

Chemins Minimaux et Modèles Déformables en Analyse d'Images

Laurent D. COHEN

CEREMADE, UMR 7534, Université Paris-Dauphine

75775 Paris cedex 16, France; Email: cohen@ceremade.dauphine.fr

Résumé

Nous présentons une synthèse d'une partie de nos travaux sur les chemins minimaux. Introduits au départ pour trouver le minimum global de l'énergie pour les contours actifs à l'aide du Fast Marching [16], nous les avons utilisés par la suite pour la recherche de contours multiples pour compléter des points ou des courbes dans des images 2D et 3D. Plusieurs variantes permettent d'améliorer le temps de calcul, de simplifier l'initialisation ou de centrer le chemin dans une structure tubulaire. Le Fast Marching est aussi un moyen efficace de résoudre l'évolution d'un modèle de contour actif ballon par level sets. Nous montrons des applications, pour la plupart médicales, notamment pour la segmentation de vaisseaux et l'endoscopie virtuelle.

Mots Clef: Chemins minimaux, contours actifs, modèles déformables, fast marching, Equation Eikonale, Ensembles de niveaux, minimisation d'énergie, Groupement perceptuel, imagerie médicale.

Abstract

We present an overview of part of our work on minimal paths. Introduced first in order to find the global minimum of active contours' energy using Fast Marching [16], we have then used minimal paths for finding multiple contours for contour completion from points or curves in 2D or 3D images. Some variations allow to decrease computation time, make easier initialization and centering a path in a tubular structure. Fast Marching is also an efficient way to solve balloon model evolution using level sets. We show applications, mostly on medical images, like for vessel segmentation and virtual endoscopy.

Keywords: Minimal paths, active contours, deformable models, fast marching, Eikonal Equation, level sets, weighted distance, reconstruction, energy minimization, Perceptual grouping, salient curve detection, 2D and 3D medical imaging.

1 Introduction

Les contours actifs ont été l'objet d'innombrables études et variations depuis leur introduction par les auteurs de [26]. La plupart des modèles introduits depuis ont cherché à résoudre les principaux défauts de la méthode originale: *initialisation, minimisation et changements de topologie*. Une initialisation manuelle proche du contour recherché est nécessaire.

Le modèle de ballon [7] permet d'être moins exigeant pour l'initialisation. La même propriété est obtenue pour les modèles géométriques par ensembles de niveau [4, 30, 5]. Bien que l'effet de lissage des contours actifs peut surmonter quelques défauts des données, des points de contour erronés générés par le bruit ou dans une image complexe peuvent arrêter l'évolution de la courbe dans un minimum local de l'énergie. La force de gonflage [7] peut aider à éviter ces minima locaux. Une approche basée régions [12, 8] permet aussi d'être moins sensible aux minima locaux. Elle considère une énergie mixte avec un terme de type contour actif sur les bords et une contrainte d'homogénéité à l'intérieur de la région. Par ailleurs, si on a des connaissances a priori, leur intégration dans un modèle paramétrique permet d'être plus robustes [3, 2, 6, 9, 39, 34].

Néanmoins pour des images comme celles de la figure 4, l'initialisation doit être une approximation polygonale assez précise. Pour trouver les minima globaux pour les contours actifs, les auteurs de [16] ont utilisé des chemins minimaux, introduits auparavant dans [28, 27] dans un cadre différent. L'objectif était d'éviter de trouver des minima locaux, et de simplifier l'étape d'initialisation classique et fastidieuse des snakes [8, 7], en la remplaçant par la localisation des deux extrémités du contour recherché. Le schéma numérique a l'avantage d'être consistant (voir [16]) et efficace, à l'aide de l'algorithme du *Fast-Marching* introduit dans [36].

Cet article contient diverses améliorations de la méthode originale, valables aussi bien en 2D qu'en 3D. Les problèmes auxquels nous nous sommes intéressés pour la segmentation et l'extraction de contours, la recherche de trajectoires et le groupement perceptuel, sont les suivants:

- Trouver un chemin minimal entre deux points : la solution proposée dans [16] à l'aide de Fast Marching est rappelée dans la section 2.
- Trouver les chemins minimaux pour un ensemble donné de paires de points est une application simple du cas précédent.
- Pour trouver les chemins minimaux à partir d'un ensemble non structuré de points, nous proposons une méthode pour déterminer les paires de points à relier [10] (Section 3).

- Trouver les chemins minimaux pour un ensemble de points clés non connus au départ et à déterminer dans l'image à partir d'un ensemble plus large de points admissibles [11].
- Pour trouver les chemins minimaux à partir d'un ensemble non structuré de composantes connexes, nous étendons la méthode précédente pour déterminer les paires de régions à relier [15, 14] (Section 4).
- Segmentation d'une structure tubulaire en 2D et en 3D [19, 21] (sections 5 et 6).
- Détermination d'un chemin centré à l'intérieur d'une structure tubulaire et application à l'endoscopie virtuelle [18, 19] (section 7).

2 Chemins Minimaux

2.1 Optique géométrique

Afin de comprendre le principe de Fermat qui est l'interprétation physique des chemins minimaux décrits plus loin, nous illustrons la propagation de la lumière dans deux cas très simples.

Selon le principe de *Fermat*, le chemin suivi par de la lumière monochromatique pour aller d'un point p_0 à un point p_1 est celui qui prend le moins de temps. Dans le cas d'un milieu homogène, la vitesse de la lumière est constante, et donc le chemin suivi est la ligne droite, le temps de parcours étant proportionnel à la distance, comme on le voit sur la figure 1 en haut à gauche. Les ensemble de points de l'image atteints à un instant donné sont des cercles.

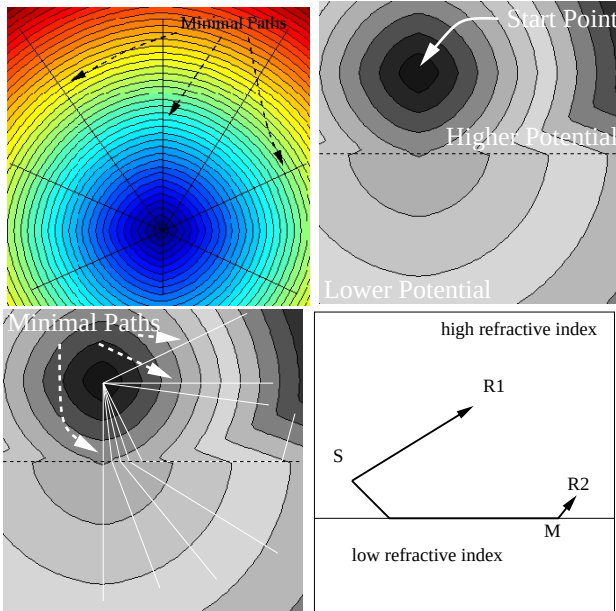


Figure 1: Fonction de coût par propagation de front et chemins minimaux pour un potentiel à une ou deux valeurs. Voir le texte.

Considérons maintenant un milieu non homogène, formé de deux régions homogènes séparées par une ligne horizontale vers le centre, comme dans la figure 1 en haut à droite. La vitesse de la lumière étant plus grande pour le rectangle du bas, la trajectoire va “préférer” rester autant que possible dans cette partie inférieure. Cela provoque l’effet de réfraction, comme nous le voyons sur différentes trajectoires à partir du point de départ dans la figure 1 en bas à gauche. Les angles des deux segments avec la normale à la ligne de séparation entre les deux milieux obéissent à la loi de Snell-Descartes (rapport des sinus égal au rapport des indices de réfraction). L’indice de réfraction $n > 1$ est le rapport entre la vitesse de la lumière dans le vide c et la vitesse dans le milieu considéré v . De la définition de cet indice, le temps de parcours T entre deux points est l’intégrale le long du chemin suivi de l’inverse de la vitesse $\frac{1}{v} = \frac{n}{c}$. Le chemin parcouru est donc celui qui minimise: $T = \frac{1}{c} \int_{p_0}^{p_1} n ds$. L’équation Eikonale a été obtenue pour cette minimisation par Hamilton, cas particulier des équations de Hamilton-Jacobi.

L’une des trajectoires, reprise sur la figure 1 en bas à droite montre l’effet bien connu du mirage. La source de lumière S est visible des points R_1 et R_2 . Mais le chemin suivi entre S et R_2 n’est pas une ligne droite, du fait que la lumière “préfère” passer par une zone d’indice plus faible pour aller plus vite. Ce phénomène est courant lorsque les variations de température sont suffisamment grandes, faisant croire à la vision d’une oasis dans le désert.

La similarité sera évidente dans les sections qui suivent où le potentiel P des contours actifs joue le rôle de l’indice de réfraction.

2.2 Minimum Global pour les contours actifs

Dans ces sections sont décrites les idées de base de la méthode introduite dans [16] pour trouver le minimum global de l’énergie d’un contour actif à l’aide des chemins minimaux. L’énergie à minimiser est similaire à celle des modèles déformables classiques ([26]), et elle combine des termes de lissage et des termes d’attraction dans des zones de l’image (Potentiel P):

$$E(C) = \int_{\Omega} \left\{ w_1 \|C'(s)\|^2 + w_2 \|C''(s)\|^2 + P(C(s)) \right\} ds \quad (1)$$

où $C(s)$ représente une courbe dans une image 2D et Ω est son domaine de définition. La méthode introduite dans [16] améliore le processus de minimisation de l’énergie car le problème devient la recherche d’un minimum global.

