

# Une méthode adaptative pour le lissage de courbes Application à la détection de contours

## An Adaptative Method for the Smoothing of Curves Edge Detection Application

par Salim DJEZIRI, Eric PETIT, Jacques LEMOINE

Laboratoire d'Etude et de Recherche en Instrumentation Signaux et Systèmes  
Université Paris-XII Val de Marne  
61, avenue du Général de Gaulle  
F-94010 Créteil cedex

### *résumé et mots clés*

La méthode de lissage de courbes discrètes présentée est fondée sur la minimisation d'un critère d'erreur quadratique appliqué sur des portions jointives de la courbe à traiter. En imposant des contraintes géométriques au niveau des points de jonctions entre intervalles, on aboutit à une formulation directe de la solution. Un paramètre unique permet de façon simple de contrôler la force du lissage qui évolue ainsi entre 2 cas extrêmes : l'interpolation et l'approximation. La méthode simule le comportement d'un scripteur cherchant à unir des points par une courbe, il peut privilégier le passage du tracé par chaque point ou au contraire respecter la forme globale définie par l'ensemble des points.

Cette méthode adaptative de lissage est utilisée comme étape finale d'un processus de segmentation d'images, le paramètre contrôlant la force du lissage étant défini à partir du gradient mesuré localement en chaque point.

Lissage de courbes, Approximation, Interpolation, Détection de contours.

### *abstract and key words*

We present a new approach to smooth discrete curves. The smoothing is realized by associating portions of regular curves which are defined on each points interval. The originality of the method consists in finding the portions of curves by minimizing the squared error over a restricted neighbourhood around each point. Adding continuity constraints at the junction points, we obtain a direct formulation of the solution. A unique parameter allows to easily control the smoothing amplitude which can be selected between two extreme cases : interpolation or approximation. It seems like a drawer behaviour trying to join points by a curve. He can choose to join each point by a curve or only take into account the global form of the set of points.

The method is particularly adapted to fit contours defined on an image and is used as a final step of image segmentation process. The parameter controlling the smoothing amplitude is computed from the value of local gradient magnitude on each pixel.

Curve fitting, Approximation, Interpolation, Contour detection.

## 1. introduction

Réaliser le lissage d'une courbe discrète consiste à déterminer une fonction continue qui en donne une représentation satisfaisante. La qualité du lissage dépend des méthodes et des contraintes utilisées pour choisir et paramétrer cette fonction. Deux métho-

dologies différentes sont principalement utilisées : l'approximation et l'interpolation.

L'**approximation** consiste à approcher un ensemble de points par une courbe. En 1963, Tihonov [Tih.63] propose un formalisme général du lissage en terme de régularisation de courbes.

Supposons qu'une courbe  $z(t)$  soit donnée par ses échantillons observables  $z(t_i) = z_i = g(t_i) + e(t_i)$  pour  $i = 1, \dots, n$  où  $g(t_i)$  représente les échantillons de la fonction idéale  $g(t)$  sur

l'intervalle  $[a, b]$  et  $e(t_i)$  est un bruit additif. La courbe  $g(t)$  est estimée par la fonction  $f(t)$  qui minimise l'expression suivante :

$$E = \frac{1}{n} \sum_{i=1}^n (f(t_i) - z_i)^2 + \lambda \sum_{r=0}^m \int_a^b w_r(t) \left( \frac{d^r f(t)}{dt^r} \right)^2 dt \quad (1)$$

où  $f(t)$  est de classe  $C^{m-1}$  sur  $[a, b]$  et  $f^{(m)} \in L^2(\mathbb{R})$  et  $w_r(t)$  est une fonction de pondération agissant sur les différents degrés de lissage.

Le premier terme de cette expression minimise une distance de la courbe de lissage à l'ensemble des points alors que le deuxième terme permet d'obtenir une courbe régulière.  $\lambda$  est un paramètre qui permet de doser l'effet relatif de chacun des deux termes.

De nombreuses méthodes, différant par le choix de la fonction  $f(t)$  et l'évaluation du second terme de l'expression (1), utilisent le formalisme de Tihonov [MuS.85, LeP.88, ChC.93, ShA.89, BoK.86].

Un cas particulier est obtenu lorsque  $\lambda$  est choisi égal à zéro. Le problème se réduit alors à la minimisation d'une erreur quadratique totale sans contrainte de régularité.

L'implantation numérique de cette méthode est aisée surtout si l'on utilise une base de fonctions orthonormées  $\phi_i$ . Néanmoins le choix des fonctions de base est critique car il suppose une connaissance *a priori* de l'allure des courbes à lisser.

L'**interpolation** consiste à faire passer une courbe régulière par un sous ensemble de points extraits de l'ensemble à lisser. Pour cela on utilise généralement des fonctions polynômiales définies sur des intervalles  $[t_i, t_{i+1}]$  qui respectent des contraintes de continuité et de dérivabilité d'ordre  $r$  aux points de jonctions ou nœuds  $t_i$  (« spline »).

Afin de réaliser une approximation de courbe par ces méthodes, les  $n$  points extraits  $P_i$  sont approchés par  $m$  points de contrôle  $V_i$  définissant une spline  $Q(t)$  telle que sa distance aux points  $P_i$  soit minimale [BBB.87, SMM.90]. La qualité du lissage dépend principalement du choix optimal des points de contrôle  $V_i$  qui doivent être distribués non uniformément de manière à être denses au voisinage des fortes courbures [GuA.92]. La détermination des points de contrôle est souvent réalisée par une approximation polygonale ou par une recherche de points dominants.

L'approximation polygonale consiste à remplacer des portions de la courbe 2D par des segments de droite afin d'obtenir un polygone comportant le minimum de points et qui approche au mieux la courbe initiale. Ramer [Ram.72] et Duda [DuH.73] ont proposé un des premiers algorithmes d'approximation polygonale en procédant par division de plus en plus fine de segments jusqu'à satisfaction d'un critère d'erreur sur chaque segment. De nombreuses équipes se sont penchées sur cet algorithme itératif de

fusion-division de segments afin d'obtenir une meilleure approximation ou des temps de traitements plus courts [PaH.74] [Dav.79] [SIG.80] [BaP.82] [KuD.82] [Rob.85] [Dun.86] [RaR.91].

On considère souvent que les points dominants d'une courbe correspondent aux régions de forte courbure. Différentes méthodes de détection de ces points ont été proposées en fonction de la définition de la courbure adoptée dans un cadre discret [RoJ.73] [RoW.75] [FrD77] [SaS.78] [WuW.93]. Teh et al. ont réalisé une étude comparative entre différentes méthodes [TeC.89].

Un des problèmes fondamentaux qui se pose lors de la segmentation d'une image est la fermeture et le lissage des contours d'un objet à partir d'un ensemble de points obtenus par des méthodes d'extraction de contours. Les méthodes d'approximation et d'interpolation citées ci-dessus ne permettent pas de prendre en compte des informations locales pouvant être soit déduites de l'image soit des connaissances *a priori* sur le contour de l'objet. Dans le présent article nous proposons une méthode de lissage fondée sur la minimisation de l'erreur quadratique calculée sur des portions jointives de la courbe à traiter. Cette méthode qui ne nécessite pas l'extraction de points dominants ni la recherche d'un optimum global permet, au moyen d'un seul paramètre  $\mu$ , d'adapter le lissage entre l'interpolation et l'approximation. Ce paramètre peut être déterminé en fonction des variations locales d'une grandeur  $\phi$  quelconque  $\mu = \Psi(\phi)$ , ce qui rend le lissage local et adaptatif. Toutefois le paramètre  $\mu$  peut être fixé globalement soit en tant que connaissance *a priori* (méthode supervisée) soit déduit par optimisation d'une fonction globale d'énergie (méthode globale non supervisée). Enfin cet algorithme permet de déterminer l'incidence des irrégularités de la courbe à lisser sur le résultat final par analogie avec certaines données de la perception visuelle [FiB.86].

