

Contours déformables: Snakes.

Application à la segmentation d'images.

CPE Lyon
damien.rohmer@cpe.fr

17 Novembre 2010

1 Motivation

- Motivation
- Principe générale

2 Methode

- Mise en équation
- Minimisation
- Cas discret

3 Limitations et extensions

- Limitations
- Extensions possibles

Terminologie

- Contours deformables.
- Contours Actifs.
- Snakes.
- Active contours.
- Deformables Models.

1 Motivation

- Motivation

- Principe générale

2 Methode

- Mise en équation

- Minimisation

- Cas discret

3 Limitations et extensions

- Limitations

- Extensions possibles

Problématiques

- Méthodes de segmentations automatiques :
 - Histogramme
 - Régions
 - Contours

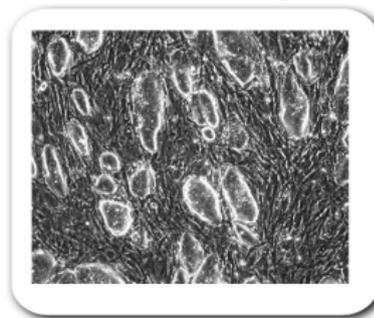
- Problèmes

- ⊖ Détails locaux
- ⊖ Image bruitée
- ⊖ Niveaux de gris non homogènes

Image de test



Vraie image



Cas d'exemple

- Même sur une image "simple"



- Constatations :
 - Plusieurs solutions possibles
 - Notre cerveau cherche à fermer les contours
 - (Il y arrive facilement)

1 Motivation

- Motivation
- Principe générale

2 Methode

- Mise en équation
- Minimisation
- Cas discret

3 Limitations et extensions

- Limitations
- Extensions possibles

Principe

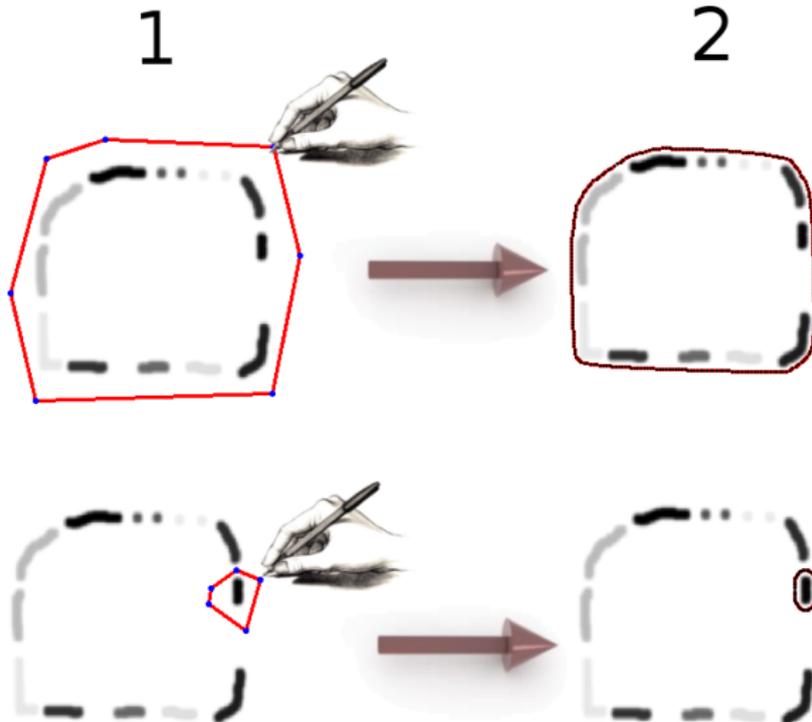
■ Constat :

- 1 Il est facile de définir une forme globale manuellement.
- 2 Longer localement les contours manuellement est fastidieux et moins aisé.

■ Idee

- 1 Intégrer des **informations a-priori** de l'utilisateur sur la **forme globale** du contour.
- 2 On ajuste cette forme automatiquement d'après l'image.