2.3 Formulation du problème

Comme dans [16], nous sommes amenés à minimiser

$$E(C) = \int_{\Omega=[0,L]} \{w + P(C(s))\} ds, \quad (2)$$

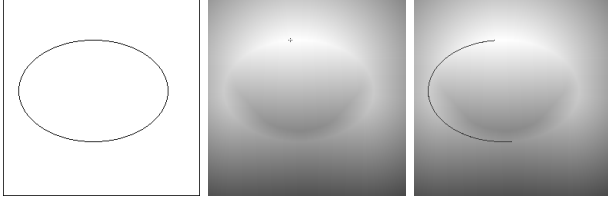


Figure 2: Trouver le chemin minimal entre deux points. A gauche: le potentiel est minimal sur l'ellipse. Au milieu: l'action minimale ou distance pondérée au point de départ. A droite: chemin minimal par rétro-propagation, à partir du second point.

où s est l'abscisse curviligne ($\|C'(s)\| = 1$). La régularisation de ce modèle est maintenant faite par la constante $w > 0$ (voir [16] pour les détails). Etant donné un potentiel $P \geq 0$, l'énergie est comme une distance pondérée par $\tilde{P} = P + w$. Les auteurs de [16] ont relié ce problème avec la formulation de type contour actifs par ensembles de niveau [36]. En particulier, sa formulation d'Euler est équivalente à celle des contours actifs géodésiques[5].

L'action minimale $\mathcal{U}$ est définie comme l'énergie minimale intégrée le long d'un chemin entre un point de départ p_0 et n'importe quel point p par:

$$\mathcal{U}(p) = \inf_{\mathcal{A}_{p_0,p}} E(C) = \inf_{\mathcal{A}_{p_0,p}} \left\{ \int_{\Omega} \tilde{P}(C(s)) ds \right\} \quad (3)$$

où $\mathcal{A}_{p_0,p}$ est l'ensemble des chemins admissibles entre p_0 et p . Le chemin minimal entre p_0 et n'importe quel point p_1 de l'image peut être facilement déduit du calcul de l'action, par une simple descente de gradient sur $\mathcal{U}$ depuis p_1 jusqu'à p_0 .



Figure 3: Minimum Global d'un contour actif par propagation de front. Le contour minimal entre les deux points donnés à gauche est trouvé sur l'image du centre. A droite on voit la fonction de coût à partir du point de départ. Remarquer la propagation plus rapide le long des routes.

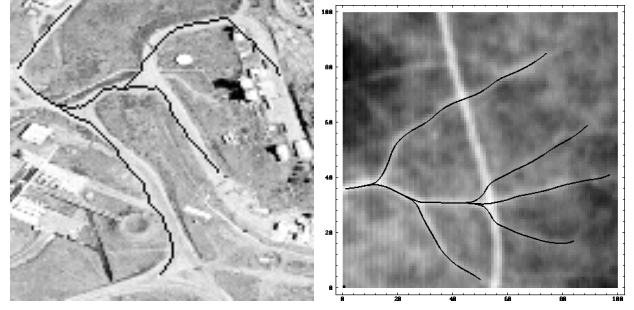


Figure 4: Plusieurs chemins minimaux sont obtenus à partir d'un même point de départ et de plusieurs points d'arrivée. Cela permet d'extraire l'ensemble des routes à gauche et l'ensemble des vaisseaux dans l'image de fond de l'oeil à droite.

2.4 Résolution du Fast Marching

Afin de calculer $\mathcal{U}$, une équation de propagation de front liée à l'équation (3) est résolue:

$$\frac{\partial C}{\partial t} = \frac{1}{\tilde{P}} \vec{\nabla} \cdot \vec{\nabla} C. \quad (4)$$

Cette équation fait évoluer un front, en partant d'un cercle infinitésimal autour de p_0 jusqu'à ce que chaque point de l'image soit visité et se voit donné une valeur pour $\mathcal{U}$. La valeur de $\mathcal{U}(p)$ est le temps t auquel le front passe par p .

La technique du *Fast Marching*, introduite dans [36], a été utilisée dans [16], se basant sur le fait que $\mathcal{U}$ satisfait l'équation *Eikonale*:

$$\|\nabla \mathcal{U}\| = \tilde{P} \quad \text{et} \quad \mathcal{U}(p_0) = 0. \quad (5)$$

La manière d'arriver à cette équation sera expliquée en section 6. Les schémas de différences finies classiques ont tendance à diverger et sont instables pour cette équation. Un schéma décentré amont, proposé dans [36], permet de stabiliser le schéma numérique en ne dérivant que d'un seul côté:

$$(\max\{u - \mathcal{U}_{i-1,j}, u - \mathcal{U}_{i+1,j}, 0\})^2 + (\max\{u - \mathcal{U}_{i,j-1}, u - \mathcal{U}_{i,j+1}, 0\})^2 = \tilde{P}_{i,j}^2, \quad (6)$$

et donnant la solution de viscosité correcte u pour $\mathcal{U}_{i,j}$. L'amélioration apportée par le *Fast Marching* est d'introduire un ordre de sélection des points de la grille. Cet ordre est basé sur le fait que l'information se propage vers l'extérieur dans une seule direction, car l'action peut seulement augmenter du fait de l'équation quadratique (6).

Cette technique de ne considérer à chaque pas que l'ensemble nécessaire de points de la grille fut originellement introduit pour la construction de chemins

Algorithme du Fast Marching en 2D

- Définitions:
 - points *Alive*: points où la valeur de l'action minimale $\mathcal{U}$ est déterminée et ne changera pas;
 - points *Trial*: prochains points de la grille à être examinés (voisins en 4-connexité). une estimation U de $\mathcal{U}$ a été calculée en ces points à l'aide de l'équation (6) seulement à partir de points *Alive*;
 - points *Far*: tous les autres points de la grille, et qui n'ont pas d'estimation de U ;
- Initialisation:
 - points *Alive*: le point de départ $p_0, \mathcal{U}(p_0) = \mathcal{U}(p_0) = 0$;
 - points *Trial*: réduit aux 4 voisins p de p_0 avec comme valeur initiale $U(p) = \tilde{P}(p)$ ($\mathcal{U}(p) = \infty$);
 - points *Far*: tous les autres points, $\mathcal{U} = U = \infty$;
- Boucle:
 - Soit $p = (i_{min}, j_{min})$ le point *Trial* avec l'action U la plus faible;
 - on l'enlève des points *Trial* pour le mettre avec les points *Alive*;
 - Pour chaque voisin (i, j) de (i_{min}, j_{min}) :
 - * Si (i, j) est *Far*, on l'ajoute à l'ensemble *Trial*;
 - * Si (i, j) est *Trial*, on met à jour $U_{i,j}$ avec l'équation (6).

Table 1: Algorithme du *Fast Marching*

de longueur minimale dans un graphe entre deux noeuds du graphe [22]. L'algorithme est détaillé dans la Table 1. Des exemples sont donnés sur les Figures 2 à 4. Le *Fast Marching* sélectionne à chaque itération le point *Trial* avec la valeur d'action minimum. Pour calculer cette valeur, on doit résoudre l'équation (6) pour chaque point *Trial*, comme précisé en section 2.5.

2.5 Algorithme du schéma 2D décentré amont

Remarquons que pour résoudre l'équation (6), seuls les points *Alive* sont considérés. Considérant les voisins du point (i, j) en 4-connexité, on note $\{A_1, A_2\}$ et $\{B_1, B_2\}$ les deux couples de voisins opposés tels que nous puissions les ordonner comme $\mathcal{U}(A_1) \leq \mathcal{U}(A_2)$, $\mathcal{U}(B_1) \leq \mathcal{U}(B_2)$, et $\mathcal{U}(A_1) \leq \mathcal{U}(B_1)$. Considérant que nous avons $u \geq \mathcal{U}(B_1) \geq \mathcal{U}(A_1)$, on en dérive l'équation

$$(u - \mathcal{U}(A_1))^2 + (u - \mathcal{U}(B_1))^2 = \tilde{P}_{i,j}^2 \quad (7)$$

Basé sur le test du discriminant Δ de l'équation (7), un ou deux voisins servent à la résoudre:

1. Si $\tilde{P}_{i,j} > \mathcal{U}(B_1) - \mathcal{U}(A_1)$, le solution de l'équation (7) est

$$u = \frac{\mathcal{U}(B_1) + \mathcal{U}(A_1) + \sqrt{2\tilde{P}_{i,j}^2 - (\mathcal{U}(B_1) - \mathcal{U}(A_1))^2}}{2}.$$

2. sinon $u = \mathcal{U}(A_1) + \tilde{P}_{i,j}$.

2.6 Chemins minimaux en 3D

Une extension 3D du *Fast Marching* a été présentée dans [18]. De manière similaire, l'action minimale $\mathcal{U}$ est définie par

$$\mathcal{U}(p) = \inf_{\mathcal{A}_{p_0,p}} \left\{ \int_{\Omega} \tilde{P}(C(s)) ds \right\} \quad (8)$$

où $\mathcal{A}_{p_0,p}$ est maintenant l'ensemble des chemins 3D admissibles entre p_0 et p . Etant donné un point de départ p_0 , pour calculer $\mathcal{U}$ on commence à partir d'un front sphérique infinitésimal autour de p_0 . Le schéma numérique 2D de l'équation (6) est étendu au 3D, pour donner:

$$\begin{aligned} &(\max\{u - \mathcal{U}_{i-1,j,k}, u - \mathcal{U}_{i+1,j,k}, 0\})^2 + \\ &(\max\{u - \mathcal{U}_{i,j-1,k}, u - \mathcal{U}_{i,j+1,k}, 0\})^2 + \\ &(\max\{u - \mathcal{U}_{i,j,k-1}, u - \mathcal{U}_{i,j,k+1}, 0\})^2 = \tilde{P}_{i,j,k}^2 \end{aligned} \quad (9)$$

donnant la solution de viscosité correcte u pour $\mathcal{U}_{i,j,k}$.

Des exemples sont donnés dans la figure 23 de la section 7.