Nous traitons le problème localement sur chaque intervalle  $[t_i, t_{i+1}]$ . Chaque point  $t_i$  est associé à un intervalle  $[t_i, t_{i+1}]$  et à un voisinage  $[t_i, t_{i+\mu}]$  sur lequel on cherche la meilleure fonction approchée au sens des moindres carrés. La taille  $\mu$  du voisinage  $[t_i, t_{i+\mu}]$  est variable. Des contraintes de continuités géométriques sont imposées en chaque point de manière à obtenir une courbe finale régulière par raccordement des portions de courbes définies sur chaque intervalle.

Nous étudions l'effet du paramètre  $\mu$  sur l'erreur quadratique totale et nous montrons comment contrôler aisément la force du lissage au moyen de ce paramètre. Nous introduisons un modèle physique qui nous permet de déduire une valeur du paramètre  $\mu$  par optimisation d'une fonction d'énergie.

La méthode, définie sur des courbes non fermées, est facilement étendue à des courbes fermées.

Nous utilisons cette technique de lissage afin de détecter des contours réguliers dans des images numériques. Une ligne polygonale, initialisée par un ensemble de points dans l'image, est approchée par une courbe régulière dont la force de lissage varie en fonction du gradient en chaque point.

## 2. principe

Appelons  $C$  un ensemble de  $N$  points :  $\{P(t_i, y_i)/i = 1, \dots, N$  et  $t_{i+1} > t_i\}$ .

Dans la méthode des moindres carrés, on recherche une fonction  $f(t)$ , formée linéairement à partir d'une base de fonctions indépendantes  $\phi_i(t)$  de dimension  $m + 1$ , soit :

$$f(t) = \sum_{i=0}^m a_i \phi_i(t) \quad (2)$$

$f(t)$  minimise l'erreur quadratique totale exprimée par :

$$E = \sum_{i=1}^N (y_i - f(t_i))^2 \quad (3)$$

En posant :  $Y^T = (y_1, y_2, \dots, y_N)$ ,  $A^T = (a_0, a_1, \dots, a_m)$  et

$$F = \begin{bmatrix} \phi_0(t_1) & \dots & \phi_m(t_1) \\ \vdots & \ddots & \vdots \\ \phi_0(t_N) & \dots & \phi_m(t_N) \end{bmatrix}$$

l'erreur devient :  $E = (Y - FA)^T(Y - FA)$ .

La solution minimum est obtenue lorsque la dérivée de  $E$  par rapport à  $A$  est nulle, soit :

$$\frac{\partial E}{\partial A} = -2F^T Y + 2F^T F A = 0 \quad (4)$$

soit en supposant  $N > m$  :

$$A = (F^T F)^{-1} F^T Y \quad (5)$$

Dans notre méthode, nous nous intéressons sur chaque intervalle  $[t_i, t_{i+1}]$ , à la fonction  $g_i(t) = \sum_{j=0}^m a_{ij} \phi_j(t)$  qui minimise l'erreur quadratique totale sur un voisinage  $V_i$  de cet intervalle.

L'erreur relative à l'intervalle  $[t_i, t_{i+1}]$  s'écrit :

$$E_i = \sum_{k \in V_i} (y_{i+k} - g_i(t_{i+k}))^2$$

La résolution par la méthode des moindres carrés aboutit à la détermination des composantes du vecteur  $A$  de l'équation (5). Il faut évidemment considérer des contraintes de continuité pour que les courbes définies sur deux intervalles voisins forment une courbe continue et dérivable au point de jonction des deux intervalles.

Le traitement que nous proposons débute au premier intervalle à gauche jusqu'au dernier intervalle à droite. Il recherche une fonction d'approximation  $g_i(t)$  sur un intervalle  $[t_i, t_{i+1}]$  en se basant sur  $g_{i-1}(t)$  définie sur l'intervalle précédent  $[t_{i-1}, t_i]$ .

Choisir le voisinage  $V_i$  centré autour du point courant  $t_i$  aboutit à imposer de fortes contraintes sur la fonction  $g_i(t)$ . En effet la fonction est contrainte par les conditions de continuité en  $t_i$  sur les deux demi-intervalles gauche et droit de part et d'autre de  $t_i$ . Le minimum de l'erreur quadratique  $E_i$  peut alors atteindre des valeurs importantes et faire que la courbe représentant  $g_i(t)$  s'éloigne des points à approcher.

De ce fait, nous prenons comme voisinage de minimisation, le demi-intervalle à droite du point courant, soit :  $[t_i, t_{i+\mu}]$  (figure 1). Le lissage est alors anti-causal et l'erreur quadratique à minimiser s'écrit :

$$E_i = \sum_{k=0}^{\mu} (y_{i+k} - g_i(t_{i+k}))^2 \quad (6)$$

Ainsi, soit  $g_{i-1}$  la fonction définie par  $A_{i-1}$  sur  $[t_{i-1}, t_i]$ , les contraintes de continuité au point de jonction  $t_i$  s'expriment comme suit :

$$g_{i-1}^{(k)}(t_i) = g_i^{(k)}(t_i) \quad \text{pour } k = 0, \dots, r \quad (7)$$

Les fonctions  $g_i$  et  $g_{i-1}$  ont les mêmes dérivées jusqu'à l'ordre  $r$  au point de jonction  $t_i$ . La minimisation de l'expression (6) doit tenir compte de l'égalité (7). En réécrivant l'expression de  $g_i(t)$  sous forme matricielle, on obtient :

$$g_i^{(k)}(t_i) = \Phi^{(k)}(t_i) A_i = [\phi_0^{(k)}(t_i), \phi_1^{(k)}(t_i), \dots, \phi_m^{(k)}(t_i)] A_i \quad (8)$$

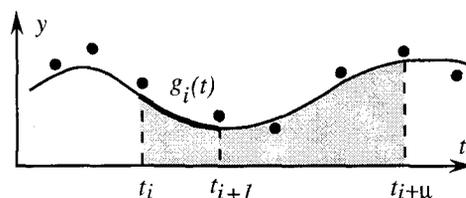


Figure 1. – Recherche de la fonction  $g_i(t)$  relative à l'intervalle  $[t_i, t_{i+1}]$  en minimisant l'erreur sur l'intervalle  $[t_i, t_{i+\mu}]$ .

Nous définissons alors la matrice des contraintes  $M_i$  comme suit :

$$M_i = \begin{bmatrix} \phi_0^{(0)}(t_i) & \phi_1^{(0)}(t_i) & \dots & \phi_m^{(0)}(t_i) \\ \vdots & \vdots & \dots & \vdots \\ \phi_0^{(r)}(t_i) & \phi_1^{(r)}(t_i) & \dots & \phi_m^{(r)}(t_i) \end{bmatrix} \quad (9)$$

L'équation (7) devient alors au point  $t_i$  :  $M_i A_i = M_i A_{i-1}$ .

Pour simplifier, posons  $M_i A_{i-1} = D_i$ .

$D_i$  est un vecteur de composantes connues car dépendant de l'intervalle précédent. L'équation matricielle des contraintes se décompose en un système à  $m + 1$  inconnues et  $r + 1$  équations. Si on se place dans le cas où  $m$  est strictement supérieur à  $r$ , on obtient  $r + 1$  composantes de  $A_i$  liées à  $m - r$  autres composantes. Nous décomposons le vecteur  $A_i$  en deux vecteurs  $B_i$  et  $C_i$ .  $B_i$

## Méthode adaptative pour le lissage de courbes

représente les  $r + 1$  composantes liées, alors que  $C_i$  représente les  $m - r$  composantes indépendantes. Ainsi on obtient :

$$A_i = \begin{bmatrix} B_i \\ C_i \end{bmatrix} \quad (10)$$

En reportant l'expression de  $A_i$  dans l'équation des contraintes et en décomposant  $M_i$  de manière à isoler le nouveau vecteur inconnu  $C_i$ , on obtient :

$$\begin{bmatrix} U_i & V_i \end{bmatrix} \begin{bmatrix} B_i \\ C_i \end{bmatrix} = D_i$$

soit :

$$U_i B_i + V_i C_i = D_i \quad (11)$$

Pour une base de fonctions orthogonales la matrice  $M_i$  est de rang  $r + 1$ , on peut donc trouver une matrice carrée  $U_i$  inversible de dimension  $r + 1$ .

