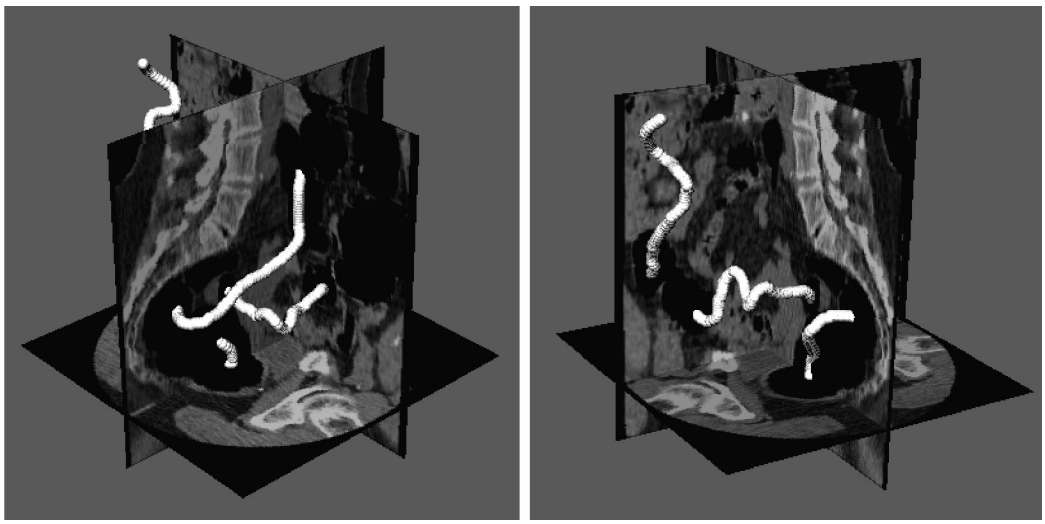


Fig. 17. Views of the minimal path inside the colon volume.

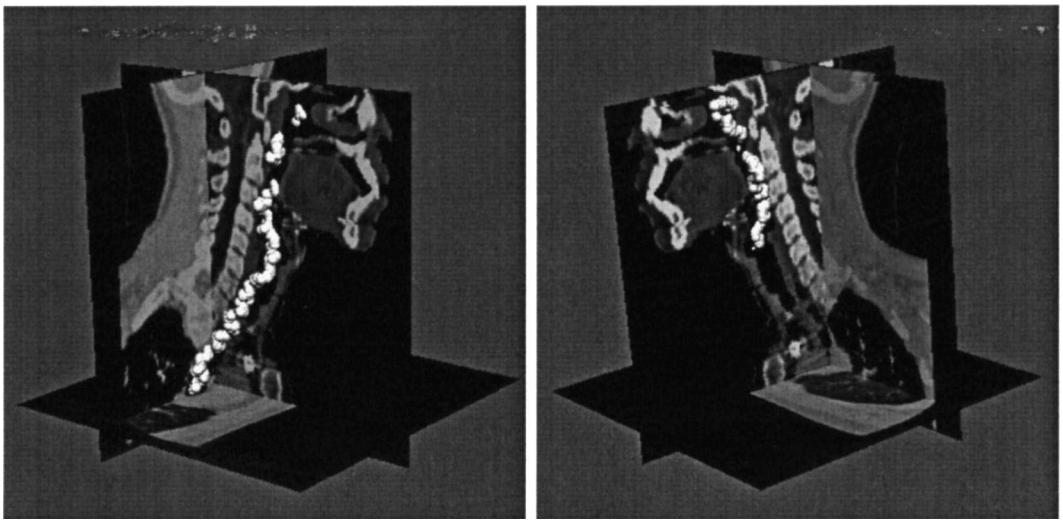


Fig. 18. Views of the minimal path inside the trachea.

1). Two views of the extracted path in 3D are displayed in Fig. 19 together with 3 orthogonal slices of the dataset. A sample of the virtual fly-through along the brain vessel is displayed in Fig. 22.

5.3.3. Aorta MR scans

A test was made on an aorta MR dataset, shown in Fig. 20. The propagation measure is based on a nonlinear function of the intensity of the contrast solution that fills the aorta. This data set is difficult since the intensity of the contrast product will vary along the aorta (the contrast bolus dilutes during the acquisition time). Due to this non-uniformity, paths can cross other anatomical structures with similar intensities if the mean value inside the aorta is not set correctly by the user.