Principe



Origine

Publication Originale

[M. Kass, A. Witkin, D. Terzopoulos. **Snakes : Active Contour Models**. *International Journal of Computer Vision*, 1988]



- Depuis, beaucoup d'améliorations + généralisation en 3D.
- Classique en imagerie médicale.

Demetri Terzopoulos :

- Doctorat du MIT (1994)
- Professeur à UCLA
- Le premier à intégrer les méthodes physiques en informatique graphique.



1 Motivation

- Motivation
- Principe générale

2 Methode

- Mise en équation
- Minimisation
- Cas discret

3 Limitations et extensions

- Limitations
- Extensions possibles

1 Motivation

- Motivation
- Principe générale

2 Methode

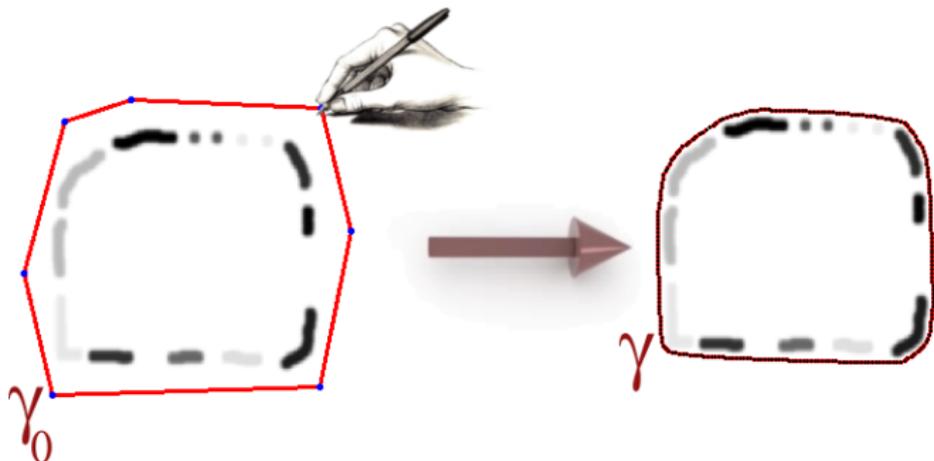
- **Mise en équation**
- Minimisation
- Cas discret

3 Limitations et extensions

- Limitations
- Extensions possibles

Principe

- On considère une courbe initiale γ_0 approximant la forme souhaitée.
- On recherche à déformer la courbe en γ tel que
 - 1 γ soit une courbe lisse.
 - 2 γ se place sur les contours de la forme.

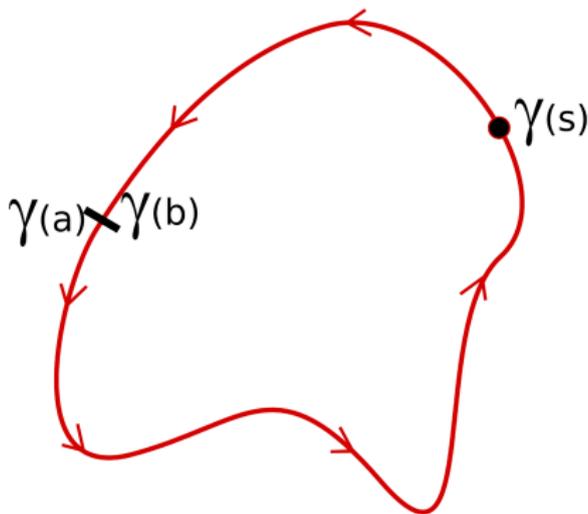


Mise en equation

- γ est une courbe 2D paramétrée par s .

$$\gamma : \begin{cases} [a, b] \subset \mathbb{R} & \rightarrow \mathcal{D} \subset \mathbb{R}^2 \\ s & \mapsto \gamma(s) = (\gamma_x(s), \gamma_y(s)) \end{cases}$$

γ est une courbe fermée : $\gamma(a) = \gamma(b)$.