Ce qui permet d'exprimer la partie liée  $B_i$  en fonction de la partie libre  $C_i$ .

$$B_i = U_i^{-1} D_i - U_i^{-1} V_i C_i \quad (12)$$

en remplaçant cette expression dans  $A_i$ , on obtient :

$$A_i = \begin{bmatrix} B_i \\ C_i \end{bmatrix} = \begin{bmatrix} U_i^{-1} D_i \\ 0 \end{bmatrix} + \begin{bmatrix} -U_i^{-1} V_i \\ I_d \end{bmatrix} C_i \quad (13)$$

qui est de la forme :  $A_i = Z_i + W_i C_i$ .

avec :

$$Z_i = \begin{bmatrix} U_i^{-1} D_i \\ 0 \end{bmatrix}, \quad W_i = \begin{bmatrix} -U_i^{-1} V_i \\ I_d \end{bmatrix} \quad (14)$$

où  $I_d$  est la matrice identité.

Posons :

$$Y_i = \begin{bmatrix} y_{i+1} \\ \vdots \\ y_{i+\mu} \end{bmatrix}, \quad F_i = \begin{bmatrix} \phi_0(t_i + 1) & \dots & \phi_m(t_i + 1) \\ \vdots & \ddots & \vdots \\ \phi_0(t_i + \mu) & \dots & \phi_m(t_i + \mu) \end{bmatrix} \quad (15)$$

$E_i$  devient :  $E_i = (Y_i - F_i A_i)^T (Y_i - F_i A_i) + (y_i - g_i)^2$

avec :  $g_i = g_i(t_i) = \Phi(t_i) A_i = g_{i-1}(t_i) = \Phi(t_i) A_{i-1}$ .

En remplaçant  $A_i$  par sa nouvelle forme et en dérivant  $E_i$  par rapport à  $C_i$ , on obtient :

$$\frac{\partial E_i}{\partial C_i} = 2(-W_i^T F_i^T Y_i + W_i^T F_i^T F_i Z_i + W_i^T F_i^T F_i W_i C_i) \quad (16)$$

L'égalité à zéro permet de trouver un minimum local, soit :

$$C_i = (W_i^T F_i^T F_i W_i)^{-1} (W_i^T F_i^T Y_i - W_i^T F_i^T F_i Z_i) \quad (17)$$

la matrice  $F_i W_i$  étant de dimension  $(\mu, m - r)$ , une condition

nécessaire pour que  $W_i^T F_i^T F_i W_i$  soit inversible est que :  $\mu \geq m - r$ .

La solution complète s'écrit alors :

$$A_i = Z_i + W_i (W_i^T F_i^T F_i W_i)^{-1} W_i^T F_i^T (Y_i - F_i Z_i)$$

soit sous forme récurrente :

$$A_i = A_{i-1} + W_i (W_i^T F_i^T F_i W_i)^{-1} W_i^T F_i^T (Y_i - F_i A_{i-1}) \quad (18)$$

L'algorithme de lissage consiste en un balayage des intervalles  $[t_i, t_{i+1}]$  de gauche à droite. A chaque étape, une remise à jour des matrices  $F_i$  et  $Y_i$  est nécessaire. On calcule également la matrice  $W_i$  à partir des valeurs de la fonction déterminée à l'étape précédente et de ses dérivées successives.

Nous faisons deux exceptions pour le premier et le dernier intervalle. Pour le premier intervalle, aucune contrainte n'étant imposée, une valeur  $A_0$  est calculée à l'aide de l'équation (5) sur l'intervalle  $[t_0, t_{\max(\mu, m)}]$ . Au voisinage du dernier intervalle, la zone de calcul de l'erreur quadratique devient inférieure à  $\mu$ . On peut utiliser plusieurs artifices pour déterminer les portions de courbes concernant les derniers points. Nous prenons comme valeur de  $\mu$  le nombre de points restants jusqu'à ce que  $\mu = m - r$ . L'algorithme est alors arrêté et la dernière fonction déterminée sert à approcher les  $m - r$  points restants.

Cet artifice confère à l'algorithme des caractéristiques importantes. Les traitements du premier et du dernier point ne sont pas identiques (effets de bord). Lors du traitement du dernier intervalle, au point courant  $t_{N-m+r}$ , l'algorithme affecte la valeur  $m - r$  à  $\mu$ . Ce qui a pour effet de faire passer la courbe par ces derniers points.

Le caractère anti-causal de l'algorithme, dû à la minimisation de l'erreur sur  $[t_i, t_{i+\mu}]$ , confère à la courbe au point  $P_i$ , la propriété de « réagir » en fonction des points d'indices supérieurs. La solution semble détecter les variations des points avant leur apparition réelle dans le sens de parcours. Cet effet peut être gênant, dans certaines applications d'analyse de formes où il s'agit de localiser avec précision les points anguleux. Pour supprimer cet inconvénient et rendre le lissage indépendant du sens de parcours, nous proposons un traitement par double balayage. Il consiste à prendre comme solution  $S(u)$ , pour une même valeur du paramètre  $\mu$ , la moyenne d'un lissage dans le sens direct  $f_D(u)$  avec un lissage dans le sens inverse  $f_I(u)$  (figure 2). Cette solution permet de diminuer les effets de bord et aboutit à un traitement symétrique.



Figure 2. – Selon que la forme est traitée dans le sens direct ou inverse, l'algorithme aboutit à des solutions différentes. La solution  $S(u)$  est indépendante du sens de traitement.

### 3. effet du paramètre $\mu$ sur l'erreur d'approximation

Cet algorithme dépend de deux choix importants :

- la base de fonctions linéairement indépendantes  $\phi_j(t)$ ;
- la dimension de la base  $m$ .

Pour mesurer l'incidence du paramètre  $\mu$  sur le lissage, nous avons étudié son influence sur l'erreur quadratique totale et l'énergie totale de courbure.

Reprenons l'écriture de  $A_i$  proposée dans l'expression (10) et considérons l'équation (6) qui définit  $E_i$ .

Si on impose à  $E_i$  d'être nulle, on obtient :

$$y_{i+k} - g_i(t_{i+k}) = 0 \quad \forall k = 0, \dots, \mu.$$

soit :

$$y_{i+k} - \Phi_i(t_{i+k}) \begin{bmatrix} B_i \\ C_i \end{bmatrix} = 0 \quad \forall k = 0, \dots, \mu$$

Ce système possède *a priori*  $\mu + 1$  équations linéaires et  $m - r$  inconnues. En constatant que la première équation (pour  $k = 0$ ) est redondante, nous distinguerons trois cas :

- $\mu < m - r$ .

Il y a plus d'inconnues que d'équations, le système admet alors une infinité de solutions. Le lissage se traduit par une interpolation. Des contraintes de régularité peuvent être ajoutées afin de résoudre le système de façon unique.

- $\mu = m - r$ .

Le système admet une solution unique. L'erreur quadratique totale est nulle. L'algorithme réalise une interpolation qui présente des oscillations (grande énergie de courbure) d'autant plus faibles que  $m - r$  est grand. Il est à noter pour  $m = 3, r = 2, \mu = 1$  et la base de fonctions polynômiales que l'on retrouve l'interpolation par spline cubique [Don.91].

- $\mu > m - r$ .

Le système a plus d'équations que d'inconnues, il n'est plus possible à l'erreur quadratique d'être nulle et l'algorithme réalise de ce fait une approximation.

Le choix du paramètre  $\mu$  obéit donc à la règle suivante :

Lorsque  $\mu = m - r$  on obtient une interpolation (erreur quadratique nulle et énergie de courbure importante).

Lorsque  $\mu > m - r$ , on obtient une approximation (l'erreur quadratique totale croît avec  $\mu$  tandis que l'énergie de courbure décroît ce qui se traduit par un lissage de plus en plus fort).

Quand  $\mu$  tend vers  $N$  (le nombre total de points à traiter), la solution se rapproche de celle des moindres carrés standard.