Our example path tracks one illiaca, using the potential $\tilde{P}(x) = |I(x) - 1000|^2 + 10$ in the MR scan. The dataset contains noise, and we must use an important weight to smooth the extracted paths. We have displayed a sample of the endoscopic views of the aorta along the path in Fig. 22.

6. Conclusion

In this paper we presented a fast and efficient algorithm that computes a path useful for guiding endoscopic viewing that only depends on a start and end point. This work was the extension to 3D of a level-set technique developed in (Cohen and Kimmel, 1997) for extracting paths in 2D images, given only the two extremities of the path and the image as inputs, with a front propagation equation. We improved this front propagation method by creating new

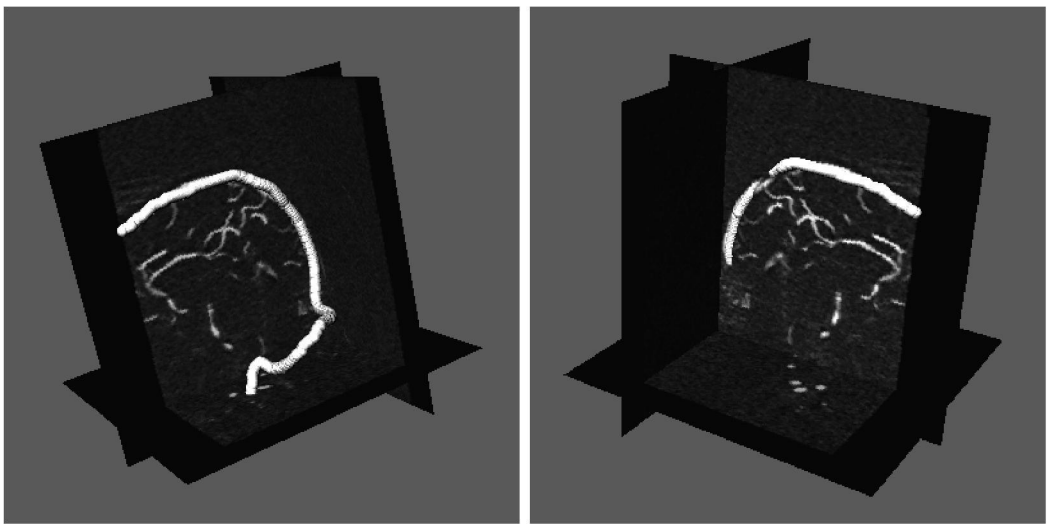


Fig. 19. Views of the minimal path inside a brain vessel.

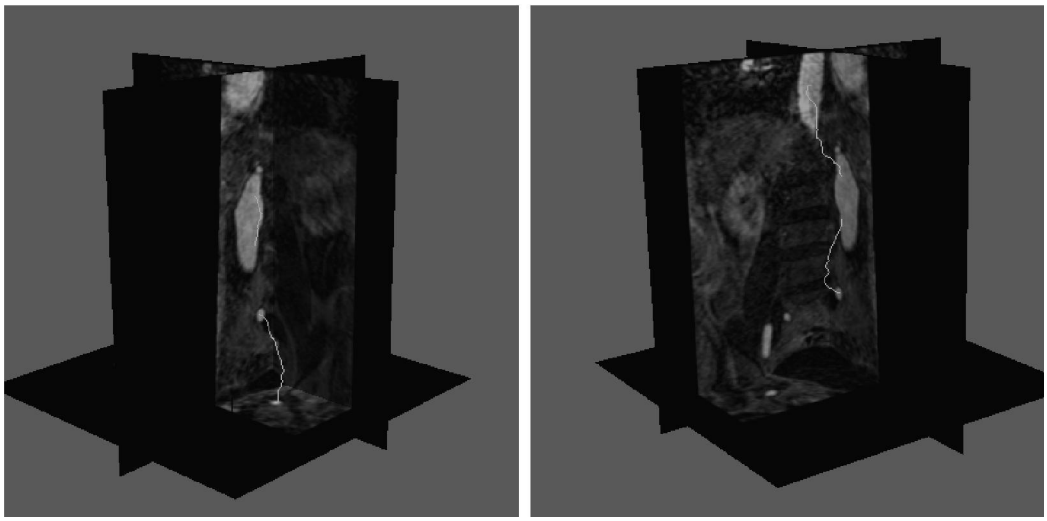


Fig. 20. Views of the minimal path inside the MR dataset of the aorta.

algorithms which decrease the minimal path extraction computing cost, and reduce user interaction in the case of path tracking inside tubular structures. We have proved the benefit of our method towards manual path construction, showing that only a few seconds are necessary to build a complete trajectory inside the body, giving only one or two end points and the image as input.