Mise en Equation

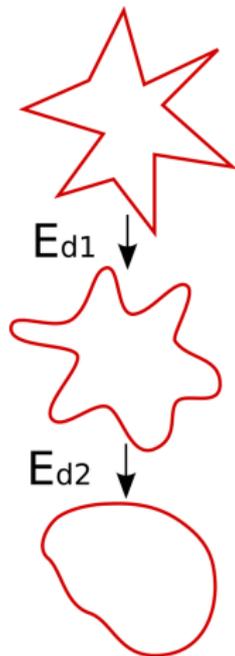
- γ est une courbe lisse
 - γ minimise ses dérivées premières et secondes.

$$E_{d1}(\gamma) = \oint \|\gamma'\|^2 = \int_{s=a}^b \left\| \frac{\partial \gamma}{\partial s}(s) \right\|^2 ds$$

$$E_{d2}(\gamma) = \oint \|\gamma''\|^2 = \int_{s=a}^b \left\| \frac{\partial^2 \gamma}{\partial s^2}(s) \right\|^2 ds$$

- Energie *interne* de la courbe :

$$E_{inter}(\gamma) = \lambda_1 E_{d1}(\gamma) + \lambda_2 E_{d2}(\gamma)$$



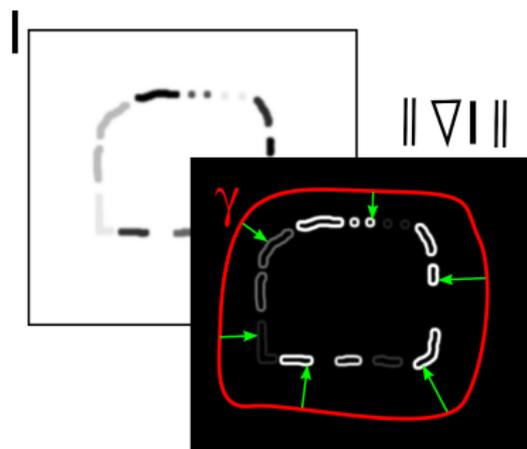
Mise en Equation

- γ se place sur les contours de la forme.
- γ passe par les maximums du gradient de l'image I .

$$E_{image}(\gamma) = - \oint \|\nabla I\|^2(\gamma) = - \int_{s=a}^b \|\nabla I\|^2(\gamma(s)) ds$$

$\Rightarrow \gamma$ minimise l'énergie externe dépendant de l'image

$$E_{exter} = \lambda_3 E_{image}$$



Mise en Equation

- $\gamma_{optimal}$ minimize la somme pondérée de ces energies

$$\gamma_{optimal} = \mathit{argmin}_{\gamma}(E)$$

$$\text{avec } E = E_{inter} + E_{exter}$$

- En rassemblant les termes

$$E(\gamma) = \oint \lambda_1 \|\gamma'\|^2 + \lambda_2 \|\gamma''\|^2 - \lambda_3 \|\nabla I\|^2(\gamma)$$

Mise en Equation

Ou plus précisément, il faut trouver $\gamma : \mathbf{s} \mapsto \gamma(\mathbf{s}) = (\gamma_x, \gamma_y)(\mathbf{s})$
tel que $E(\gamma)$ soit minimal

$$E(\gamma) = \int_{\mathbf{s}=a}^b \left[\lambda_1 \left\| \frac{\partial \gamma}{\partial \mathbf{s}} \right\|^2 (\mathbf{s}) + \lambda_2 \left\| \frac{\partial^2 \gamma}{\partial \mathbf{s}^2} \right\|^2 (\mathbf{s}) - \lambda_3 \|\nabla I\|^2(\gamma(\mathbf{s})) \right] d\mathbf{s}$$

1 Motivation

- Motivation
- Principe générale

2 Methode

- Mise en équation
- **Minimisation**
- Cas discret

3 Limitations et extensions

- Limitations
- Extensions possibles

(Rappel :) Minimisation

- En dimension finie :