### 4. illustration de la méthode

Pour illustrer cette méthode, nous avons utilisé une base de fonctions indépendantes simples  $(1, t, \dots, t^m)$  et nous avons imposé la continuité jusqu'à l'ordre 1 ( $r = 1$ ). La figure 3 illustre les résultats obtenus pour une répartition quasi-sinusoïdale de points, pour différentes valeurs de  $\mu$  et deux valeurs de  $m$  (2 et 3).

Nous retrouvons, en fonction de la valeur de  $\mu$ , les deux cas que nous avons distingués théoriquement au paragraphe précédent :

- $\mu = \mu_0 = m - r$  l'interpolation obtenue présente d'autant moins d'oscillations que  $m - r$  est petit.
- $\mu > m - r$  l'approximation est d'abord très douce, c'est le cas pour  $(\mu, m) = (2, 2)$  et  $(3, 3)$  pour devenir de plus en plus forte à mesure que  $\mu$  augmente.

$\mu = \mu_\infty = N - m + r$ , ce cas particulier montre que l'algorithme tend à se rapprocher d'une approximation aux moindres carrés standard représentée sur la figure par une courbe en gras. Pour les premiers points, les deux courbes se superposent (c'est la même erreur qui est minimisée par les deux méthodes); vers la fin de la répartition, le voisinage de minimisation est au fur et à mesure tronqué, ce qui amène notre solution à se démarquer de celle des moindres carrés standard.

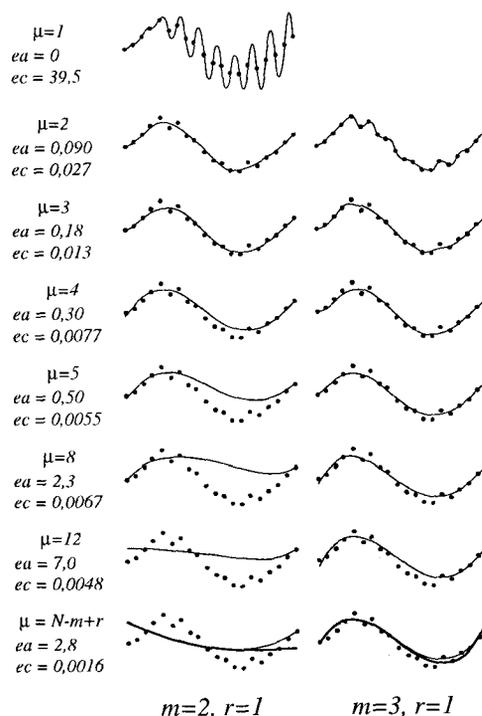


Figure 3. – Lissage d'une courbe 1D en simple balayage. Passage graduel de l'interpolation à une approximation de plus en plus forte. La courbe en gras (pour  $\mu = N - m + r$ ) représente l'approximation par moindres carrés standard.

## Méthode adaptative pour le lissage de courbes

L'application de cet algorithme pour des courbes 2D est illustrée en figure 4 pour 4 types de répartition de points. Nous avons utilisé la base de fonctions  $(1, t, t^2, t^3)$  et avons imposé la continuité géométrique à l'ordre 1 ( $r = 1$ ). L'influence du paramètre  $\mu$  sur le lissage est confirmé par l'évaluation de l'erreur quadratique et de la courbure moyenne (tableau 1).

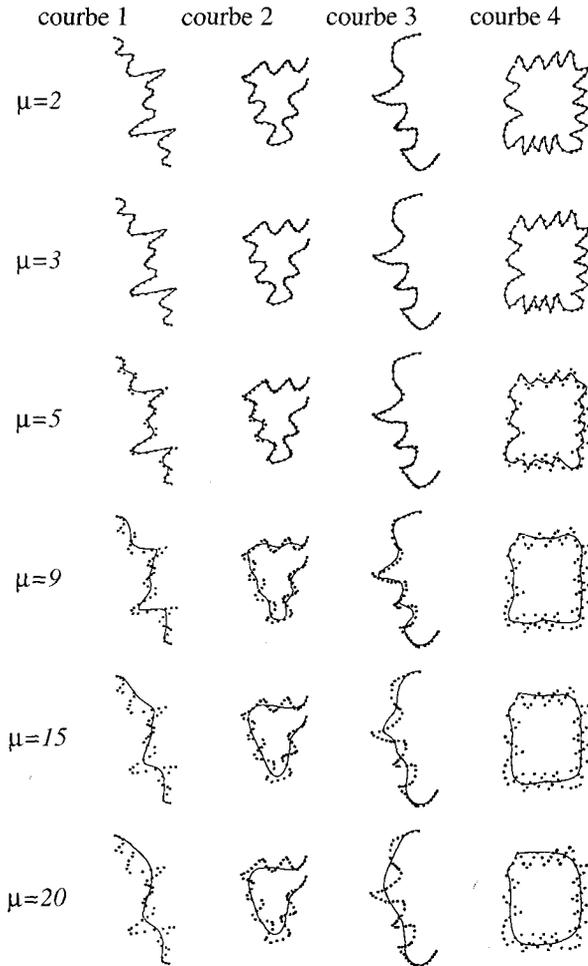


Figure 4. - Lissage d'une courbe 2D par simple balayage. Contrôle du lissage au moyen de  $\mu$  (avec  $m = 3$  et  $r = 1$ ).  
 $\mu = m - r$  : interpolation ( $\mu = 2$ ).  
 $\mu > m - r$  : approximation d'autant plus forte que  $\mu$  est grand.

Tableau 1 : Evolution de l'erreur quadratique ( $ea$ ) et de la courbure ( $ec$ ) en fonction du paramètre  $\mu$ , pour les courbes de la figure 4 ( $m = 3, r = 1$ )

	courbe 1		courbe 2		courbe 3		courbe 4	
$\mu$	$ea$	$ec$	$ea$	$ec$	$ea$	$ec$	$ea$	$ec$
2	0	160	0	111	0	146	0	396
3	7,6	71	5	50	5,4	43	7,5	154
5	45	28	42	18	22	19	46	67
9	248	8,4	175	5,8	129	7,7	292	15
15	746	3,2	246	2,2	416	3,9	650	5,8
20	1131	2,2	363	2,1	758	2,8	792	2,0

La figure 5 illustre le traitement par double balayage de la forme quasi-sinusoidale vue précédemment. On peut constater que pour  $\mu = 1$ , les interpolations réalisées dans les sens direct et inverse présentent des oscillations qui n'ont pas la même localisation. Pour des valeurs de  $\mu$  supérieures à 1 on obtient par double balayage (en particulier pour  $\mu > 4$ ) un lissage plus satisfaisant que par simple balayage, ce qui est confirmé par une variation régulière de l'erreur quadratique en fonction du paramètre  $\mu$  (tableau 2).

L'intérêt du traitement par double balayage est mis en évidence par le lissage d'une forme 2D (figure 6). Pour des valeurs faibles de  $\mu$  ( $\mu < 9$ ) les résultats obtenus par simple ou double balayage sont équivalents. Pour des valeurs plus élevées de  $\mu$  ( $\mu > 15$ ), on

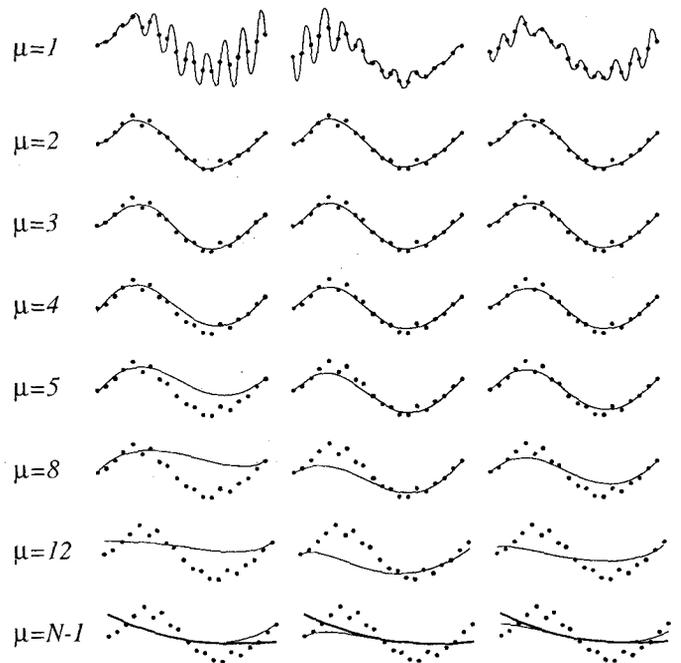


Figure 5. - Lissage d'une répartition 1D avec la base de fonctions  $\{1, t, t^2\}$ . Pour  $\mu = 1$ , les oscillations sont mieux réparties avec  $S(u)$ . La forme  $S(u)$  reste fidèle à la forme globale des points quel que soit  $\mu$ . La courbe en gras, pour  $\mu = N - 1$ , est la solution obtenue par les moindres carrés standard.