Concerning validation of the results, first we have noticed the enthusiasm of clinicians who have either seen demos of our work or have used it. Moreover, we have obtained such good results for very different kinds of medical images. We can assess of the fact that our paths are acceptable from the good quality of the virtual endoscopy video generated. Indeed, our work has been integrated in the next version of EasyVision Workstation delivered by Philips Medical Systems. A more thorough systematic validation is currently made by colleagues at Philips Medical Systems together with clinicians.

Future works will focus on the definition of potentials for objects with non-uniform grey-level contrast where the success of the tracking approach critically depends on the design of the cost function. It will also include the generalization of the path extraction techniques to tubular anatomical structure with branches, like arterial and bronchial trees.

7. Videos

Videos of Virtual Endoscopy Fly Through using our minimal path technique described in this paper is available on the following web pages: http://www.ceremade.dauphine.fr/~cohen/MPEG

It includes the four following sequences:

- Aorta Fly Through
- Colon Fly Through
- Brain Fly Through
- Trachea Fly Through

Acknowledgements

We thank Drs. Jean-Michel Létang and Sherif Makram-Ebeid for fruitful collaborations, and very interesting discussions. We thank Jean Pergrale, group leader of the Medical Imaging Systems Group at Philips Research France, for constant support. And we thank Roel Truyen, Dr. Bert Verdonck and all the MIMIT team of Dr. Frans Gerritsen at Philips Medical Systems, The Netherlands, for providing datasets and helpful ideas on the subject.

Appendix A. Paths of minimal action

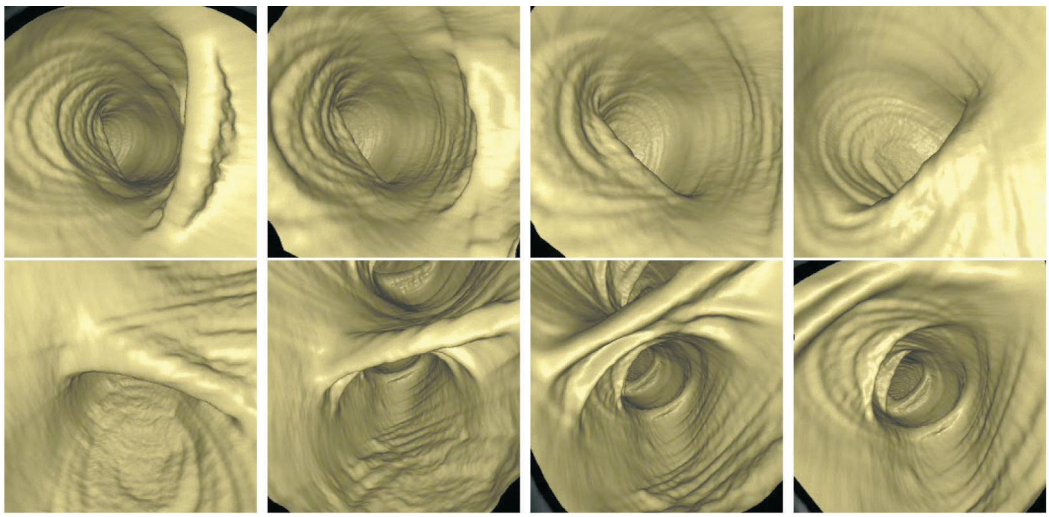
We give here some remarks and comments on the minimal path approach described in Section 2 and introduced for 2D in (Cohen and Kimmel, 1997).