2.7 Propagation simultanée de Fronts

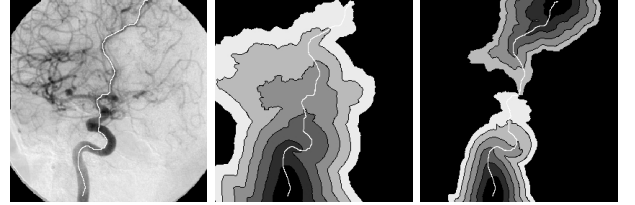


Figure 5: **Propagation simultanée:** A gauche, la donnée image, utilisée comme potentiel pour extraire un vaisseau. Au milieu, la carte d'action calculée depuis le premier point jusqu'à atteindre le second. A droite, l'action obtenue par propagation simultanée depuis les deux points, et les deux chemins depuis le point d'intersection.

L'idée est de propager simultanément un front depuis chacune des extrémités p_0 et p_1 [18]. Considérons le premier point p où ces fronts se rencontrent. Ce point est forcément sur le chemin minimal joignant p_0 à p_1 . Du fait que l'action ne peut que croître pendant la propagation, on peut l'arrêter à cette étape. En joignant les deux chemins minimaux obtenus, respectivement entre p_0 et p , et entre p_1 et p , cela donne une approximation du chemin minimal entre p_0 et p_1 . L'algorithme est résumé dans la table 2. Cette approche permet un calcul parallèle des deux propagations. Par ailleurs, la région de l'image à couvrir par

Algorithme

- Calcul des cartes d'action minimale U_0 et U_1 depuis respectivement p_0 et p_1 jusqu'à avoir un point *Alive* p_2 en commun;
- Calcul du chemin minimal entre p_0 et p_2 par back-propagation sur U_0 depuis p_2 ;
- Calcul du chemin minimal entre p_1 et p_2 par back-propagation sur U_1 depuis p_2 ;
- Joindre les deux chemins obtenus.

Table 2: Chemin minimal par intersection de deux cartes d'action.

le Fast Marching est diminuée de manière importante. La Figure 5 montre un exemple.

2.8 Calcul simultané de la longueur

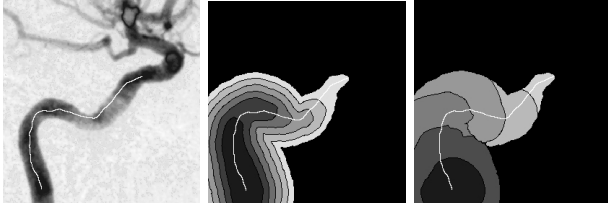


Figure 6: **Calcul simultané de la longueur du chemin.** A gauche, le potentiel; Au milieu, la carte d'action minimale; à droite la longueur du chemin minimal. Ces deux cartes sont calculées seulement jusqu'au premier point atteignant la longueur voulue.

Nous avons vu la capacité des méthodes de propagations de fronts à extraire un chemin entre deux points. Dans certains cas, comme par exemple pour donner manuellement des points dans une image 3D, il est préférable de n'avoir à donner qu'un point de départ et de déterminer le second point automatiquement. Nous décrivons ici une méthode qui construit un chemin à partir d'un point de départ et d'une longueur de chemin à atteindre. Nous pouvons calculer simultanément en chaque point parcouru par le front l'énergie U du chemin minimal et sa longueur. On choisit alors comme seconde extrémité le premier point traversé par le front qui atteint la longueur attendue. Du fait que le front se propage plus vite le long des faibles valeurs de potentiel, les chemins intéressants sont plus longs pour une valeur donnée de U . Ainsi on arrête la propagation dès qu'un point atteint la longueur voulue et on calcule le chemin minimal depuis ce point. Lorsque le front se propage dans une structure tubulaire, tous les points pour lesquels ce critère de longueur est atteint en premier sont dans une même région de l'image, loin du point de départ et en restant dans la structure tubulaire. Cela justifie

de choisir comme seconde extrémité le premier point qui atteint cette longueur (voir la Figure 6).

3 Chemins Minimaux multiples 2D entre points p_k

3.1 Groupement Perceptuel

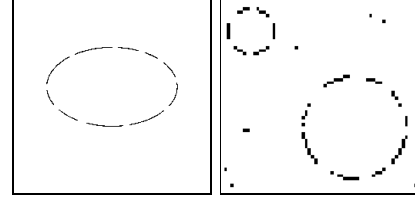


Figure 7: Exemples de contours incomplets

On s'intéresse au groupement perceptuel, et à trouver un ensemble de contours dans une image, à l'aide de courbes d'énergie minimale. Depuis leur introduction, les contours actifs [26] ont été utilisés dans de nombreux travaux pour trouver les contours d'un objet dans une image, par l'intermédiaire d'une minimisation d'énergie. Afin d'obtenir un ensemble de contours dans une image, on a besoin d'initialiser un grand nombre de contours. La formulation du problème à l'aide des contours actifs par ensembles de niveau [31, 5], permet des changements de topologie: elle permet ainsi d'obtenir une multitude de contours à partir d'un seul. Mais ces méthodes ne donnent pas de résultats satisfaisants dans notre cas, car lorsque la donnée est incomplète, les contours peuvent se propager et fuir là où la donnée est manquante, se séparant en plusieurs contours indésirables, lorsqu'on cherche à obtenir un unique contour. Ce problème relève du groupement perceptuel, où l'initialisation est un ensemble de contours à compléter. Par exemple, dans une image comme celles de la figure 7, où est représentée une forme incomplète, le système de vision humain complète facilement les données manquantes et extrait la courbe complète. Le groupement perceptuel est un problème qui a déjà fait l'objet de nombreux travaux par le passé. Il a été reformulé récemment à l'aide de méthodes de minimisation d'énergie dans [37, 25, 40]. Ces méthodes associent une mesure de représentativité à chaque composante d'une courbe, ou à chaque point de l'image. La définition d'une mesure de représentativité est alors basée indirectement sur une énergie de régularisation du second ordre du chemin contenant un de ces points, comme pour les snakes ([26]). Dans ce cas, les courbes résultantes sont obtenues dans un deuxième temps comme des lignes de crête de la mesure de représentativité, après seuillage. C'est cette relation étroite entre les courbes d'énergie minimale, comme les snakes, et la complétion de contours qui nous

amène ici à vouloir détecter un ensemble de contours pour faire la complétion sur une image, en les définissant comme un ensemble de courbes d'énergies minimales.

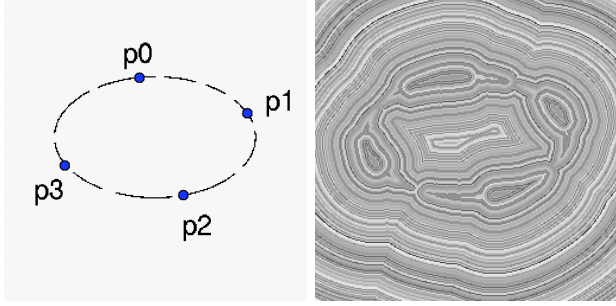


Figure 8: A gauche, le potentiel est une ellipse incomplète et les p_k sont donnés. A droite, les ensembles de niveau de l'action minimale U à partir des p_k .

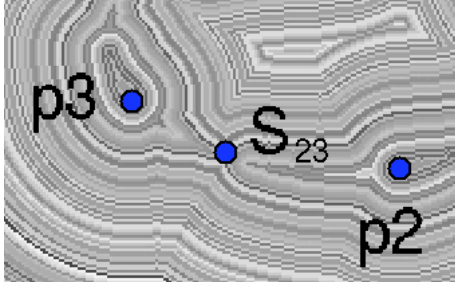


Figure 9: Zoom sur un *point selle*.

3.2 Chemins Minimaux multiples

Nous proposons d'utiliser cette approche de chemins minimaux dans le but d'extraire un ensemble de contours à partir d'un ensemble de points dans une image. Afin de trouver l'ensemble des contours les plus représentatifs dans l'image, nous cherchons les chemins minimaux entre couples de composantes connexes. Nous décrivons ici brièvement la méthode lorsque les points p_k sont déjà connus. Une méthode de détermination automatique des points p_k les plus représentatifs parmi un ensemble plus large de points admissibles est présentée dans [11, 10]. Nous supposons ici que les points p_k sont connus. Si on savait aussi quelles paires de points doivent être reliées parmi les p_k , trouver tous les contours est une application simple de la section 2. Cela serait similaire à la méthode de [24] qui utilisait elle une approche de programmation dynamique pour relier les points successifs donnés par l'utilisateur. Le problème qui nous intéresse ici est de trouver aussi quelles sont les paires de points à relier. Du fait que l'ensemble de points p_k est supposé non structuré, on ne sait pas à l'avance

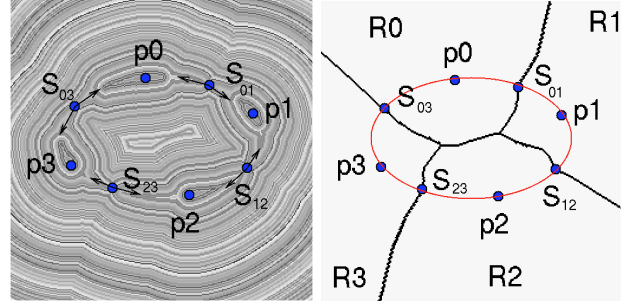


Figure 10: Ellipse. A gauche, les *points selle* sont trouvés, et une retro-propagation depuis chacun d'eux vers ses deux points sources donne les chemins à droite, ainsi que le diagramme de Voronoi.