Tableau 2 : Evolution de l'erreur quadratique ( $ea$ ) et de la courbure ( $ec$ ) en fonction du paramètre  $\mu$  et du sens de balayage, pour la courbe de la figure 5 ( $m = 2, r = 1$ ).

	gauche/droite		gauche/droite		gauche/droite	
$\mu$	$ea$	$ec$	$ea$	$ec$	$ea$	$ec$
1	0	39,5	0	30	0	11
2	0,09	0,027	0,089	0,023	0,085	0,02
3	0,18	0,013	0,14	0,01	0,13	0,01
4	0,3	0,0077	0,2	0,008	0,21	0,0074
5	0,5	0,0055	0,37	0,006	0,38	0,0055
8	2,3	0,0067	1,5	0,0036	0,97	0,0048
12	7	0,0048	5,1	0,0036	0,97	0,0048
19	2,8	0,0016	2,8	0,0011	2,4	0,0008

obtient par simple balayage des défauts de localisation des formes anguleuses qui n'apparaissent pas quand le lissage résulte d'un double balayage.

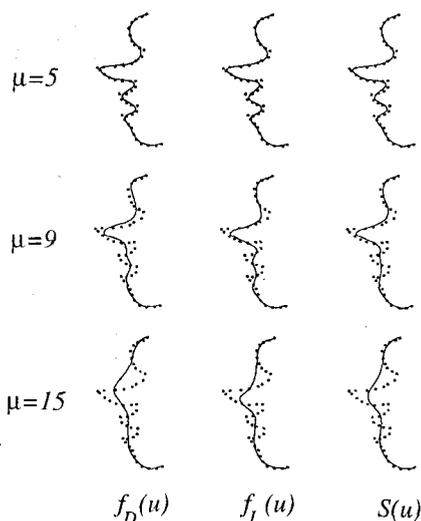


Figure 6. - Lissage par double balayage d'une forme 2D.

## 5. adaptation de l'algorithme à des courbes fermées

L'algorithme proposé n'assure pas la régularité de la solution pour des courbes fermées. Ainsi proposons-nous de transformer une courbe fermée initiale en une courbe non fermée par périodisation (figure 7).

Le principe de la transformation est le suivant. Soit  $C_0 = \{P_i, i = 0, \dots, N\}$  une courbe fermée. On génère une courbe non fermée  $C_1 = \{V_i/V_i = P_j, j \equiv i \text{ modulo } N + 1\}$ . L'indice  $i$  allant de  $-B$  à  $N + B$ , où  $B$  désigne l'intervalle de recouvrement de même ordre de grandeur que le nombre total de points (typiquement  $N/2$ ).

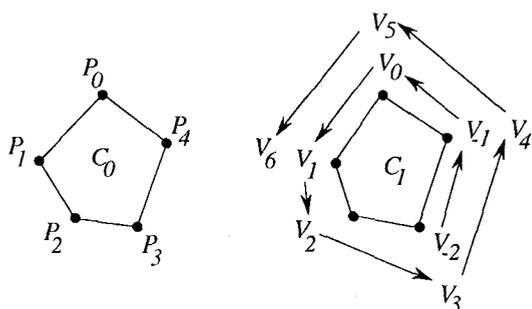
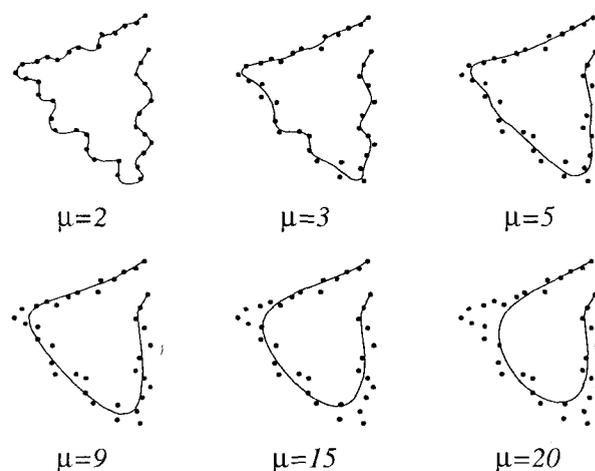


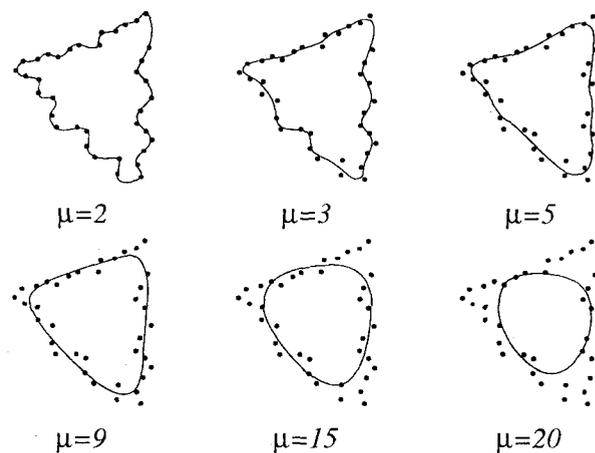
Figure 7. - Transformation de  $C_0$  en  $C_1$  avec  $B = 2$ .

L'algorithme est appliqué à l'ensemble des points de  $C_1$ , mais la solution retenue ne concerne que la partie de la courbe comprise entre  $V_0$  et  $V_{N+1}$ .

Avec cette transformation, le point  $P_0$  devenu  $V_0$  possède des prédécesseurs (ce qui permet de calculer  $g_0(t)$  au moyen des contraintes déduites du calcul de  $g_{-1}(t)$ ). De même,  $P_N$  devenu  $V_N$  possède des successeurs ce qui permet de calculer  $g_N(t)$  par minimisation de l'erreur quadratique sur l'intervalle  $[V_N, V_{N+\mu}]$ .



a) Lissage directe de la courbe originale.



b) Lissage après périodisation de la courbe.

Figure 8. - Lissage d'une courbe fermée ( $B = N/2$ ).

Les fonctions  $g_0(t)$  au point  $V_0 (= P_0)$  et  $g_N(t)$  au point  $V_{N+1} (= P_0)$  tendent à vérifier les équations de contraintes au point de jonction  $P_0$ . La courbe entre  $P_0$  et  $V_{N+1}$  est pratiquement régulière et fermée. La figure 8 montre l'intérêt de cette adaptation de l'algorithme dans le traitement des courbes fermées. La méthode par double balayage peut être appliquée mais n'apporte plus le même intérêt dans le cas d'une courbe fermée, en effet les effets de bord sont déjà éliminés par le recouvrement.

## 6. modèle physique et lissage optimal

Pour un ensemble de points donné, la méthode que nous avons présentée fournit plusieurs courbes en fonction de  $\mu$ . Nous cherchons ici à déterminer un résultat optimal de lissage en introduisant un modèle défini comme suit : la courbe est assimilée à une baguette déformable et chaque point  $(x_i, y_i)$  correspond au point d'attache fixe d'un ressort lui même lié à la baguette (figure 9).