A.1. Understanding the role of the potential map

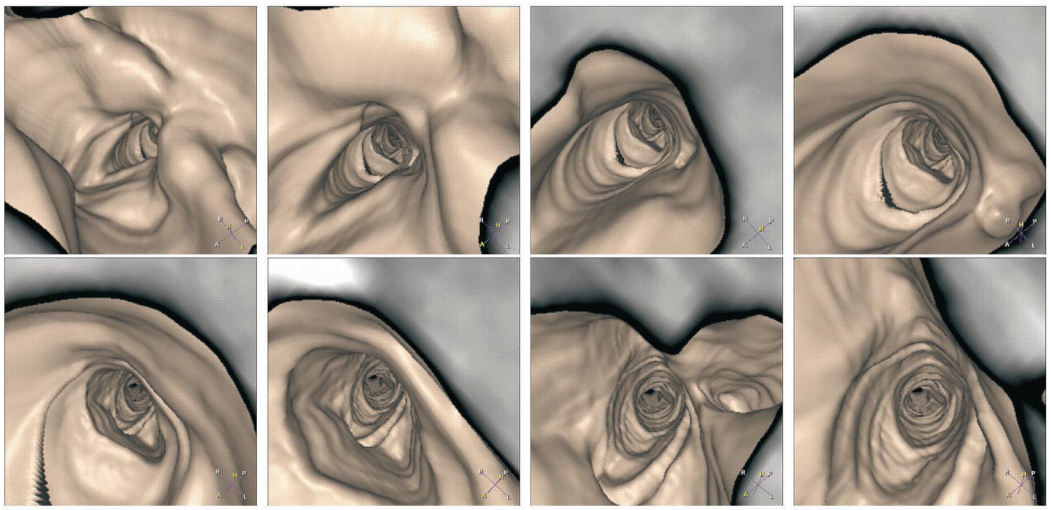
The aim of the potential used in Eq. (4) is to propagate the front in the desired regions, in order to extract a minimal path corresponding to the wanted features.

In Fig. 23(a) one can see the iso-action lines of the surface of minimal action provided by a front propagation on a univalued potential. Visualization is focused on the lines of iso-action. Without any obstacle, the front is propagating in every direction at the same speed. The corresponding iso-action lines are circles, and their radius is the Euclidean distance to the start point. The minimal paths are straight lines.

The potential is multi-valued in Fig. 23(b), the higher value being the upper-half part of the image. One can



Volume rendering in a Colon 3D CT scanner



Volume rendering in a trachea 3D CT scanner

Fig. 21. Virtual endoscopy in the colon and in the trachea.

easily see that the front propagation speed is quicker in the lower half part, because the space between the iso-action lines (level sets of the surface) is bigger. The minimal paths are piecewise linear.

This is similar to *Fermat*'s principle on the minimality of the light path: we can observe on Fig. 23(a) that paths are straight lines in homogeneous media, and that paths are deviated at the junction between two different homogeneous media on Fig. 23(b). The path joining the point in the middle right corresponds to the well-known mirage effect.

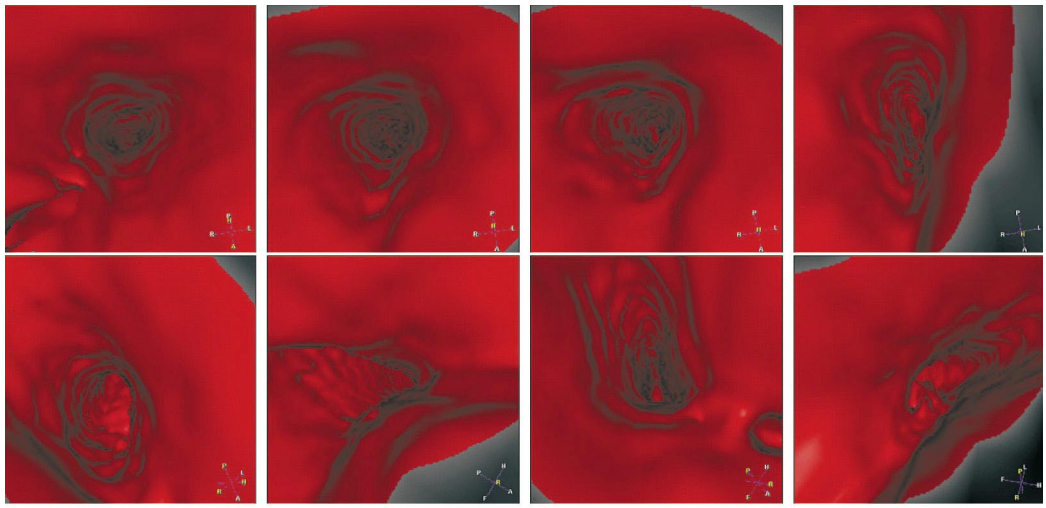
A.2. The regularity of the path

In (Cohen and Kimmel, 1997), it is proven that weight w in Eq. (2) can influence curvature and be used as a smoothing term. An upper bound for the curvature magnitude $|\kappa|$ along the minimal path is found, $\mathcal{I}$ being the image domain:

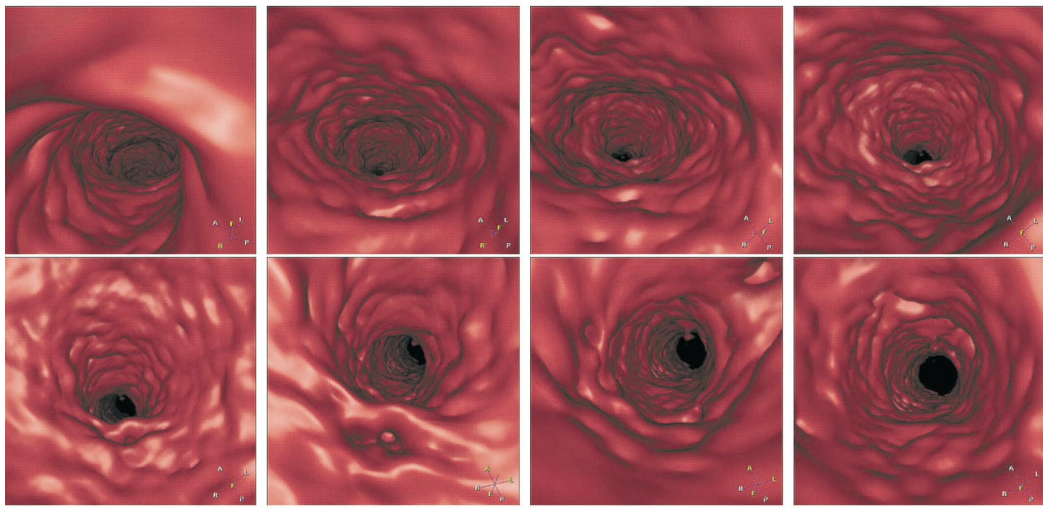
$$|\kappa| \leqslant \frac{\sup_{\mathcal{I}} \|\nabla P\|}{w}. \tag{A.1}$$

A.2.1. Influence on the gradient descent scheme. The exact minimal path is obtained with a gradient descent. But care must be paid on the choice of the gradient step to avoid oscillations.

If the weight w is set to a small value ϵ the extracted path length is not limited at all, nor the curvature magnitude in Eq. (A.1). Therefore in zones where the action map is flattened, the slope being as small as ϵ, the path can have a spaghetti-like trajectory. The minimal path being obtained by steepest gradient descent, directions are evalu-



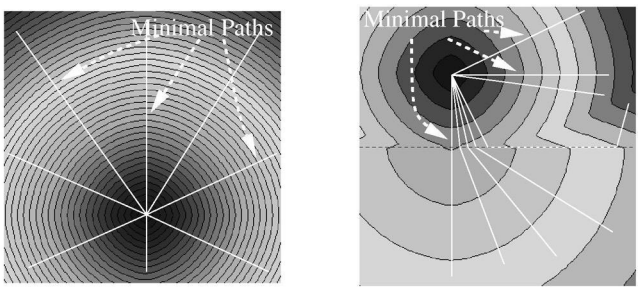
Volume rendering in a brain 3D MR image



Volume rendering in an aorta 3D MR image

Fig. 22. Virtual endoscopy in the brain vessels and in the aorta.

ated by interpolation based on nearest neighbors on the Cartesian grid. If the discrete gradient step $\Delta \boldsymbol{x}$ is too large, the approximation of this trajectory will produce oscillations between relative positions. Those oscillations can lead to a huge number of path points larger than forecasted allocations.



(a) Univalued potential (b) Bivalued potential

Fig. 23. Propagation and minimal paths on synthetic cases.

We have made a test on a region of the data shown in Fig. 24-left where the steepest gradient fails (with a number of path points limited). The cost map when tracking a vessel is displayed in Fig. 24-middle. Taking $w = 0.1$ leads to a curvature magnitude $\kappa \leqslant 10^3$. The steepest gradient scheme oscillates, for a given step size, and stops as shown in Fig. 24-right. Therefore, increasing w maintains a lower upper-bound on the curvature magnitude and makes the steepest gradient descent scheme robust. Another method is to use more robust gradient descent techniques like Runge-Kutta where the step size of the gradient descent can be locally adapted.