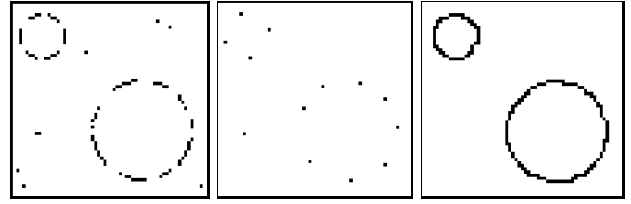


Figure 11: Deux cercles: de gauche à droite, l'ensemble des points de donnée, incomplet et bruité; les points déterminés comme extrémités; et les chemins minimaux obtenus entre eux.

quels sont les points à relier. C'est le problème que nous résolvons à l'aide d'une carte d'action minimale.

Le but principal de notre méthode est d'obtenir tous les chemins significatifs qui relient les points. Chaque point ne doit pas être connecté à tous les autres points, mais uniquement à ceux qui sont proches, au sens de l'énergie. Il y a plusieurs possibilités pour décider quels points doivent être reliés, qui dépendent du type de données, et d'application. Dans certains cas, il faut détecter les courbes fermées et éviter de créer des embranchements. Le critère choisi est alors de contraindre un point p_k à être relié avec au plus deux points p_k , dans le but de créer des courbes fermées. Si on cherche des arborescences, le critère est différent, comme en section 4.2

Pour le groupement perceptuel, le potentiel P à minimiser le long des chemins est souvent une image de points de bords, formant des contours incomplets, comme sur la figure 7. Ces points de bords sont représentés par une image binaire, avec des faibles valeurs de potentiels le long des bords, et des fortes valeurs dans le fond de l'image. Ce potentiel peut être défini comme un potentiel d'attraction à un ensemble de points (voir [13]). Le potentiel peut aussi être défini comme une fonction du gradient de l'image elle-même, $P = g(\|\nabla I\|)$, comme pour les contours actifs classiques. Il peut aussi être une image de niveaux de

Chemin Minimaux entre les points p_k

- Initialisation:
 - Les p_k sont donnés.
 - $\forall k, V(p_k) = 0; l(p_k) = k$; est l'indice de front, p_k alive.
 - $\forall p \notin \{p_k\}, V(p) = \infty; l(p) = -1$; p est far sauf les voisins en 4-connexité des p_k qui sont *trial* avec l'estimation U obtenue par (7).
- Boucle pour le calcul de $V = \mathcal{U}_{\{p_k, 0 \leq k \leq N\}}$:
 - Soit $p = (i_{min}, j_{min})$ le point *Trial* avec l'action U la plus faible;
 - On le déplace de l'ensemble *Trial* vers l'ensemble des points *Alive* avec $V(p) = U(p)$;
 - on met à jour $l(p)$ avec le même indice que le point A_1 dans l'équation (6). Si $l(A_1) \neq l(B_1)$ et que nous sommes dans le cas 1 de la section 2.5 où deux points sont utilisés et s'il s'agit de la première fois que les régions des labels $l(A_1)$ et $l(B_1)$ se rencontrent, $S(p_{l(A_1)}, p_{l(B_1)}) = p$ est défini comme le *point selle* entre $p_{l(A_1)}$ et $p_{l(B_1)}$. Si ces points n'ont pas encore deux voisins reliés, ils sont alors définis comme *voisins reliés* et $S(p_{l(A_1)}, p_{l(B_1)}) = p$ est sélectionné,
 - Pour chaque voisin (i, j) de (i_{min}, j_{min}) :
 - * Si (i, j) est *Far*, on l'ajoute aux points *Trial*;
 - * Si (i, j) est *Trial*, on met à jour l'action $U_{i,j}$.
- On obtient tous les chemins entre les *voisins reliés* sélectionnés par rétro-propagation de chaque coté de leurs *points selle*.

Table 3: Algorithme de la Section 3

gris comme dans [16], ou une fonction plus complexe des niveaux de gris. Dans tous les cas, il doit être positif.

3.3 Idées principales de la méthode

Notre approche est similaire à calculer la carte de distance à un ensemble de points, et leur diagramme de Voronoi. Dans notre cas, nous utilisons une distance pondérée, définie à travers un potentiel P . Cette distance est obtenue comme la carte d'action minimale, vis-à-vis de P , avec des valeurs d'action nulles à tous les points p_k . Au lieu de calculer une carte d'action minimale pour chaque paire de points, comme dans la section 2.3, on a juste besoin de calculer une carte d'action minimale pour trouver tous les chemins. En même temps que la carte d'action est calculée, on détermine les paires de points qui doivent être reliés. Cette méthode est basée sur la détection des points de rencontre des fronts de propagation. Il s'agit des *points selle* de l'action minimale $\mathcal{U}$. Dans la section 2.3, on montrait que le calcul de l'action minimale peut être vu comme la propagation d'un front à l'aide

de l'équation (4).

Comme l'action minimale est calculée avec le *Fast-Marching*, les ensembles de niveaux de $\mathcal{U}$ donnent l'évolution du front. Pendant l'exécution du *Fast-Marching*, la frontière de l'ensemble des points *Alive* donne la position du front.

Dans la section 2.3, nous avions un seul front évoluant à partir du point de départ p_0 . Comme tous les points p_k sont donnés avec $\mathcal{U}(p_k) = 0$, on a maintenant un front évoluant à partir de chaque point de départ p_k . Par la suite, on désignera par rencontre de fronts aussi bien la position du point où les fronts provenant de deux points p_k différents se rencontrent, que le premier point *Alive* dans l'algorithme discret qui connecte deux composantes venant de points p_k différents (voir Figures 8 et 9).

Ce problème est lié à l'approche présentée à la fin de [16] pour trouver un contour fermé. A partir d'un seul point, le second point était déterminé comme un point selle. Ce point était le lieu de rencontre des fronts se propageant de chaque coté du contour.

On utilise ici le fait qu'étant donnés deux points p_1 et p_2 , le point selle S où les deux fronts partant de chaque point se rencontrent peut être utilisé pour trouver les chemins minimaux entre p_1 et p_2 . Ceci est analogue à ce qui est présenté en section 2.7. En fait, le chemin minimal entre deux points doit passer par le point de rencontre S . Ce point est le milieu (en terme d'énergie) du chemin minimal entre p_1 et p_2 . Les rétro-propagation de S vers p_1 et de S vers p_2 donnent les deux moitiés du chemin.

Les figures 8 à 10 illustrent les étapes de l'algorithme. La figure 11 montre le résultat avec des points p_k trouvés automatiquement. Plus de détails sont donnés dans [11, 10].

4 Chemins Minimaux multiples entre régions R_k

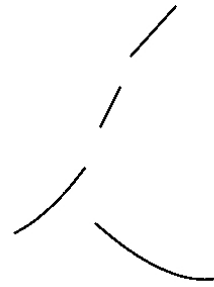


Figure 12: Exemple de régions connexes à relier

On considère le problème du groupement perceptuel et de la complétion de contours, lorsque la donnée est un ensemble non structuré de régions dans

une image. Nous présentons notre méthode pour trouver des courbes complètes à partir d'un ensemble de points sur ces contours [15, 14]. Les contours sont définis comme des chemins minimaux reliant des composantes connexes que l'on calcule à l'aide de l'algorithme du *fast marching*, jusqu'à ce que l'ensemble de ces régions soient connectées. La méthode est une extension à des régions R_k des notions précédentes pour les points p_k de la section 3.

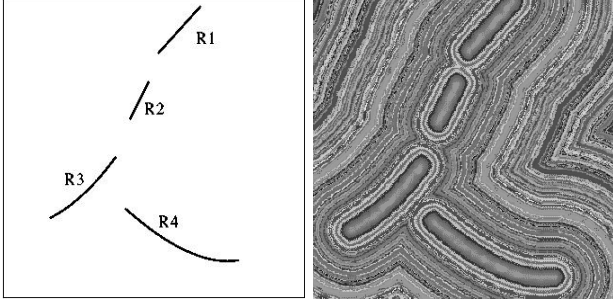


Figure 13: Ensembles de niveaux de la carte d'Action Minimale pour les 4 régions de la figure 12.

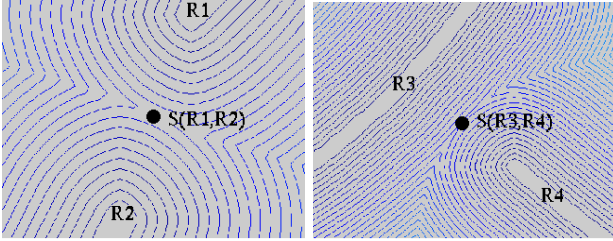


Figure 14: Zoom sur les *points selle* entre les regions.

4.1 Chemin Minimal entre deux Régions

La méthode de [16], détaillée plus haut, permet de trouver un chemin minimal entre deux points. C'est une extension facile de définir un chemin minimal entre deux régions d'une image. Prenons deux régions connexes d'une image, R_0 et R_1 , et considérons R_0 comme la région de départ et R_1 comme un ensemble de points d'arrivée. Le problème est donc de trouver un chemin qui minimise une énergie sur l'ensemble des chemins qui ont un point de départ dans R_0 et un point d'arrivée dans R_1 . L'action minimale est maintenant définie par

$$\mathcal{U}(p) = \inf_{\mathcal{A}_{R_0,p}} E(C) = \inf_{p_0 \in R_0} \inf_{\mathcal{A}_{p_0,p}} E(C) \quad (10)$$

où $\mathcal{A}_{R_0,p}$ est l'ensemble des chemins commençant avec un point de R_0 et se terminant en p . L'action minimale

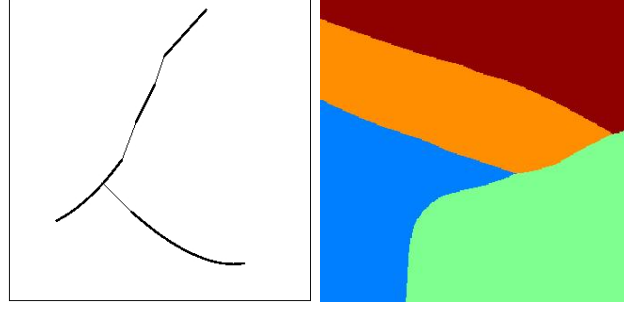


Figure 15: A gauche, on voit les chemins minimaux obtenus par rétro-propagation à partir des trois *points selle* de chacune des 4 régions d'où vient le front; A droite, le diagramme de Voronoi généré.

peut être calculée de la même manière que ci-dessus dans la table 1, avec l'ensemble *Alive* initialisé par l'ensemble des points de R_0 , avec $\mathcal{U} = 0$ et l'ensemble *Trial* étant l'ensemble des voisins en 4-connexité des points de R_0 qui ne sont pas dans R_0 . La rétro-propagation par descente de gradient sur $\mathcal{U}$ à partir de n'importe quel point p de l'image donnera le chemin minimal qui joint ce point à la région R_0 .