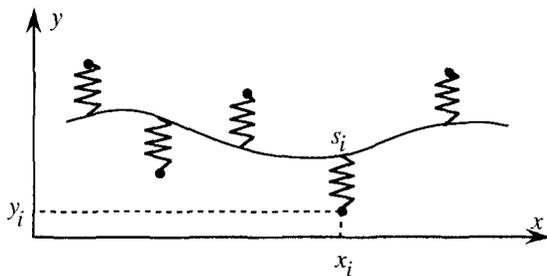


Figure 9. – Baguette en équilibre.

La valeur optimale  $\hat{\mu}$  correspond alors à un état d'équilibre.

Soit  $E_{tot}(\mu)$  l'énergie potentielle totale du système pour une valeur donnée de  $\mu$  et soit  $\nu_\mu(s) = \{x_\mu(s), y_\mu(s)\}$  les fonctions représentatives de la courbe  $C$  paramétrées par l'abscisse curviligne  $s$ , on peut écrire que :

$$E_{tot}(\mu) = E_{def}(\mu) + E_{all}(\mu).$$

$E_{def}(\mu)$  représente l'énergie potentielle de déformation de la courbe et  $E_{all}(\mu)$  l'énergie potentielle d'allongement des ressorts.

$$E_{def}(\mu) = \alpha \int_C \|\nu_\mu^{(2)}(s)\|^2 ds \quad \text{et}$$

$$E_{all}(\mu) = K \sum_1^N \|\nu_\mu(s_i) - \nu_i\|^2$$

$\alpha$  représente le coefficient de rigidité de la baguette et  $K$  la raideur des ressorts.

L'équilibre stable du système est obtenue lorsque l'énergie potentielle  $E_{tot}(\mu)$  est minimum.

$$\hat{\mu} = \arg \min_{\mu} \{E_{tot}(\mu)\} \quad (19)$$

Lorsque  $\mu$  augmente,  $E_{def}(\mu)$  décroît rapidement du fait de la diminution des oscillations, tandis que  $E_{all}(\mu)$  croît.

A titre d'exemple, en fixant  $\alpha = 10$  et  $K = 1$ , la valeur optimale  $\hat{\mu}$  trouvée pour la répartition sinusoïdale de la figure 3, conduit à la solution  $\hat{\mu} = 2$  pour  $m = 2$  et  $\hat{\mu} = 3$  pour  $m = 3$  (figure 10).

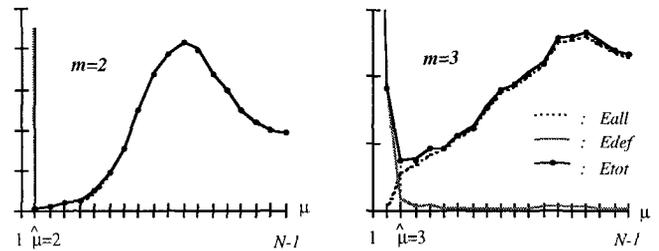


Figure 10. – Evolution de l'énergie interne et de l'erreur quadratique en fonction de  $\mu$ .

La diminution de  $E_{all}(\mu)$  pour des valeurs de  $\mu$  proches de  $N - 1$  est due à un effet de bord et s'élimine lorsqu'on utilise le double balayage.

La même technique de lissage a été appliquée aux configurations 2D de la figure 4. Les solutions optimales correspondantes sont présentées en figure 11.

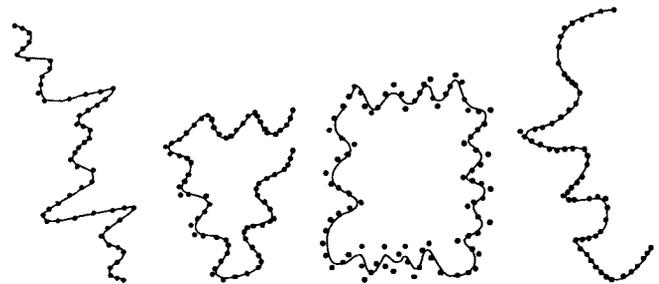


Figure 11. – Lissage de courbes 2D.

Ce modèle permet d'établir une relation entre le paramètre  $\mu$  et le paramètre  $\lambda$  intervenant dans la formulation de Tihonov. En effet, si l'on pose  $\lambda = N \frac{\alpha}{K}$ , minimiser  $E_{tot}(\mu)$  est équivalent à chercher la fonction  $\nu_\mu(s)$  qui minimise la fonctionnelle :

$$\hat{\mu}(\lambda) = \arg \min_{\mu} \left\{ \frac{1}{N} \sum_{i=1}^N \|\nu_\mu(s_i) - \nu_i\|^2 + \lambda \int_a^b \|\nu_\mu^{(2)}(s)\|^2 ds \right\}$$

avec

$$\nu_\mu(s) \in V^r \quad \text{et} \quad V^r = \{\nu_\mu : \nu_\mu(s) \in C^r \times C^r[a, b], \nu_\mu^{(2)} \in L_2\}$$

Ainsi pour effectuer un lissage global, le choix du paramètre  $\mu$  est remplacé par celui du paramètre de régularisation  $\lambda$  plus aisé à interpréter (force de lissage).

Si le modèle choisi est tel que  $\nu_i = g(s_i) + e_i$ , où  $e_i$  représente un bruit pseudo-blanc, on peut chercher la valeur de  $\lambda$  qui minimise  $E_{tot}(\hat{\mu})$  en calculant l'erreur quadratique moyenne de validation croisée ordinaire [CrW.79] [Sto.74].

$$E_{ocv}(\lambda) = \frac{1}{N} \sum_{k=1}^N \|\nu_{\hat{\mu}}^{[k]}(s_k) - \nu_k\|^2$$

où  $\nu_{\hat{\mu}}^{[k]}(s)$  est la courbe lissée obtenue à partir des points  $\nu_i$  excepté le  $k$  ième.

Finalement, ceci revient à minimiser  $E_{ocv}$  par rapport à  $\mu$ . Cette erreur représente aussi l'erreur de prédiction du modèle.

L'exemple de la figure 12 reprend la courbe de la figure 3 pour  $m = 3$  et  $r = 1$ . Dans ces conditions, on obtient un optimum en  $\mu = 4$ .

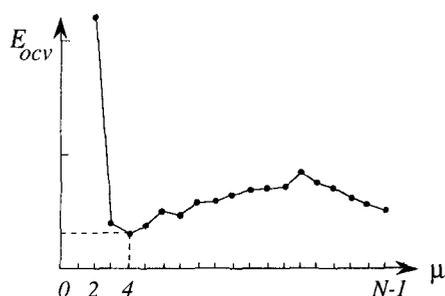


Figure 12. – Evolution de l'erreur quadratique moyenne de validation croisée en fonction de  $\mu$ .

## 7. lissage adaptatif : application à la segmentation d'images

En segmentation d'images, les méthodes de détection de contours conduisent à définir des frontières d'objets au moyen d'un ensemble de pixels connexes. Ces méthodes fondées généralement sur des dérivations numériques sont très sensibles aux bruits dans l'image. Des post-traitements sont mis en œuvre pour former à partir de l'ensemble des points de contours candidats, une courbe fermée. Deux étapes sont généralement nécessaires pour obtenir ce résultat : la sélection de points candidats connexes définissant une chaîne de contour, puis l'association de ces chaînes candidats par des opérations de fermeture de contours.

Nous utilisons notre méthode de lissage comme modèle de contour déformable pour obtenir une courbe régulière fermée et épousant les contours de l'objet. Une ligne polygonale est initialisée sur l'image puis lissée en adaptant la valeur de  $\mu$  en fonction de la valeur du gradient en chacun des points. Nous nous fondons sur l'heuristique suivante [Koh.81] : les points de forts contrastes dans l'image indiquent des contours bien localisés tandis que les points de faibles contrastes correspondent à des contours mal définis.