- Pour trouver $\mathbf{x}_{opt} = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^N} F(\mathbf{x})$
- On sait que si $\nabla F(\mathbf{x}) = 0$ alors on est sur un extrema.
⇒ On part de \mathbf{x}^0 et on itère suivant le gradient (direction de plus grande pente)

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \Delta t \nabla F(\mathbf{x}^k)$$

(Rappel :) Minimisation

- En dimension infinie :

- Pour trouver $\gamma_{opt} = \operatorname{argmin}_{\gamma \in \mathcal{F}} E(\gamma)$, avec

$$\gamma : \begin{cases} [a, b] \subset \mathbb{R} & \rightarrow \mathbb{R} \\ s & \mapsto \gamma(s) \end{cases}$$

- On s'intéresse aux *methodes variationnelles* (calculus of variations).
 - Euler-Lagrange nous dit que pour minimiser

$$\int G(s, \gamma, \gamma', \gamma'') ds, \quad \text{il faut} \quad \frac{\partial G}{\partial \gamma} - \frac{\partial}{\partial s} \frac{\partial G}{\partial \gamma'} + \frac{\partial^2}{\partial s^2} \frac{\partial G}{\partial \gamma''} = 0$$

- Pour le résoudre, on transforme cela en équation d'évolution temporelle en considérant $\gamma(s, t)$ avec :

$$\frac{\partial \gamma}{\partial t}(s, t) = - \left(\frac{\partial G}{\partial \gamma} - \frac{\partial}{\partial s} \frac{\partial G}{\partial \gamma'} + \frac{\partial^2}{\partial s^2} \frac{\partial G}{\partial \gamma''} \right) (s, t)$$

Resolution

Equation à minimiser

$$E(\gamma) = \int \lambda_1 \|\gamma'\|^2 + \lambda_2 \|\gamma''\|^2 - \lambda_3 \|\nabla I\|^2(\gamma)$$

- On considère une courbe γ de paramètre s évoluant au cours du temps t : $\gamma(s, t) = (\gamma_x(s, t), \gamma_y(s, t))$.

Système à résoudre

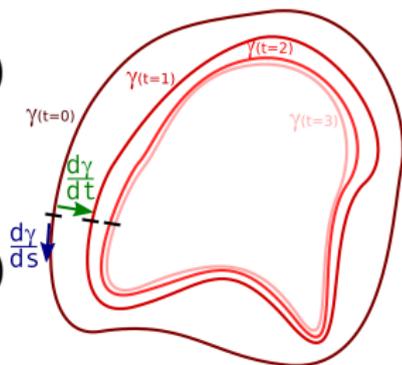
$$\begin{cases} \frac{\partial \gamma}{\partial t} = \lambda_1 \gamma'' - \lambda_2 \gamma'''' + \lambda_3 \nabla(\|\nabla I\|^2)(\gamma) \\ \gamma(t=0) = \gamma_0 \end{cases}$$

Resolution

Ou plus précisément, 2 equations scalaires sur

$$\gamma(\mathbf{s}, t) = (\gamma_x(\mathbf{s}, t), \gamma_y(\mathbf{s}, t)).$$

$$\left\{ \begin{array}{l} \frac{\partial \gamma_x}{\partial t}(\mathbf{s}, t) = \lambda_1 \frac{\partial^2 \gamma_x}{\partial \mathbf{s}^2}(\mathbf{s}, t) - \lambda_2 \frac{\partial^4 \gamma_x}{\partial \mathbf{s}^4}(\mathbf{s}, t) \\ \quad + \lambda_3 \nabla_x(\|\nabla I\|^2)(\gamma(\mathbf{s}, t)) \\ \\ \frac{\partial \gamma_y}{\partial t}(\mathbf{s}, t) = \lambda_1 \frac{\partial^2 \gamma_y}{\partial \mathbf{s}^2}(\mathbf{s}, t) - \lambda_2 \frac{\partial^4 \gamma_y}{\partial \mathbf{s}^4}(\mathbf{s}, t) \\ \quad + \lambda_3 \nabla_y(\|\nabla I\|^2)(\gamma(\mathbf{s}, t)) \end{array} \right.$$