De manière à trouver le chemin minimal entre les régions R_1 et R_0 , on détermine un point $p_1 \in R_1$ tel que $\mathcal{U}(p_1) = \min_{p \in R_1} \mathcal{U}(p)$. Ensuite, on rétro-propage de p_1 vers R_0 pour trouver le chemin minimal entre p_1 et R_0 , qui est aussi chemin minimal entre R_1 et R_0 .

4.2 Chemins minimaux entre un ensemble de composantes connexes

Nous sommes maintenant intéressés par trouver une multitude de contours dans une image. Supposons que nous ayons un ensemble initial de contours, provenant d'une étape préliminaire de détection. On note R_k les différentes composantes connexes de ces contours.

On se propose de trouver les contours comme l'ensemble des chemins minimaux qui relient des paires de régions parmi les R_k . Si nous savons quelles paires de régions doivent être reliées entre elles, trouver l'ensemble des contours serait une application triviale de la section précédente. Le problème qui nous intéresse ici est aussi de trouver quelles paires de régions doivent être connectées entre elles. Comme l'ensemble des contours R_k est donné de manière non structurée, on ne sait pas par avance comment les régions sont connectées. C'est le problème principal que l'on s'efforcera de résoudre à l'aide des chemins minimaux. Deux régions différentes parmi les R_k seront dites *régions connectées*, si elles sont sélectionnées pour être reliées entre elles.

De plus, dans notre contexte, nous sommes intéressés par détecter des structures arborescentes,

Chemin Minimaux entre les Régions R_k

- Initialisation:
 - Étant donnés les R_k
 - $\forall k, \forall p \in R_k, V(p) = 0; l(p) = k$ est l'indice du front ; p est *alive*.
 - $\forall p \notin \cup_k R_k, V(p) = \infty; l(p) = -1$; p est *far* excepté les voisins en 4-connexité de R_k qui sont *Trial* avec U estimé à l'aide de l'équation (7).
- Boucle pour le calcul de $V = \mathcal{U}_{\cup_k R_k}$:
 - Soit $p = (i_{min}, j_{min})$ le point *Trial* avec l'action U la plus faible;
 - On l'enlève des points *Trial* pour le mettre dans les points *Alive* avec $V(p) = U(p)$;
 - on met à jour $l(p)$ avec le même indice que le point A_1 dans l'équation (6). Si $R(A_1) \neq R(B_1)$ et que nous sommes dans le cas 1 de la section 2.5 où deux points sont utilisés et s'il s'agit de la première fois que les régions des labels $l(A_1)$ et $l(B_1)$ se rencontrent, $S(R_{l(A_1)}, R_{l(B_1)}) = p$ est défini comme le *point selle* entre $R_{l(A_1)}$ et $R_{l(B_1)}$. Si ajouter un lien entre ces régions ne crée pas de *cycle*, elles sont considérées comme *régions connectées* et $S(R_{l(A_1)}, R_{l(B_1)}) = p$ est sélectionné, Pour chaque voisin (i, j) de (i_{min}, j_{min}) :
 - * Si (i, j) est *Far*, on l'ajoute aux points *Trial*;
 - * Si (i, j) est *Trial*, on met à jour l'action $U_{i,j}$.
- On extrait tous les chemins entre les *régions connectées* sélectionnées par rétro-propagation de chaque coté de leurs *points selle*.

Table 4: Algorithme de la Section 4

et nous voulons éviter les *cycles*. Un *cycle* est une séquence de régions différentes $R_k, 1 \leq k \leq K$, telles que pour $1 \leq k \leq K - 1$, R_k et R_{k+1} sont des *régions connectées* et R_K et R_1 sont aussi des *régions connectées*. Par conséquent, le critère pour que deux régions R_i et R_j se connectent est que leurs fronts en se rencontrant ne créent pas de *cycle*.

L'algorithme de cette section est décrit dans la table 4 et est illustré par les figures 13 à 15. La méthode pour détecter les cycles est la suivante. Étant donné un point selle trouvé pour les régions R_i et R_j , On teste s'il existe déjà un lien entre ces régions; cela revient à chercher une suite de régions distinctes $R_k, 1 \leq k \leq K$, avec $R_1 = R_i$ et $R_K = R_j$, telles que pour $1 \leq k \leq K - 1$, R_k et R_{k+1} soient des *régions connectées*. Ce genre de condition peut être facilement implémenté à l'aide d'un algorithme récursif. Si deux régions sont déjà connectées, le point où leurs fronts se rencontrent n'est pas considéré comme un candidat valable pour la rétro-propagation. L'algorithme

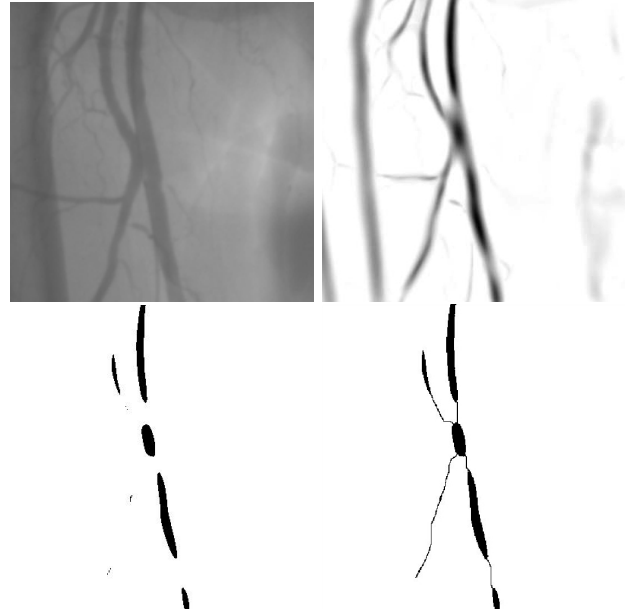


Figure 16: Image Médicale. Première ligne: image originale et potentiel pré-calculé; Deuxième ligne: à partir d'un ensemble de régions obtenues en seuillant le potentiel, notre méthode permet de relier ces régions entre elles par des chemins minimaux vis-à-vis de ce potentiel.

s'arrête automatiquement quand toutes les régions sont connectées, sans cycles.

5 Structures tubulaires 2D et 3D

En utilisant comme mesure de potentiel une fonction basée sur la réponse d'un filtre adapté à la détection de structures tubulaires, nous montrons une application de notre méthode de la section précédente à la recherche de ce type de structures dans des images médicales 2D.

Nous étendons ensuite notre méthode au 3D. La donnée est dans ce cas un ensemble de composantes connexes dans une image 3D, et on extrait les chemins minimaux qui relient entre elles ces composantes.

Elle peut aussi être utilisée quand un ensemble caractéristique de points a été extrait à l'aide d'un traitement préliminaire particulier. Par exemple, sur la figure 16, on peut voir une image 2D au niveau de la hanche sur laquelle on s'intéresse aux veines. Le potentiel est défini en se servant des idées développées dans [23] sur les filtres à base de Hessien (détaillées dans la section 5.2).

5.1 Chemins multiples 3D

On peut étendre notre approche à la recherche d'ensemble de chemins minimaux 3D entre régions dans des images tridimensionnelles. Toutes les