Nous procédons en deux phases :

- Génération d'une courbe polygonale  $C_0 = \{P_i, i = 1, \dots, N\}$  sur les frontières d'un objet. Cette courbe est créée soit manuellement, soit à partir d'un processus de croissance de région, ou de détection de contours.
- Lissage de  $C_0$ . Pour appliquer l'heuristique, il faut modifier la valeur de  $\mu$  en fonction du gradient de chaque point de  $C_0$ .

$$\mu_i = \Psi(\|\nabla P_i\|) \quad \text{pour } \mu \in [\mu_0, \mu_\infty]$$

La fonction  $\Psi$  permet d'imposer à la courbe de passer par les points de fort gradient ( $\mu$  doit donc être proche de  $\mu_0$ ) et inversement de privilégier un lissage par approximation quand le gradient est faible ( $\mu$  doit donc se rapprocher de  $\mu_\infty$ ).

La plage des valeurs du module du gradient est divisée en  $p$  classes, mises en correspondance avec les valeurs de  $\mu$  comprises entre  $\mu_0$  et  $\mu_\infty$  au moyen de la fonction  $\Psi$ .

La fonction  $\Psi$  est une connaissance *a priori*, elle peut être établie à partir de l'histogramme des gradients caractéristique du type d'image à analyser.

La figure 13 montre une forme typique de la fonction  $\Psi$ .

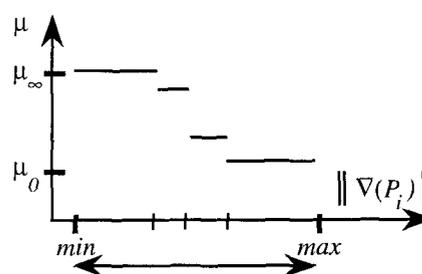


Figure 13. – Exemple de correspondance établie au moyen de la fonction  $\Psi$  entre les  $p$  classes (4) des valeurs du gradient et le paramètre  $\mu$ .

La figure 14 montre les résultats obtenus au moyen de cette méthode sur des répartitions de points définissant quatre formes différentes (carré parfait, ligne, triangle et carré avec oscillations).

Sur l'image originale du carré parfait, les flèches associées à un signe + ou - indiquent respectivement la croissance ou la décroissance du gradient. Pour les autres figures, la valeur du gradient en chaque point est représentée par un niveau de luminosité. Le lissage est réalisé pour chacune des images originales pour trois valeurs du paramètre  $p$  ( $p = 2, 3, 5$ ).

Nous constatons l'effet d'interpolation quand le gradient est fort et l'effet d'approximation quand il est faible (voir par exemple l'approximation réalisée au niveau des coins du carré parfait).

Affecter la valeur 2 à  $p$ , revient à faire l'hypothèse que le gradient est réparti en deux classes. On obtient alors une courbe lissée qui passe brusquement d'un mode de lissage à un autre. Le passage brusque de l'approximation à l'interpolation se traduit sur la courbe lissée par une singularité signalée par une flèche (voir sur la figure 13 les images de la deuxième colonne). En considérant une répartition des valeurs du gradient en un nombre de classes plus important ( $p = 3$  ou  $5$ ), ces phénomènes transitoires sont atténués.

Enfin la méthode est utilisée pour effectuer un lissage des contours de la cavité du ventricule gauche sur des images échocardiographiques. La figure 15 montre un exemple de résultat obtenu par cette méthode sur des images réelles.

Les points lumineux apparaissant sur l'image 15.b sont les sommets du contour polygonal à lissée obtenu par une méthode de croissance de régions [DPL.94].

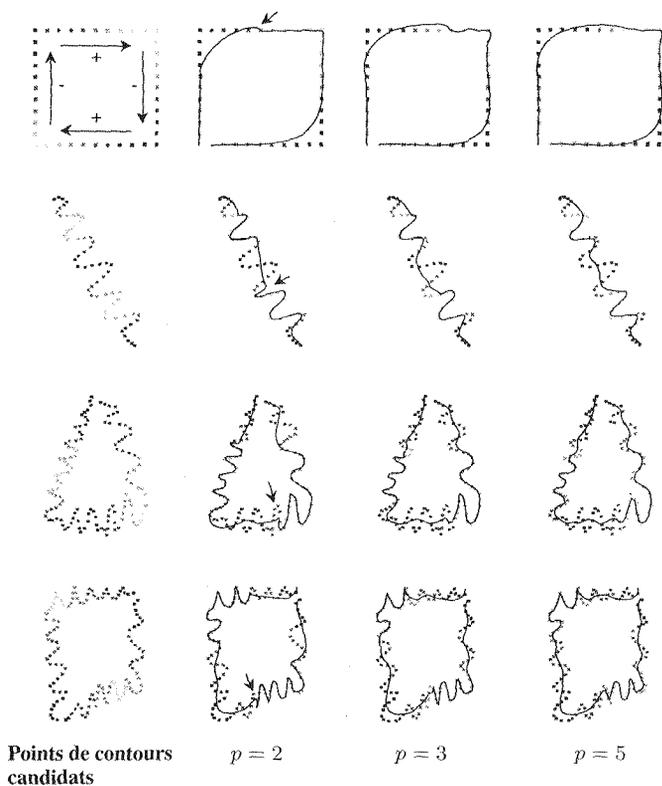
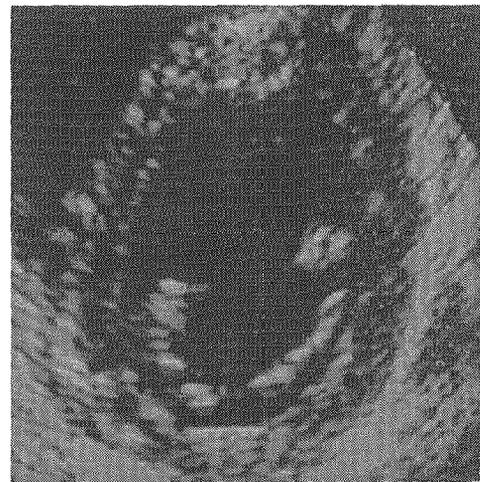
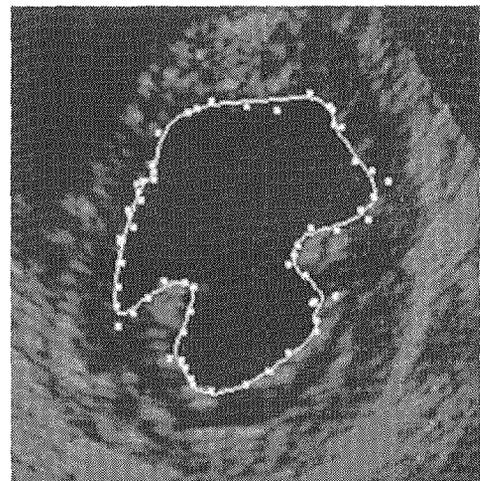


Figure 14. – Lissage de courbes en fonction du gradient. La variable  $p$  représente le nombre de classes des valeurs du gradient. Sur chaque figure le module du gradient croît et décroît 2 fois le long de la courbe, de ce fait le lissage réalise 2 fois une interpolation suivie d'une approximation. Le passage de l'approximation à une interpolation, indiquée par une flèche, engendre une singularité qui s'atténue quand  $p$  augmente.



a) image originale



b) contour lissé

Figure 15. – Détection de contours dans une image échocardiographique.

## 8. conclusion

Nous avons présenté un algorithme original de lissage de courbes fondé sur une minimisation locale de l'erreur avec contraintes géométriques. La formulation de la solution pour chaque intervalle est directe. Son implantation machine n'exige pas de grandes capacités en mémoire si on se limite, comme dans nos exemples, à des valeurs de  $m$  de l'ordre de 3 et si l'on choisit une base polynômiale standard. L'effet du lissage peut être contrôlé simplement avec un seul paramètre  $\mu$ . En modifiant ce paramètre, on peut obtenir une courbe présentant de nombreux détails ou, au contraire, privilégier la forme globale déterminée par les points dans l'espace. Notre approche de lissage associe la méthode de lissage par les moindres carrés et la méthode d'approximation par les splines. Elle met à profit la souplesse des splines (car la solution est recherchée par morceaux sur chaque intervalle) et l'avantage des moindres carrés (car elle utilise une base de fonctions quelconque : polynomiales, trigonométriques,...). De plus, par rapport aux méthodes d'approximation par les B-splines proposées par Bartels et al. et St Marc et al. [BBB.87] [SMM.90], la

formulation directe de la solution évite l'étape de recherche des points dominants qui constitue souvent une difficulté importante de mise en œuvre [GuA.92].