1 Motivation

- Motivation
- Principe générale

2 Methode

- Mise en équation
- Minimisation
- **Cas discret**

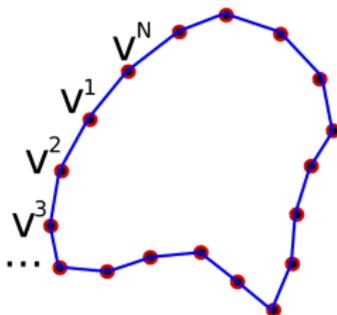
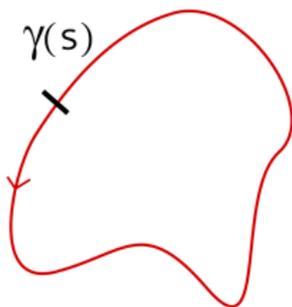
3 Limitations et extensions

- Limitations
- Extensions possibles

Discretisation spatiale

- Discretisation spatiale suivant $s_k = a + k \Delta s$ $k \in [1, N]$.

$$\gamma_{x/y}(s_k) = [v_{x/y}^1, v_{x/y}^2, \dots, v_{x/y}^N]$$



$$v_x = \begin{pmatrix} v_x^1 \\ v_x^2 \\ \vdots \\ v_x^N \end{pmatrix} \quad v_y = \begin{pmatrix} v_y^1 \\ v_y^2 \\ \vdots \\ v_y^N \end{pmatrix}$$

Discretisation spatiale

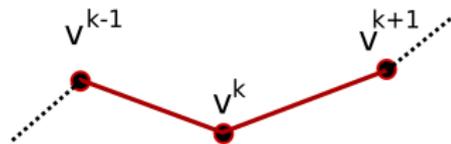
Ex. Syntaxe type Matlab

```
N=20; s=[0:N-1]/(N-1);  
vx=cos(2*pi*s);vy=sin(2*pi*s);
```

Discretisation spatiale

- Rappel : Matrice de dérivée seconde pour une courbe fermée

$$\frac{\partial^2 \gamma_x}{\partial s^2} \Rightarrow \gamma_x(s_{k+1}) - 2\gamma_x(s_k) + \gamma_x(s_{k-1})$$

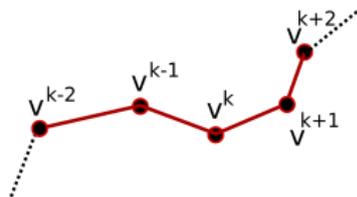


$$\Leftrightarrow \begin{pmatrix} -2 & 1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \\ 1 & 0 & 0 & 0 & \cdots & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} v_x^1 \\ v_x^2 \\ v_x^3 \\ \vdots \\ v_x^{N-1} \\ v_x^N \end{pmatrix}$$

$$\Leftrightarrow D_2 v_x$$

Discretisation spatiale

- Rappel : Matrice de dérivée 4ième pour une courbe fermée



$$\frac{\partial^4 \gamma_x}{\partial s^4} \Rightarrow \gamma_x(s_{k+2}) - 4\gamma_x(s_{k+1}) + 6\gamma_x(s_k) - 4\gamma_x(s_{k-1}) + \gamma_x(s_{k-2})$$

$$\Leftrightarrow \begin{pmatrix} 6 & -4 & 1 & 0 & \dots & 0 & 1 & -4 \\ -4 & 6 & -4 & 1 & \dots & 0 & 0 & 1 \\ 1 & -4 & 6 & -4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \dots & -4 & 6 & -4 \\ -4 & 1 & 0 & 0 & \dots & 1 & -4 & 6 \end{pmatrix} \begin{pmatrix} v_x^1 \\ v_x^2 \\ v_x^3 \\ \vdots \\ v_x^{N-1} \\ v_x^N \end{pmatrix}$$

