

Paramètres locaux pour une méthode de contours actifs

J.-J. ROUSSELLE, N. VINCENT, N. MONTIER, J.-L. MONTEIRO

Laboratoire d'Informatique (LI) Université François Rabelais de Tours, 64, av. Jean Portalis, 37200 Tours, France

`rouselle, vincent@univ-tours.fr`

Résumé – L'introduction des contours actifs (snakes) comme nouvelle méthode d'extraction des contours dans le domaine du traitement d'images a constitué une avancée majeure en ce qui concerne les méthodes de segmentation. Malheureusement, la modélisation de la fonctionnelle d'énergie qui doit être associée à chaque image est délicate. En effet, elle dépend de nombreux paramètres qui sont souvent fixés empiriquement par le concepteur du système. Nous proposons ici une méthode originale qui permet de résoudre, dans la plupart des cas, le problème de la détermination de ces paramètres. La méthode est présentée dans le cadre de l'algorithme "greedy". Nous considérons ici que les coefficients de la fonctionnelle d'énergie ne sont pas globaux, ni indépendants des points considérés mais qu'ils sont locaux. Nous les recherchons donc en chaque point en effectuant le meilleur choix dans un ensemble de paramètres qui est créé par des tirages aléatoires. L'application sur un ensemble d'images variées, dont des radiographies, montre la convergence de la méthode proposée.

Abstract – *Into the field of the image processing, the introduction of active contours (snakes) as a new method of contour extraction has constituted a major progress with regard to the segmentation method. Unfortunately, the functional of energy which must be associated to each image is not easy to settle. Indeed, it depends on many parameters which are often empirically determined by the system designer. We propose here an original method which makes possible, in most of the cases, to solve the problem of the determination of these parameters. The method presented is based on the greedy algorithm. We consider here that the coefficients of the functional of energy are not global, neither independent of the considered points, but that they are local. We thus seek them on each point by carrying out the best choice in a whole of parameters which are created randomly. The application on a varied set of images, including X-Ray, shows the convergence of the suggested method.*

1. Introduction

La méthode des contours actifs a été introduite en 1987 par Kass [4]. Elle permet d'extraire un objet par détermination de son contour. Le contour est modélisé par une courbe qui doit se déformer en fonction de l'image pour se fixer sur les frontières de l'objet. C'est l'introduction de forces et d'énergies s'appliquant en tout point de la courbe, qui conduit à sa déformation. La puissance de la méthode provient de la possibilité de choisir les énergies en fonction du problème que l'on veut résoudre, de l'image et de l'objectif. Ainsi l'utilisation de cette approche est devenue très courante et de nombreuses publications prouvent leur utilisation dans des domaines très divers ; citons en particulier celui de la biométrie [7], du tracking [6] ou du traitement de l'écrit [5]. Des variantes ou améliorations sont régulièrement proposées. Elles concernent souvent la définition des forces en présence, telle la force ballon proposée dans [2] ou le "Gradient Vector Flow" [10].

Si la méthode permet d'obtenir des résultats remarquables, même dans les cas les plus difficiles, la qualité des résultats dépend fortement du choix des nombreux paramètres qui interviennent dans la fonctionnelle d'énergie à minimiser. Tous les auteurs sont d'accord pour dire que la méthode est délicate lors de ce choix des paramètres. Ces paramètres permettent de doser le rapport entre les différentes forces du modèle. Ce réglage s'effectue généralement par essai - erreur, ce qui interdit à un expérimentateur ignorant la méthode d'utiliser la méthode des contours actifs dans ses applications particulières.

Pour répondre à cette principale critique des contours actifs, sont développées actuellement d'autres méthodes dérivées telle la méthode des « *Level Sets* » [8]. Elle permet, d'une part, de mieux gérer la topologie des objets et d'autre part, de réduire le nombre des paramètres de la méthode. Par contre, les temps de calcul s'en trouvent augmentés et il est difficile d'utiliser la méthode dans des applications temps réel.

Nous proposons ici, tout en restant dans le cadre des contours actifs proprement dits, d'introduire une nouvelle modélisation de ces coefficients. Cela permet de réaliser leur optimisation de manière automatique en même temps qu'est réalisée l'optimisation de la fonctionnelle d'énergie, associée au problème de détection du contour.

Après avoir rappelé les principes généraux des contours actifs et en particulier la modélisation qui est faite dans l'algorithme traditionnel du *greedy*, nous présenterons notre modèle. Quelques exemples d'applications nous permettront d'illustrer l'évolution des paramètres au cours des itérations et ainsi de mieux comprendre le rôle des paramètres, et d'expliquer certains choix qui ont été faits dans la littérature de manière empirique.