La méthode que nous proposons, appliquée au lissage de points de contours détectés dans une image, constitue une étape simple et efficace pour achever un processus de segmentation. A partir d'une ligne polygonale définissant approximativement un contour on obtient une courbe régulière qui respecte d'une part les informations locales du gradient mesuré et d'autre part la forme globale de l'objet à détecter.

## BIBLIOGRAPHIE

- [BaP.82] F. Badii & B. Peikari, « Functional approximation of planar curves via adaptive segmentation », *Int. J. Syst. Sci.*, Vol. 13, n°6, 1982, pp. 667-674.
- [BBB.87] R. Bartels, J. Beatty & B. Barsky, *An introduction to splines for use in computer graphics and geometric modelling*, Morgan Kaufmann publishers, 1987.

- [BoK.86] T. Boulton & J. Kender, « Visual surface reconstruction using sparse depth data », *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, 1986, pp. 68-76.
- [ChC.93] Mei-Hsing Chen & Roland T. Chin, « Partial Smoothing Splines for Noisy + Boundaries with corners », *IEEE PAMI*, Vol. 15, n°11, 1993, pp. 1208-1216.
- [CrW.79] P. Craven & G. Wahba, « Smoothing noisy data with spline functions - Estimating the correct degree of smoothing by the method of generalized cross-validation », *Numerische Mathematik*, n°31, 1979, pp. 377-403.
- [Dav.79] L. S. Davis, « Shape matching using relaxation techniques », *IEEE Trans. PAMI*, Vol. 1, 1979, pp. 60-72.
- [DPL.94] S. Djéziri, E. Petit et J. Lemoine, « Un algorithme de croissance de régions par déformation géométrique de contours », *9<sup>ème</sup> Congrès RFIA-AFCET, Reconnaissance des Formes et Intelligence Artificielle*, Vol. 1, 1994, pp. 715-720.
- [Don.91] R. Dony, *Calcul des parties cachées : Approximation des courbes par la méthode de BEZIER et des B-splines*, 2<sup>ème</sup> édition Masson.
- [DuH.73] R. O. Duda & P. E. Hart, *Pattern classification and scene analysis*, New York : Wiley, 1973, pp. 328-329.
- [Dun.86] J. G. Dunham, « Optimum uniform piecewise linear approximation of planar curves », *IEEE trans. PAMI*, Vol. 8, n°1, 1986.
- [FiB.86] M. A. Fischler & R. C. Bolles, « Perceptual Organization and Curve Partitioning », *IEEE PAMI*, Vol. 8, n°1, 1986, pp. 100-105.
- [FrD.77] H. Freeman & L. S. Davis, « A corner-finding algorithm for chain-coded curves », *IEEE Trans. Comput.*, Vol. C-26, 1977, pp. 297-303.
- [GuA.92] A. Guézic & N. Ayache, « Lissage et reconnaissance de courbes gauches bruitées », *Traitement du Signal*, Vol. 9, n°6, 1992, pp. 473-490.
- [Koh.81] R. R. Kohler, « A segmentation based on thresholding », *Computer Graphics and Image Processing*, Vol. 15, 1981, pp. 319-338.
- [KuD.82] Y. Kurozumi & W. A. Davis, « Polygonal approximation by the minmax method », *Comput. Graphics Image Processing*, Vol. 19, 1982, pp. 248-264.
- [LeP.88] D. Lee & T. Pavlidis, « One-dimensional regularization with discontinuities », *IEEE PAMI*, Vol. 10, n°6, 1988, pp. 822-829.
- [MuS.85] D. Mumford & J. Shah, « Boundary detection by minimizing functionals », *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, 1985, pp. 22-26.
- [PaH.74] T. Pavlidis & S. L. Horowitz, « Segmentation of planes curves », *IEEE Trans. Comput.*, Vol. C-23, 1974, pp. 860-870.
- [Ram.72] U. Ramer, « An iterative procedure for the polygonal approximation of plane curves », *Comp. Graphics Image Processing*, Vol. 1, 1972, pp. 244-256.
- [RaR.91] Kr. B. Ray & Ku. S. Ray, « A new approach to polygonal approximation », *Pattern Recognition Letters* 12, 1991, pp. 229-234.
- [Rob.85] J. Roberge, « A data reduction algorithm for planar curves », *Comput. Vision, Graphics, Image Processing*, Vol. 29, 1985, pp. 168-195.
- [RoJ.73] A. Rosenfeld & E. Johnston, « Angle detection on digital curves », *IEEE Trans. Comput.*, Vol. C-22, 1973, pp. 875-878.
- [RoW.75] A. Rosenfeld & J.S. Weszka, « An improved method of angle detection on digital curves », *IEEE Trans. Comput.*, Vol. C-24, 1975, pp. 940-941.
- [SaS.78] P. V. Sankar & C. V. Sharma, « A parallel procedure for the detection of dominant points on a digital curve », *Comput. Graphics Image Processing*, Vol. 7, 1978, pp. 403-412.
- [ShA.89] B. Sharayha & D. Anderson, « Optimal estimation of contour properties by cross-validated regularization », *IEEE PAMI*, Vol. 11, n°6, 1989, pp. 600-610.
- [SIG.80] J. Slansky & V. Gonzalez, « Fast polygonal approximation of digitized curves », *Pattern Recognition*, Vol. 12, 1980, pp. 327-331.
- [SMM.90] P. Saint-Marc & G. Medioni, « B-spline contour representation and symmetry detection », In *First European Conference on Computer Vision (ECCV)*, Antibes 1990.
- [Sto.74] M. Stone, « Cross-validated choice and assessment of statistical prediction », *J. Roy. Stat. Soc. Ser. B*, Vol. 36, n°2, 1974, pp. 11-147.
- [TeC.89] Cho-Huak Teh & Roland T. Chin, « On the detection of dominant points on digital curves », *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 11, 1989, n°8, pp. 859-872.
- [Tih.63] A. N. Tihonov, « Regularisation of incorrectly posed problems », *Sov. Math. Dokl.*, Vol. 4, 1963, pp. 1624-1627.
- [WuW.93] Wen-Yen Wu & Mao-Jiun J. Wang, « Detecting the dominant points by the curvature based polygonal approximation », *CVGIP : Graphical Models and Image Processing*, Vol. 55, n°2, 1993, pp. 79-88.

Manuscrit reçu le 6 Septembre 1994.

## LES AUTEURS

Salim DJEZIRI



Salim Djéziri est docteur d'université depuis 1994. Sa thèse a porté sur l'élaboration de méthodes de segmentation par des approches coopératives région/contour. Il a développé une méthode de contours déformable par critères géométriques. Il est actuellement chercheur à l'Université des Trois Rivières au Québec où il s'intéresse particulièrement à la reconnaissance hors ligne de signatures.

Eric PETIT



Eric Petit a soutenu en 1985 une thèse de doctorat portant sur la segmentation d'images échocardiographiques. Nommé enseignant-chercheur à l'Université Paris XII Val de Marne, il a contribué à la création de la division Traitement d'Images et Reconnaissance des Formes du L.E.R.I.S.S. et constitue un des principaux animateurs d'une équipe qui développe des travaux de recherche sur des opérateurs bas niveau de segmentation d'images (détection de contours, modèles déformables, analyse de texture).

Jacques LEMOINE



Jacques Lemoine est ingénieur Supélec et a soutenu une thèse de docteur-ingénieur en 1981 portant sur la réalisation d'une unité de traitement d'image en temps réel. Il est actuellement professeur à l'Université Paris XII Val de Marne. Ses activités de recherche au L.E.R.I.S.S. concernent la reconnaissance en ligne d'écriture manuscrite et la segmentation d'images texturées multirésolutions.