$$\Leftrightarrow D_4 v_x$$

Discretisation spatiale

$$\left\{ \begin{array}{ll} \gamma_{x/y}(\mathbf{s}, t) & \rightarrow \mathbf{v}_{x/y} = [v_{x/y}^1, v_{x/y}^2, \dots, v_{x/y}^N] \\ \frac{\partial^2 \gamma_{x/y}}{\partial \mathbf{s}^2}(\mathbf{s}, t) & \rightarrow D_2 \mathbf{v}_{x/y} \\ \frac{\partial^4 \gamma_{x/y}}{\partial \mathbf{s}^4}(\mathbf{s}, t) & \rightarrow D_4 \mathbf{v}_{x/y} \\ \nabla(\|\nabla I\|^2)(\gamma(\mathbf{s}, t)) & \rightarrow (P_x(v_x, v_y), P_y(v_x, v_y)) \end{array} \right.$$

- Equation discrète en espace :

$$\left\{ \begin{array}{l} \frac{\partial v_x}{\partial t}(t) = (\lambda_0 D_2 - \lambda_1 D_4) v_x(t) + \lambda_3 P_x(v(t)) \\ \frac{\partial v_y}{\partial t}(t) = (\lambda_0 D_2 - \lambda_1 D_4) v_y(t) + \lambda_3 P_y(v(t)) \end{array} \right.$$

Discretisation temporelle

- Discrétisation en temps suivant $t_k = k \Delta t$.

$$v_x(t_k) = v_{x,k}$$

$$\frac{v_{x,k} - v_{x,k-1}}{\Delta t} = (\lambda_1 D_2 - \lambda_2 D_4) v_{x,k} + \lambda_3 P(v_{x,k-1}, v_{y,k-1})$$

$$\underbrace{(Id + \Delta t(-\lambda_1 D_2 + \lambda_2 D_4))}_{A} v_{x,k} = \underbrace{(v_{x,k-1} + \Delta t P(v_{x,k-1}, v_{y,k-1}))}_{b_{x,k}}$$

$$\Rightarrow v_{x,k} = A^{-1} b_{x,k} \quad \text{Système linéaire}$$

Snakes in a nutshell

Equation à minimiser

$$E(\gamma) = \oint \lambda_1 \|\gamma'\|^2 + \lambda_2 \|\gamma''\|^2 - \lambda_3 \|\nabla I(\gamma)\|^2$$

Système à résoudre

$$\frac{\partial \gamma}{\partial t} = \lambda_0 \gamma'' - \lambda_1 \gamma'''' + \lambda_3 \nabla_x (\|\nabla I(\gamma)\|^2) \quad \text{avec} \quad \gamma(t=0) = \gamma_0$$

Solution numérique

$$\begin{aligned} A &= Id + \Delta t (-\lambda_1 D_2 + \lambda_2 D_4) \\ P_k &= \nabla (\|\nabla I\|^2)(v_{k-1}) \end{aligned} \quad \begin{cases} v_{x,k} = A^{-1}(v_{x,k-1} + \Delta t P_{x,k}) \\ v_{y,k} = A^{-1}(v_{y,k-1} + \Delta t P_{y,k}) \end{cases}$$

Algorithme

```
Initialise [vx,vy]=[gamma0_x,gamma0_y];  
Construit A=Id+dt (-lambda_1 D_2+lambda_2 D_4);  
Construit [Px,Py]=gradient(norm(gradient(I)).^2);  
Precalcul A_inv=inv(A);
```

Tant que (non convergence)

```
    new_x=A_inv*(vx+dt Px(vx,vy))  
    new_y=A_inv*(vy+dt Py(vx,vy))  
    vx=new_x;vy=new_y;
```

fin tant que

```
gamma=[cx,cy];
```

Limitations

- Nécessite une initialisation manuelle.
- Nécessite d'être proche de la forme finale.
- Topologie fixe.
- Courbe se contracte uniquement.
- Ne suit pas les concavités.