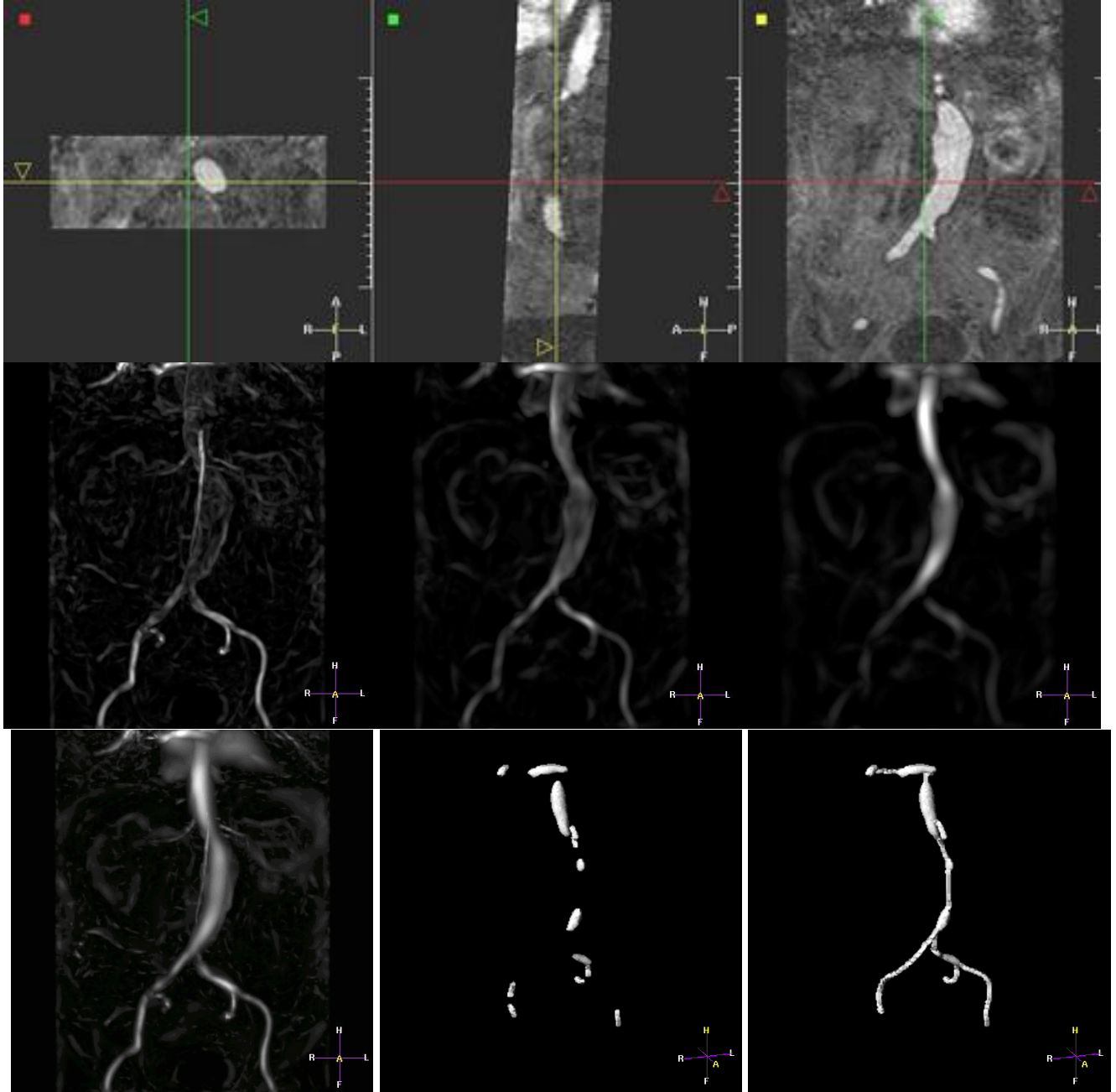


Figure 17: ligne du haut: trois coupes orthogonales de l'image 3D de l'aorte en resonance magnétique. Ligne du milieu: Détection de tube à trois échelles différentes ($\sigma = 1, 5, 10$) (visualisation des images 3D à l'aide de MIP). Ligne du bas: Groupement perceptuel dans l'aorte de la figure du haut: de gauche à droite, le potentiel 3D en MIP; la pré-détection de l'aorte; l'aorte reconstruite.

définitions et les algorithmes des sections 3 et 4 ne sont pas modifiés par ce changement de dimension du problème. La principale différence est que les algorithmes comme le *Fast-Marching* utilisent de la 6-connexité, et que les images d'action minimale ainsi que les chemins sont maintenant tridimensionnels. Nous avons brièvement présenté l'extension 3D du *Fast-Marching* en section 2.6; plus de détails sur les chemins minimaux dans les images 3D sont disponibles dans [18, 17].

5.2 Application à un cas réel: un scanner MR de l'aorte

Le problème est dans ce cas de compléter un objet partiellement pré-détecté. Sur la figure 17, on peut observer une image de résonance magnétique 3D de l'aorte qui présente une pathologie particulière: un anévrisme abdominal aortique. L'aorte est visible dans l'image grâce à l'injection d'un produit de contraste avant l'acquisition.

Nous proposons d'extraire à l'aide de notre méthode un ensemble de chemins qui donnera une représentation sous forme de squelettes de la structure arborescente de l'aorte. Notre méthode est basée sur une pré-détection d'un ensemble de régions connexes qui appartiennent à l'objet concerné.

La méthode utilisée pour extraire une information valable dans le but de construire des chemins est de calculer un filtre multi-échelle de rehaussement des veines, à partir des travaux de [23] sur ces filtres. Si on extrait les trois valeurs propres de la matrice hessienne calculée à l'échelle σ , et qu'on les ordonne $|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$, on définit une fonction de rehaussement des structures tubulaires de l'image par

$$\nu(s) = \begin{cases} 0, & \text{si } \lambda_2 \geq 0 \text{ ou } \lambda_3 \geq 0 \\ (1 - \exp \frac{-R_A^2}{2\alpha^2}) \exp \frac{-R_B^2}{2\beta^2} (1 - \exp \frac{-S^2}{2\epsilon^2}) & \text{sinon} \end{cases}$$

avec $R_A = \frac{|\lambda_2|}{|\lambda_3|}$, $R_B = \frac{|\lambda_1|}{\sqrt{|\lambda_2 \lambda_3|}}$, et $S = \sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}$. Une explication détaillée de la méthode et de son paramétrage peut être trouvée dans [23].

La réponse du filtre basé sur le Hessien est montrée à trois échelles différentes $\sigma = 1, 5, 10$ sur la figure 17. La visualisation du volume est faite à l'aide d'une image de projection MIP (Maximum Intensity Projection) ou l'on projette un volume 3D sur un plan 2D en gardant l'intensité maximale le long de la direction de projection.

En utilisant cette information calculée à plusieurs échelles, le potentiel est défini comme le maximum de la réponse du filtre sur toutes les échelles (Fig. 17 en bas à gauche). On peut facilement faire un seuillage qui va nous donner un ensemble non structuré de voxels qui appartiennent à l'aorte, comme sur la figure 17 en bas au milieu.

A partir de cet ensemble de régions, on applique l'algorithme détaillé dans la section 4. On trouve ainsi l'ensemble des chemins qui relient entre elles les régions de départ dans notre image, et qui est représenté sur la figure 17 en bas à droite.

6 Segmentation par Fast Marching

De nombreuses approches sont possibles pour segmenter la surface externe d'un objet à partir de points à l'intérieur. On peut utiliser par exemple un modèle de ballon [7] ou son implementation par ensembles de niveau, comme dans [31, 38]. En fait cette sorte de croissance de régions peut être résolue rapidement aussi par l'algorithme de *Fast Marching* [32]. Cela permet de faire une segmentation tout en restant dans le même contexte que les chemins minimaux. Lorsque l'on cherche l'action minimale à partir d'un point, l'algorithme fournit les catégories de points suivants:

- Les points dont l'action est gelée, appelés *Alive*;
- Les points non encore examinés par la propagation, appelés *Far*;
- Les points à l'interface entre les points *Alive* et *Far*, dont l'action n'est pas encore estimée, appelés *Trial*.

Cette dernière catégorie, à la frontière des points visités, est un contour en 2D ou une surface en 3D. Si le potentiel est beaucoup plus grand sur les contours qu'à l'intérieur, les contours forment un obstacle à la propagation du front. Dans ce cas, les points *Trial* définissent une surface qui segmente l'objet.

Considérons l'équation d'évolution d'un interface (courbe en 2D, surface en 3D) intervenant habituellement pour les méthodes de type ensemble de niveau:

$$\begin{cases} \frac{\partial C}{\partial t}(p) &= F(\kappa)\mathbf{n} \\ C(p, 0) &= C_0(p) \end{cases} \quad (11)$$

Nous supposons que la vitesse $F = \frac{1}{F} > 0$, et ainsi le front se déplace toujours vers l'extérieur, comme un ballon qui se gonfle [7], mais avec une vitesse qui n'est pas forcément constante. Une manière de caractériser la position de l'interface est de calculer le temps d'arrivée $T(\mathbf{x})$ de l'interface lorsqu'il traverse chaque point de l'image. Utilisant des propriétés classiques des *Level sets*, l'équation suivante est obtenue comme évolution de la surface $T(\mathbf{x})$:

$$\begin{aligned} T(C(\mathbf{x}, t)) = t &\Rightarrow \nabla T \cdot C_t = 1 \\ &\Rightarrow \nabla T \cdot \left(F \frac{\nabla T}{|\nabla T|} \right) = 1 \\ &\Rightarrow F \cdot |\nabla T| = 1 \end{aligned} \quad (12)$$

où l'on retrouve l'équation Eikonale déjà vue plus haut en section 2.4. Cette équation a aussi été utilisée pour la segmentation de surface du fait qu'elle a

les mêmes avantages que la formulation par ensembles de niveau. Dans [32], le Fast Marching permet d'obtenir une initialisation rapide pour une segmentation par Level Sets avec la fonction de vitesse $F(\mathbf{x}) = e^{-\alpha|\nabla I_\sigma(\mathbf{x})|}$, $\alpha > 0$, où le front ralentit lorsqu'il est en présence de forts gradients. Cette expression est utilisée à son tour dans l'équation Eikonale:

$$|\nabla T(\mathbf{x})| = \frac{1}{F(\mathbf{x})}. \quad (13)$$

Cette équation est résolue à l'aide du Fast Marching en 3D (voir section 2.6). Des exemples pour la segmentation du cortex et du colon sont donnés dans les figures 18 et 19, [19, 17].