2. Contour actif et *greedy* algorithm

La méthode des contours actifs est une méthode générique. Elle peut être abordée suivant une approche continue ou suivant une approche discrète.

Dans le cas continu, la courbe \mathcal{C} est confondue avec sa représentation paramétrique, la fonction continue v à valeur dans le plan \mathcal{P} . On note s une abscisse curviligne.

$$v : [0,1] \rightarrow \mathcal{P}, s \mapsto v(s) \quad (1)$$

Dans le cas du traitement d'images numériques, le contour est un ensemble de points chaînés. On peut considérer que le contour est connu par l'intermédiaire d'un certain nombre de points qui seront plus ou moins nombreux en fonction de la précision recherchée. Avant de préciser le principe de l'algorithme *greedy*, nous présentons la fonctionnelle d'énergie qu'il s'agit de minimiser pour caractériser le contour.

2.1 Energies et fonctionnelle d'énergie

Pour définir les propriétés du contour recherché, plusieurs types de forces sont considérés. Les forces internes gèrent la cohésion intrinsèque du contour, la régularisation de la courbe, la présence ou non de coins, la répartition régulière des points le long de la courbe dans une approche discrète. Les forces externes correspondent à l'attache aux données environnantes, c'est-à-dire l'image elle-même. Divers éléments peuvent être considérés, par exemple le gradient, une intensité lumineuse, une couleur ou une propriété de texture. Les forces de contexte peuvent être introduites comme la force ballon [2] ou d'autres contraintes propres à l'application ; elles font évoluer le contour de manière particulière. Finalement, l'évolution du contour s'effectue de manière à minimiser l'énergie globale du contour au sein de l'image. On peut exprimer cette énergie par la formule :

$$E(C) = \alpha E_{\text{continuité}} + \beta E_{\text{courbure}} + \gamma E_{\text{gradient}} + \kappa E_{\text{ballon}} + \text{etc.} \quad (2)$$

Les différentes énergies sont définies de manière très précise. Par contre, on voit apparaître des paramètres qui permettent de doser le rapport entre les différentes énergies du modèle, pour lesquels aucune technique automatique de calcul n'est donnée. Le choix des paramètres est effectué par l'utilisateur qui ne peut être que le développeur. Une étape de mise au point de la méthode est nécessaire pour fixer de « bonnes valeurs ». L'optimum de ces paramètres n'est pas assuré et deux développeurs peuvent proposer des ensembles de paramètres qui sont très différents mais qui obtiennent finalement des résultats tout à fait similaires.

Dans le cas d'une approche discrète, le contour est un ensemble de points ordonnés $M_i, i \in [1, N]$. L'énergie du contour est alors assimilée à la somme des énergies associées aux N points définissant le contour.

$$E(\mathcal{C}) = \sum_{i=1}^N (\alpha E_{\text{cont}}(M_i) + \beta E_{\text{cour}}(M_i) + \gamma E_{\text{grad}}(M_i) + \kappa E_{\text{ball}}(M_i)) \quad (3)$$

Il s'agit alors de déterminer \mathcal{C} , c'est-à-dire la fonction de paramétrisation de \mathcal{C} , ou bien l'ensemble des N points qui définissent \mathcal{C} .

2.2 Méthode d'optimisation : algorithme *greedy*

Plusieurs méthodes ont été proposées pour résoudre ce problème. On en compte trois principales, la famille des ap-

proches variationnelles [4], la famille des approches reposant sur la programmation dynamique [1], et l'algorithme *greedy* [9] qui est reconnu pour son temps de traitement plus rapide [3]. Nous nous plaçons dans le cadre d'une implémentation par un algorithme *greedy*.

Dans la méthode du *greedy*, on écrit que l'énergie en un point M est une somme pondérée d'énergies E de différentes sortes :

$$E(M) = \sum_j V_j E_j(M) \quad (4)$$

avec V vecteur des pondérations. Soit \mathcal{C} une courbe discrétisée par N points ordonnés M . L'énergie de la courbe $E(\mathcal{C})$ peut être estimée par :

$$\hat{E}(\mathcal{C}) = \sum_{M \in \mathcal{M}} E(M) = E(\mathcal{M}) \quad (5)$$

avec $\mathcal{M} = \{M_1, \dots, M_N\}$. On cherche l'ensemble \mathcal{M} des points tel que $E(\mathcal{M})$ soit minimum, c'est-à-dire plutôt la suite finie (M_1, \dots, M_N) pour laquelle l'énergie est minimale.