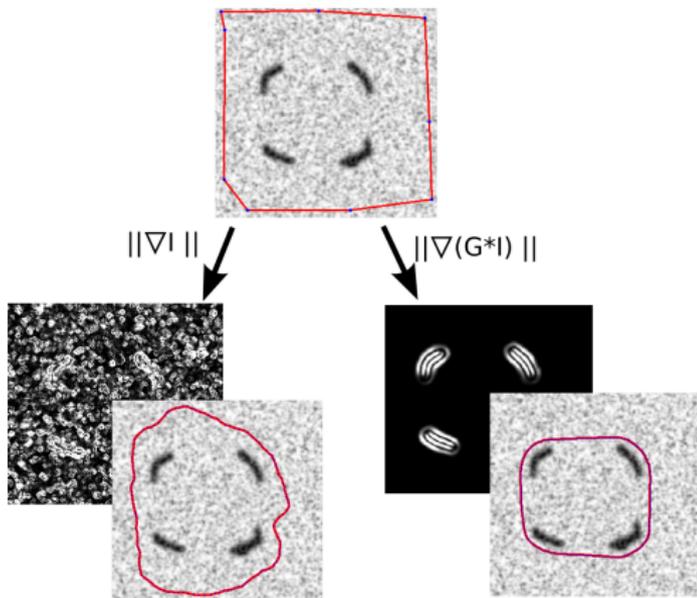
Extensions : Image bruitée

- Images bruitées \Rightarrow énergie externe modifiée.

$$\nabla I \rightarrow \nabla(G \star I)$$

avec G : noyaux Gaussien.

- Detection de contours évolués (Canny).

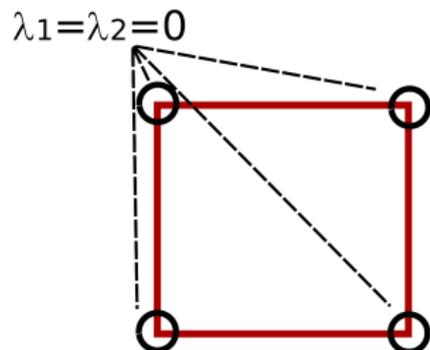


Extensions : Courbes paramétriques

- Possibilité d'utilisation de courbes paramétriques (ex. Splines, NURBS, ...) :
 - Evaluation des dérivées de γ exactes.
 - Courbes plus lisses

Extensions : Pondérations variables

- Préservation d'arêtes vives :
 $\Rightarrow \lambda_1(\mathbf{s}), \lambda_2(\mathbf{s}), \lambda_3(\mathbf{s})$ fonctions de \mathbf{s} .
- $\lambda_1(\mathbf{s}_0) = \lambda_2(\mathbf{s}_0) = 0$ sur l'arête.



Extensions : Non contraction de la courbe

- Courbe de même longueur que γ_0 .
- On considère

$$E_{d_1} = \int \|\gamma'(s) - \gamma'_0(s)\|^2 ds$$

Extensions : Contours concaves

- Courbes concaves \Rightarrow Modification de la force externe par un champs vectoriel à divergence nulle
- Gradient Vector Flow (GVF).

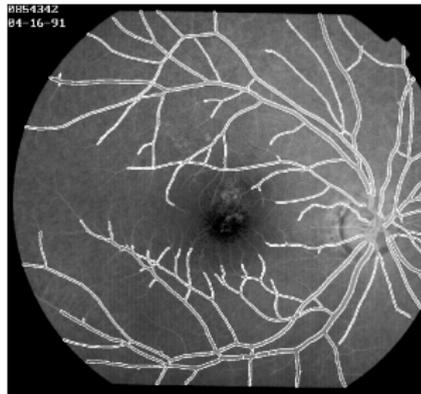
Article

[C. Xu, J. Prince. **Gradient Vector FLOW, A New External Force for Snakes.** *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997.]



Extensions : Topologie variable

- Topologie variable \Rightarrow Voir cours Level Set.



Article

[T. McInerney, D. Terzopoulos **Topologically Adaptable Snakes**. *International Conference on Computer Vision (ICCV)*, 1995.]