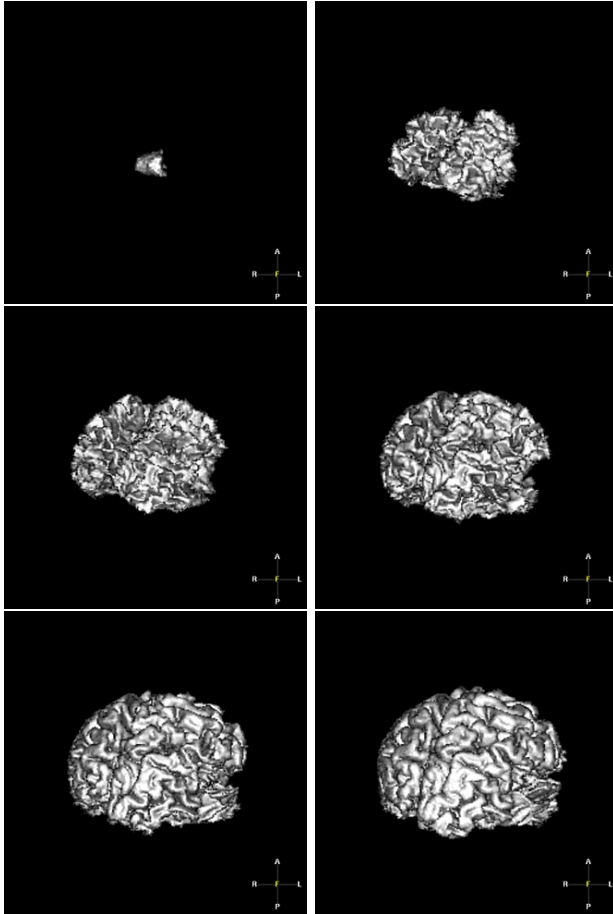


Figure 18: Segmentation sur une image de cerveau par Fast Marching .

7 Chemins Minimaux 3D centrés et endoscopie virtuelle

7.1 Chemin centré pour objet tubulaire

Un chemin minimal minimise l'intégrale du potentiel de l'équation (2). Si le potentiel est constant

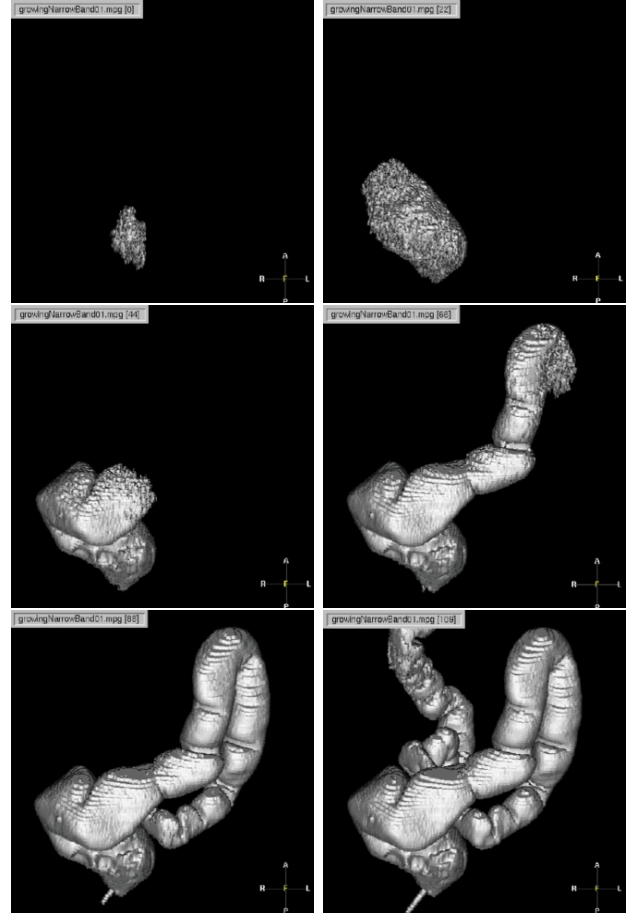


Figure 19: Propagation à l'intérieur du colon par Fast Marching.

dans certaines zones, comme l'intérieur d'un objet tubulaire, cela donnera un chemin qui minimise la longueur, c'est à dire une ligne droite. Ainsi, le chemin minimal peut souvent raser les parois, et cette propriété est gênante lorsque nous cherchons une trajectoire pour de l'endoscopie virtuelle [19]. Un chemin centré serait plus adapté, comme illustré sur la figure 20. La méthode que nous avons proposée pour obtenir un chemin centré est la suivante [20]:

- Segmentation : Le premier objectif est de segmenter, même de manière approchée la région tubulaire.
- Chemin centré : une fois la région segmentée, on cherche un chemin minimal qui est le plus loin possible des parois, à l'aide d'une fonction distance.

7.2 Description de la méthode

La méthode est décrite dans la figure 21.

1. Segmentation: La première étape est de calculer la distance pondérée à partir d'un point de

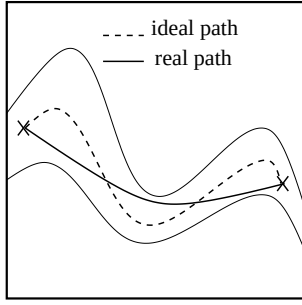


Figure 20: Extraction de chemin dans une structure tubulaire.

- départ. Le point d'arrivée peut être déterminé de manière automatique par le critère de longueur de la section 2.8.
2. Segmentation: La segmentation est obtenue comme l'ensemble des points *trial*, comme décrit dans la section 6.
3. Potentiel centré: Calcul de la carte de distance $\mathcal{E}$ à la surface obtenue à l'étape précédente (obtenue par fast marching avec $P = 1$).
4. Chemin centré: c'est le chemin minimal entre les deux extrémités avec le potentiel P_1 défini dans [19] à partir de la distance précédente. Le front se déplace maintenant plus vite à l'intérieur de l'objet.
5. Chemin centré: L'étape finale est la retro-propagation à partir du point d'arrivée.

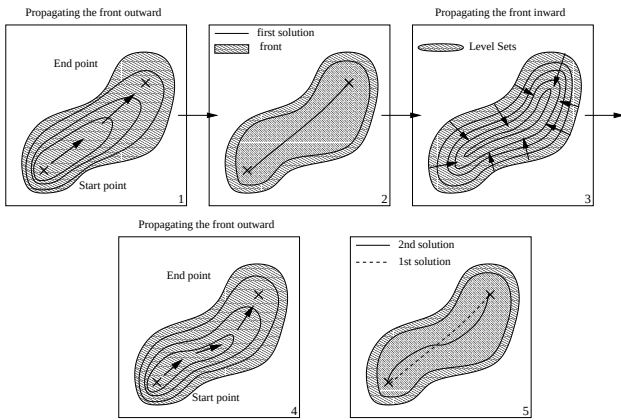


Figure 21: Les 5 étapes pour obtenir un chemin centré. Voir texte.

La figure 22 compare le résultat du chemin par une extraction classique et celle du chemin centré sur une image de vaisseaux du cerveau. La figure 23 montre des exemples de chemins minimaux

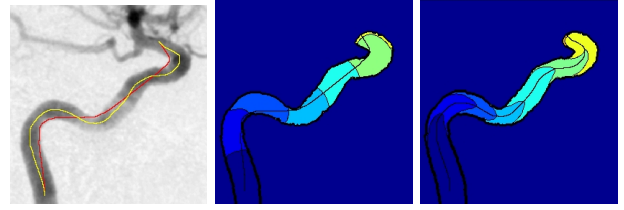


Figure 22: **Chemin centré sur un vaisseau:** A gauche, les deux chemins superposés à l'image originale. Au milieu, propagation et chemin pour un potentiel classique. A droite, propagation et chemin centré pour le potentiel de distance obtenu.

centrés obtenus sur des images de colon et de vaisseaux du cerveau. Ces chemins sont utilisés comme trajectoire d'une caméra virtuelle pour des images de synthèse (comme sur la figure 24) menant à une endoscopie virtuelle. Des vidéos sont disponibles sur <http://www.ceremade.dauphine.fr/~cohen/MPEG>

Remerciements

Nous avons présenté divers aspects des méthodes de chemins minimaux et leurs applications, notamment en imagerie médicale. Une grande partie des travaux présentés ont été effectués en collaboration avec Ron Kimmel ou à travers la direction de thèse de Thomas Deschamps que je remercie à cette occasion.

References

- [1] Andres Almensa and L.D. Cohen. Fingerprint image matching by minimization of a thin-plate energy using a two-step algorithm with auxiliary variables. In *Proc. IEEE Workshop on Applications of Computer Vision (WACV'00)*, Palm Springs, California, December 2000. IEEE.
- [2] E. Bardinet, Laurent D. Cohen, and N. Ayache. Tracking and motion analysis of the left ventricle with deformable superquadrics. *MEDIA, Medical Image Analysis, an international journal of Computer Vision, Visualisation and Image Guided Intervention in Medicine*, 1(2):129–149, November 1996. Comes with a video in the CD version of the journal.
- [3] E. Bardinet, Laurent D. Cohen, and N. Ayache. A parametric deformable model to fit unstructured 3D data. *International Journal on Computer Vision and Image Understanding*, 71(1):39–54, July 1998.
- [4] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.
- [5] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.
- [6] Isaac Cohen and Laurent D. Cohen. A hybrid hyperquadric model for 2-D and 3-D data fitting. *Computer Vision and Image Understanding*, 63(3):527–541, May 1996.
- [7] Laurent D. Cohen. On active contour models and balloons. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 53(2):211–218, March 1991.