L'algorithme *greedy* est un algorithme itératif, qui à chaque itération, déplace un point unique pour constituer un nouveau contour. Tous les points sont ainsi traités successivement.

L'initialisation conduit au choix des N premiers points. Le processus permet ainsi de construire une suite de points $M_i, i \in N$, dont N points consécutifs définissent une approximation du contour. On note \mathcal{M}_v le contour final défini par :

$$M_v = \{M_{v-N+1}, \dots, M_v\} \quad (6)$$

Un point M_i est calculé à partir des N points précédents et définit le contour :

$$\mathcal{M}_i = (\mathcal{M}_{i-1} \setminus \{M_{i-N}\}) \cup \{M_i\} \quad (7)$$

(avec \setminus pour : privé de...) où M_i est le point du voisinage de M_{i-N} qui permet de minimiser la fonction considérée comme fonction de M .

3. Modélisation locale des paramètres

3.1 Fonctionnelle d'énergie

Dans la plupart des modèles proposés, la formule de référence utilisée pour le calcul de l'énergie en un point est celle contenue implicitement dans la formule (3). Au point M , on a :

$$E((\mathcal{M}_{i-1} \setminus \{M_{i-N}\}) \cup \{M_i\}) \quad (8)$$

Cette formulation laisse sous-entendre que les a_j sont des paramètres globaux, paramètres de la méthode, tandis que $E_j(M)$ désignerait les énergies dont la valeur dépend, en fait, du point M mais aussi du contour considéré, et de l'image. Une analyse plus précise des méthodes proposées

nous amène à une reformulation de l'expression :

$$a_j E_j(M) = v_j(M) F_j(M) = b_j T_M[F_i(M)] \quad (9)$$

où $F_i(M)$ correspond à une formule générale de l'énergie qui est fonction de M , du contour et de l'image et T_M une transformation numérique qui dépend du point M . Nous faisons entrer dans cette formulation le processus de normalisation locale des énergies dans le voisinage du point décrit par Williams et Shah [9]. Le plus souvent, c'est une transformation du premier degré du type $x \mapsto a_M x + b_M$. Le choix de b_j reste difficile et global pour tous les points de l'image.

Nous proposons de ne pas rendre explicite la transformation, ayant en particulier la même expression en chaque point de l'image, mais plutôt de n'associer aucune connaissance *a priori* sur les pondérations à associer à une énergie en un point. Au contraire, cette pondération devient une inconnue dans la fonctionnelle d'énergie qui doit être minimisée. Le problème de minimisation s'écrit alors : trouver l'ensemble des points M d'un contour et l'ensemble des coefficients a_{ij} qui rendent minimum :

$$E(M_1, \dots, M_N, a_{ij}) = \sum_{i=1}^N \sum_{j=1}^p a_j(M_i) F_j(M_i) \quad (10)$$

avec $a_{ij} \in \mathfrak{R}, M_i \in I$.

3.2 Optimisation

Nous avons choisi de procéder par la méthode de l'algorithme *greedy*. Il s'agit donc, à l'itération k , de déterminer la meilleure valeur du couple (M, V) où M est un point et V un p-uplet de réels qui minimise :

$$E(M, V) = \sum_{j=1}^p v_j F_j(M) \quad (11)$$

En ce qui concerne M , il est à chercher dans le voisinage du point M_{k-N} et V est un vecteur à p composantes. La recherche ne s'effectue pas dans \mathbb{R}^p car, alors, la solution évidente $V = 0$ ne nous intéresse pas. Nous contraignons l'espace de recherche à la sphère S^p unité ; ainsi (M_k, V_k) appartient à $\mathcal{D}_k = \mathcal{V}(M_{k-N}) \times S^p$.

De manière à éviter les minima locaux, la recherche de M_k peut être effectuée dans différents types de voisinage $\mathcal{V}(M_{k-N})$; en particulier le voisinage de M_{k-N} peut ou non contenir le point M_{k-N} .

On prendra donc $\mathcal{V}(M_{k-N})$ comme ensemble de recherches.