A.2.2. Influence on the number of points visited. This

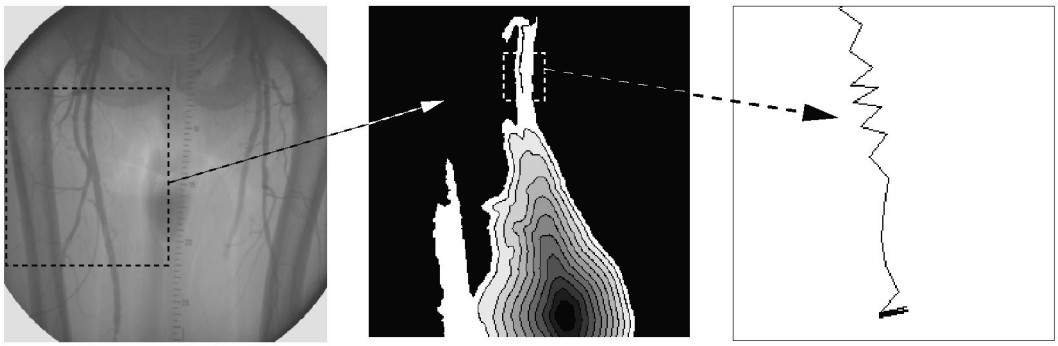


Fig. 24. Failure of the steepest gradient descent on a bolus chase reconstruction data.

section illustrates the influence of the weight w of Eq. (2) on the necessary number of voxels visited for a path extraction. In Figs. 25 is shown the tracking of a vessel in a X-Ray image of the femoral vessels, using different weights $w_1 = 1$ and $w_2 = 20$. The smoothing done by increasing the weight can be observed in a zoom on the paths shown in Fig. 25-right. We can also observe the influence of increasing the weight in Fig. 26 where each path is displayed superimposed on its respective action map. For a small weight $w_1 = 1$, the path is not smoothed, as shown in Fig. 26-left. For a weight $w_2 = 20$, leading to the inequality $|\kappa_2| \leqslant 0.75$, the path is smooth. Differences appear also in the sets of points visited during propagations: it is smaller with weight $w_1 = 1$. It means that propagation is quicker for small weights. It propagates in every directions for a higher weight (see Fig. 26-right), because the tune of w smoothes the image, as it reduces the upper-bound on curvature magnitude in Eq. (A.1).

For the virtual endoscopy application, the centering potential relies on the segmentation step described on page 3. Path sensitivity to the noise in the data is not important during this step, and we take $w \ll 1$ in order to extract a set of voxels which is a rough segmentation of our tubular object.

The path extraction is finally done using a synthetic potential representing a function of the distance to the object shape, where initial noise has disappeared. Therefore, taking w as small as possible will not lead to a path that oscillates inside the virtual fly through.

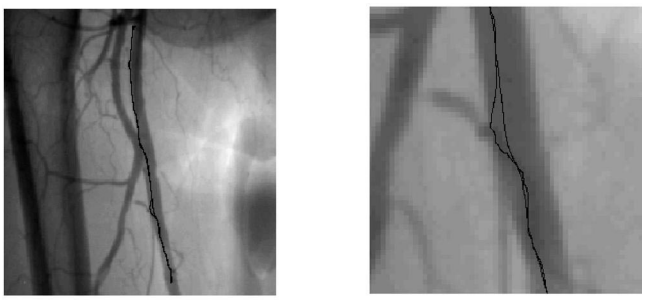


Fig. 25. Smoothing the minimal path with the weight w: the paths with $w_1 = 1$ and $w_2 = 20$.

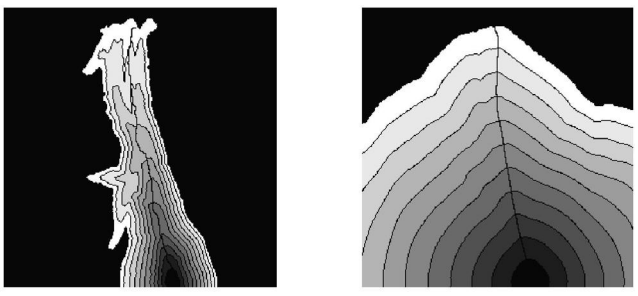


Fig. 26. Smoothing the minimal path with the weight w: the action maps.

References

Adalsteinsson, D., Sethian, J.A., 1995. A fast level set method for propagating interfaces. Journal of Computational Physics 118, 269–277.