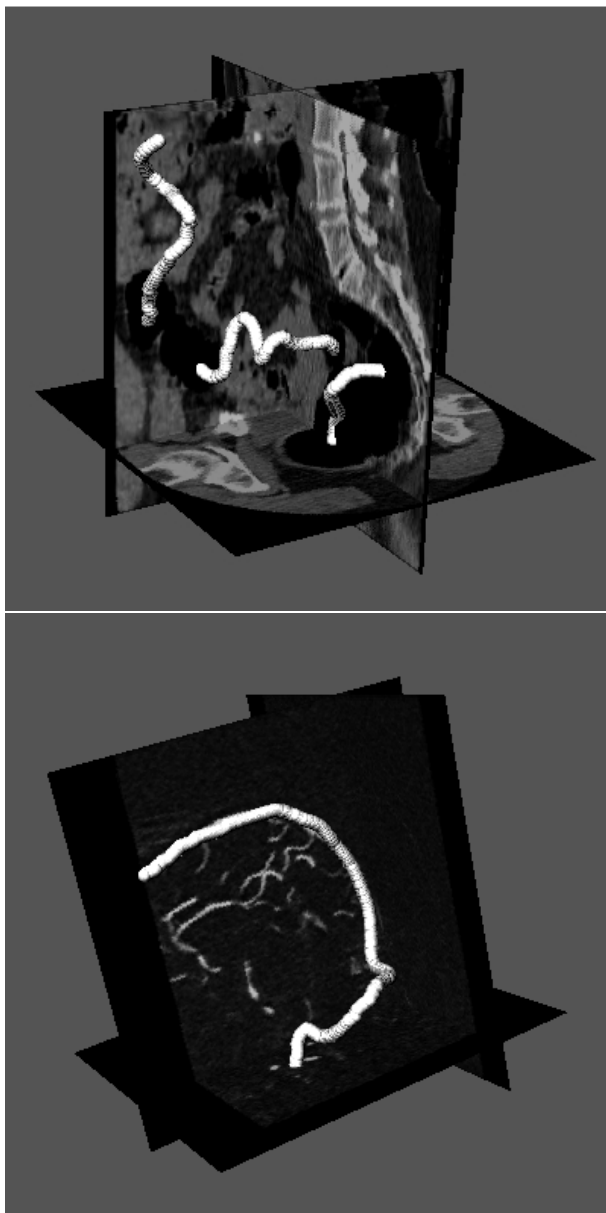


Figure 23: Exemples de chemins minimaux extraits sur des images 3D de colon et de cerveau.

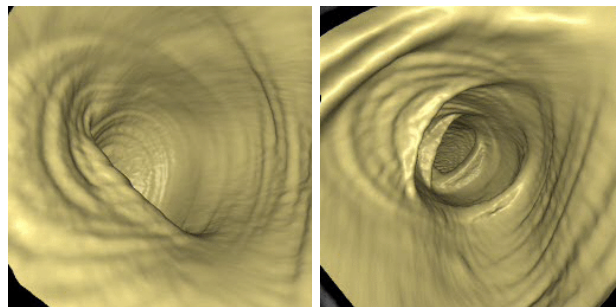


Figure 24: Endoscopie virtuelle à travers le colon (colonoscopie) à partir de l'image 3D le long du chemin minimal.

- [8] Laurent D. Cohen. Avoiding local minima for deformable curves in image analysis. In *Curves and Surfaces with Applications in CAGD*, pages 77–84. A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), 1997.
- [9] Laurent D. Cohen. Modèles déformables. Conférence plénière invitée. In *Actes de l'Ecole Thématique ISIS*, pages 1–20, Marly le Roy, April 1997.
- [10] Laurent D. Cohen. Multiple contour finding and perceptual grouping using minimal paths. In *Proc. IEEE Workshop on Variational and Level Set Methods in Computer Vision*, Vancouver, Canada, July 2001. IEEE.
- [11] Laurent D. Cohen. Multiple contour finding and perceptual grouping using minimal paths. *Journal of Mathematical Imaging and Vision*, 14(3), 2001. CEREMADE TR 0101, Jan 2001.
- [12] Laurent D. Cohen, Eric Bardinet, and Nicholas Ayache. Surface reconstruction using active contour models. In *Proceedings SPIE 93 Conference on Geometric Methods in Computer Vision*, Vol. 2031, pages 38–50, San Diego, CA, July 1993. INRIA TR 1824, December 1992.
- [13] Laurent D. Cohen and Isaac Cohen. Finite element methods for active contour models and balloons for 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-15(11):1131–1147, November 1993.
- [14] Laurent D. Cohen and Thomas Deschamps. Grouping connected components using minimal path techniques. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, Kauai, Hawaii, December 2001.
- [15] Laurent D. Cohen and Thomas Deschamps. Multiple contour finding and perceptual grouping as a set of energy minimizing paths. In Springer, editor, *Proc. Third International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR - 2001)*, 2001.
- [16] Laurent D. Cohen and R. Kimmel. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision*, 24(1):57–78, August 1997.
- [17] T. Deschamps. *Curve and Shape Extraction with Minimal Path and Level-Sets techniques - Applications to 3D Medical Imaging*. PhD thesis, Université Paris-IX Dauphine, Place du maréchal de Lattre de Tassigny, 75775 Paris Cedex, December 2001.

- [18] T. Deschamps and L.D. Cohen. Minimal paths in 3D images and application to virtual endoscopy. In *Proc. sixth European Conference on Computer Vision (ECCV'00)*, Dublin, Ireland, 26th June - 1st July 2000.
- [19] T. Deschamps and L.D. Cohen. Fast extraction of minimal paths in 3D images and applications to virtual endoscopy. *MEDIA, Medical Image Analysis, an international journal of Computer Vision, Visualisation and Image Guided Intervention in Medicine*, 5(4):281–299, December 2001. Video in the web version of the journal.
- [20] T. Deschamps, S.M. Ebeid, and L.D. Cohen. 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, March 1999.
- [21] Thomas Deschamps and Laurent D. Cohen. Grouping connected components using minimal path techniques. In Springer, editor, *Geometrical Method in Biomedical image processing*. R. Malladi (ed.), 2002.
- [22] E. W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematic*, 1:269–271, 1959.
- [23] A. Frangi, W. Niessen, K. L. Vincken, and M. A. Viergever. Multiscale vessel enhancement filtering. In *Proc. Medical Image Computing and Computer-Assisted Intervention, MICCAI'98, Cambridge*, pages 130–137, 1998.
- [24] D. Geiger, A. Gupta, L. Costa, and J. Vlontzos. Dynamic programming for detecting, tracking, and matching deformable contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(3):294–302, March 1995.
- [25] G. Guy and G. Medioni. Inferring global perceptual contours from local features. *International Journal of Computer Vision*, 20(1/2):113–133, October 1996.
- [26] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, January 1988.
- [27] R. Kimmel, A. Amir, and A. Bruckstein. Finding shortest paths on surfaces using level sets propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-17(6):635–640, June 1995.
- [28] R. Kimmel, N. Kiryati, and A. M. Bruckstein. Distance maps and weighted distance transforms. *Journal of Mathematical Imaging and Vision*, 6:223–233, May 1996. Special Issue on Topology and Geometry in Computer Vision.
- [29] M. Lefebure and L. D. Cohen. Image registration, optical flow and local rigidity. *Journal of Mathematical Imaging and Vision*, 14(2), 2001. CEREMADE TR 0102, Jan 2001.
- [30] R. Malladi, J. A. Sethian, and B. C. Vemuri. Evolutionary fronts for topology-independent shape modeling and recovery. In *Proc. Third European Conference on Computer Vision*, pages 3–13, Stockholm, Sweden, May 1994.
- [31] R. Malladi, J. A. Sethian, and B. C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. on PAMI*, 17(2):158–175, February 1995.
- [32] R. Malladi and J.A. Sethian. A real-time algorithm for medical shape recovery. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'98)*, pages 304–310, January 1998.
- [33] Frédéric Richard and Laurent D. Cohen. Une nouvelle technique de recalage d'images avec des contraintes aux bords libres: applications aux mammographies. In *Actes de la conférence Reconnaissance des Formes et Intelligence Artificielle, RFIA'02*, pages 453–462, Anger, Janvier 2002.
- [34] Patrick Sayd, Sylvie Naudet, Marc Viala, Laurent Cohen Laurent, and A. Dumont. AOMS un outil de relevé 3D d'environnements industriels. In *Actes du congrès de Vision ORASIS 2001*, Cahors, Juin 2001.
- [35] Zakaria Ben Sbeh, Laurent D. Cohen, Gérard Mimoun, and Gabriel Coscas. A new approach of geodesic reconstruction for drusen segmentation in eye fundus images. *IEEE Transactions on Medical Imaging*, December 2001.
- [36] J. A. Sethian. *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*. Cambridge Univ. Press, 1996.
- [37] A. Shaashua and S. Ullman. Structural saliency: The detection of globally salient structures using a locally connected network. In *Proc. Second IEEE International Conference on Computer Vision (ICCV'88)*, pages 321–327, December 1988.
- [38] H. Tek and B. Kimia. Image segmentation by reaction-diffusion bubbles. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'95)*, pages 156–162, Cambridge, USA, June 1995.
- [39] Samuel Vinson and Laurent D. Cohen. Extraction des bâtiments complexes à partir d'images aériennes et de MNE. In *Actes de la conférence Reconnaissance des Formes et Intelligence Artificielle, RFIA'02*, pages 125–134, Angers, Janvier 2002.
- [40] L. R. Williams and D. W. Jacobs. stochastic completion fields: a neural model of illusory contour shape and salience. In *Proc. Fifth IEEE International Conference on Computer Vision (ICCV'95)*, pages 408–415, Cambridge, USA, June 1995.

Biography

Laurent D. Cohen was born in 1962. He was student at the **Ecole Normale Supérieure**, rue d'Ulm in Paris, France from 1981 to 1985. He received the Master's and Ph.D. degrees in Applied Mathematics from University of Paris 6, France, in 1983 and 1986, respectively. From 1985 to 1987, he was member at the Computer Graphics and Image Processing group at Schlumberger Palo Alto Research, Palo Alto, California and Schlumberger Montrouge Research, Montrouge, France and remained consultant for a few years with Schlumberger afterwards. He began working with INRIA, France in 1988, mainly with the medical image understanding group Epidaure. Since 1990, he is Research Scholar (Chargé puis Directeur de Recherche) with the French National Center for Scientific Research (CNRS) in the Applied Mathematics and Image Processing group at CEREMADE, University Paris-Dauphine, Paris, France. His research interests and teaching at the university are applications of variational methods and Partial Differential Equations to Image Processing and Computer Vision, like deformable models, minimal paths, surface reconstruction, Image registration, Image segmentation and restoration.