3.2.1 Détermination de V_k

Nous souhaitons effectuer les calculs en un temps fini donc en un nombre fini d'opérations, c'est-à-dire en un nombre fini de calculs de $f(M, V)$ sur \mathcal{D}_k . Nous pourrions échantillonner \mathcal{D}_k , mais nous choisissons de balayer exhaustivement $\mathcal{V}(M_{k-n})$ et de ne faire qu'un calcul par point de ce

voisinage. La valeur de v est tirée aléatoirement et V dans S^p et l'on a :

$$\forall M \in \mathcal{V}(M_{k-N}) \begin{cases} E(M, V) < E(M, V_{k-N}) \Rightarrow V_M = V \\ E(M, V) > E(M, V_{k-N}) \Rightarrow V_M = V_{k-N} \end{cases} \quad (12)$$

Ceci implique que $E(M, V)$ est décroissante. Le résultat pour (M_k, V_k) est influencé par l'ordre du parcours du voisinage. Nous avons rendu aléatoire cette compétition ; nous ne sommes donc intéressés que par la convergence de la méthode. Les suites $E(M, V)$ sont décroissantes. Expérimentalement, nous avons observé la convergence de la méthode dans des images très diverses, dont certaines difficiles. Nous présentons les résultats dans la section suivante.

4. Résultats et interprétation

Dans chacune des trois images que nous présentons, les forces utilisées sont : la continuité, la courbure, le gradient, la force ballon et aucun paramètre n'a été fixé a priori.

4.1 Résultats

Sur une image de synthèse de forme complexe alliant convexité, concavité et angles, le contour actif se fixe bien sur le contour.

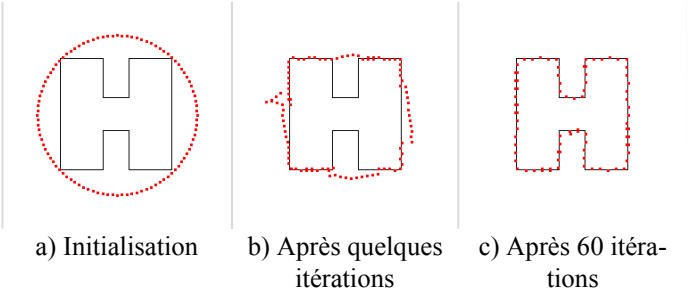


Figure 1 – Image de synthèse

Sur une radio d'épaule, où l'objet d'intérêt est la tête humérale, la progression vers le contour s'effectue correctement malgré une image fortement bruitée.

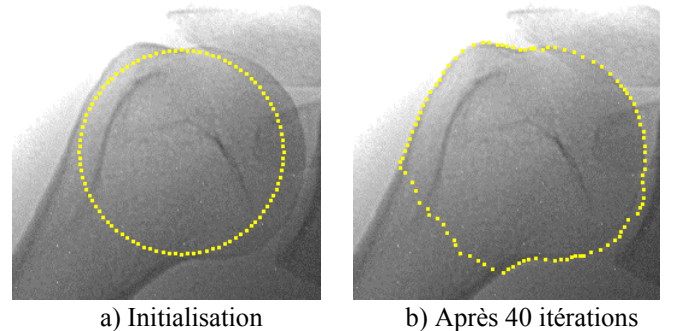
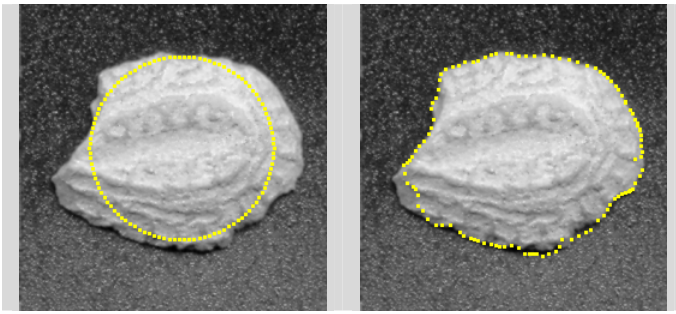


Figure 2 - Tête humérale

D'autres images de formes diverses, naturelles ou de synthèse, ont été testées avec des résultats satisfaisants.



a) Initialisation c) Résultats après 50 itérations

Figure 3 - Contour actif autonome sur une image complexe

4.2 Analyse de l'évolution des paramètres

N'ayant fait aucune hypothèse sur la valeur des paramètres, il nous a semblé intéressant d'étudier l'évolution des paramètres choisis à chaque itération. Sur la Figure 2 est représentée l'évolution des paramètres de quatre énergies sur 100 itérations. On observe une oscillation plus ou moins importante des valeurs autour d'une courbe de tendance qui évolue faiblement comme on le voit sur la Figure 3. On a donc défini un algorithme de contour actif sans paramètre mais également une manière constructive de déterminer un n-uplet de paramètres.