Barbaresco, F., Monnier, B., 2000. Minimal geodesics bundles by active contours: radar application for computation of most threathening trajectories areas and corridors. In: Proceedings of European Signal and Image Processing Conference, EUSIPCO'00, Tampere, Finland.

Blum, H., 1967. A transformation for extracting new descriptors of shape. Models for the Perception of Speech and Visual Forms, MIT Press, Amsterdam, The Netherlands, pp. 362–380.

Caselles, V., Kimmel, R., Sapiro, G., 1997. Geodesic active contours. International Journal of Computer Vision 22 (1), 61–79.

Cohen, L.D., 1991. On active contour models and balloons. Computer Vision, Graphics, and Image Processing: Image Understanding 53 (2), 211–218.

Cohen, L.D., Kimmel, R., 1996. Fast marching the global minimum of active contours. In: International Conference on Image Processing, ICIP'96, Lausanne, Switzerland, Vol. 1, pp. 473–476.

Cohen, L.D., Kimmel, R., 1997. Global minimum for active contour models: A minimal path approach. International Journal of Computer Vision 24 (1), 57–78.

Cuisenaire, O., 1999. Distance transformations: fast algorithm and applications to medical image processing. Ph.D thesis, Université Catholique de Louvain, Belgium.

Deschamps, T., Cohen, L.D., 2000. Minimal paths in 3D images and application to virtual endoscopy. In: European Conference on Computer Vision, ECCV'00, Dublin, Ireland.

Deschamps, T., Ebeid, S.M., Cohen, L.D., 1999. Image Processing Method, System and Apparatus for Processing an Image representing a tubular structure and for constructing a path related to said structure. Patent Pending.

Dijkstra, E.W., 1959. A note on two problems in connection with graphs. Numerische Mathematic 1, 269–271.

Hong, L., Muraki, S., Kaufman, A., Bartz, D., Taosong, H., 1997. Virtual voyage: interactive navigation in the human colon. In: Proceedings of 24th International Conference on Computer Graphics and Interactive Techniques, pp. 27–34.

Jolesz, F.A., Loresen, W.E., Shinmoto, H., Atsumi, H., Nakajima, S., Kavanaugh, P., Saiviroonporn, P., Seltzer, S.E., Silverman, S.G., Phillips, M., Kikinis, R., 1997. Interactive virtual endoscopy. American Journal of Radiology 169, 1229–1237.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: Active contour models. International Journal of Computer Vision 1 (4), 321–331.

Kimmel, R., Sethian, J.A., 1998. Computing geodesic paths on manifolds. Proceedings of National Academy of Sciences 15, 8431–8435.

Kimmel, R., Amir, A., Bruckstein, A., 1995. Finding shortest paths on surfaces using level sets propagation. IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (1), 635–640.

Malladi, R., Sethian, J.A., 1998. A real-time algorithm for medical shape recovery. In: International Conference on Computer Vision, ICCV'98, pp. 304–310.

Malladi, R., Sethian, J.A., Vemuri, B.C., 1995. Shape modeling with front propagation: A level set approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (2), 158–175.

Merlet, N., Zerubia, J., Hogda, K.A., Braathen, B., Heia, K., 1993. A curvature-dependent energy function for detecting lines in satellite images. Proceedings of Scandinavian Conference on Image Analysis 17 (1), 699–706.

Paik, D.S., Beaulieu, C.F., Jeffrey, R.B., Rubin, G.D., Napel, S., 1998. Automated flight path planning for virtual endoscopy. Medical Physics 25 (5), 629–637.

Sethian, J.A., 1996. A fast marching level set method for monotonically advancing fronts. Proceedings of the Natural Academy of Sciences 93, 1591–1595.

Sethian, J.A., 1999. Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences. Cambridge University Press, 1999.

Tek, H., Kimia, B., 1995. Image segmentation by reaction-diffusion bubbles. In: International Conference on Computer Vision, ICCV'95, Cambridge, USA, pp. 156–162.

Tek, H., Kimia, B., 2001. Boundary smoothing via symmetry transforms. To appear in: Journal of Mathematical Imaging and Vision, Special issue on Mathematics and Image Analysis MIA'00, Paris.

Yeorong, G., Stelts, D.R., Jie, W., Vining, D.J., 1999. Computing the centerline of a colon: a robust and efficient method based on 3D skeletons. Journal of Computer-Assisted Tomography 23 (5), 786–794.