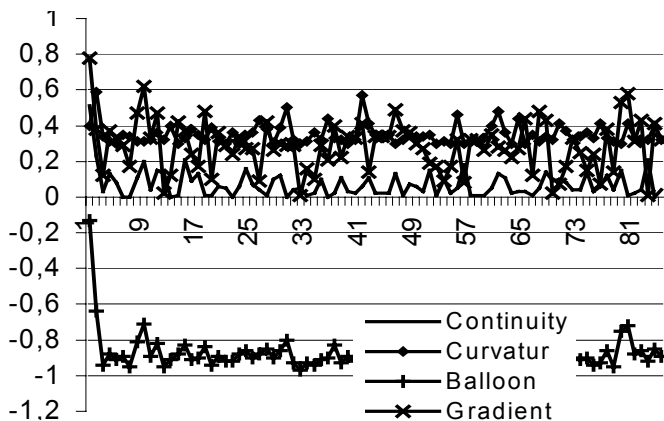


Figure 4 - Evolution des coefficients

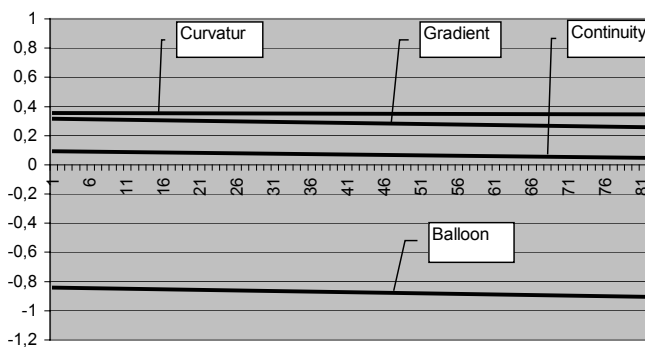


Figure 5 -Tendance de l'évolution des paramètres

5. Conclusion

Nous avons présenté le contour actif à paramètres aléatoires. Il est basé sur un algorithme *greedy*. Après l'avoir justifié théoriquement par une modélisation mathématique, nous avons présenté l'algorithme modifié puis des résultats sur des images. Il est remarquable d'obtenir des résultats aussi satisfaisants sur des images aussi complexes que des radios d'épaule sans aucun réglage de paramètres. Il ne reste que l'initialisation du contour qui peut s'automatiser facilement.

Références

- [1] A.A. Amini, S. Tehrani, T.E. Weymouth - *Using Dynamic Programming for Minimizing the Energy of Active Contours in the Presence of Hard Constraints*, 2nd Int. Conf. Comput. Vision, décembre 1988, Tampa, FL, p. 95-99.
- [2] L.D. Cohen - *On Active Contour Models and Balloons*, Computer Vision, Graphics, and Image Processing: Image Understanding, vol. 53, n° 2, mars 1991, p. 211-218.
- [3] J. Denzler et H. Niemann - *Evaluating the Performance of Active Contour Models for Real-time Object Tracking*, Second Asian Conference on Computer Vision, Singapore, 1995, p. II/341-II/345.
- [4] M. Kass, A. Witkin, D. Terzopoulos - *Snakes: Active Contour Models*, In Proceedings of the first International Conference on Computer Vision, June 1987, p. 259-268.
- [5] O. Lavielle, X. Molines, F. Angella, P. Baylou - *Active Contours Network to Straighten Distorted Text Lines*, ICIP, Thessalonique, October 2001.
- [6] M. Liévin, P. Delmas, P. Y. Coulon, F. Luthon et V. Fristot - *Automatic Lip Tracking: Bayesian Segmentation and Active Contours in a Cooperative Scheme*, ICMCS'99, IEEE Int. Conf. On Multimedia, Computer and Systems, June 1999, Florenza, Italy, vol. 1, p. 691-696.
- [7] J.-J. Rousselle, J. Brilhault, N. Vincent, D. Champion, L. Favard, - *Study of the Tibio-Femoral Joint Using Active Contours*, PCIMA'2001, International ICSC Congress on Computational Intelligence Methods and Applications, June 19-22, Bangor, Wales, p. 111-11
- [8] J.A. Sethian - *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*, Cambridge University Press, 1996-1999.
- [9] D.J. Williams et M. Shah - *A Fast Algorithm for Active Contours and Curvature Estimation*, CVIGP Computer Vision Graphics Image Process: Image Understanding, vol. 55, n° 1, janvier 1992, p. 14-26.
- [10] C. Xu, J. L. Prince - *Generalized Gradient Vector Flow External Forces for Active Contours*, *Signal Processing*, vol. 71 (2), 1998, p. 131-139.