

Une Approche de Détection de Contours Basée sur la Métaheuristique OEP

Safia Djemame

Département d'informatique
Faculté des sciences
Université Ferhat Abbas
Sétif, 19000, ALGERIE
djemame@univ-setif.dz

Mohamed Batouche

Computer Science Department
College of engineering
Mentouri University
Constantine, 25000, ALGERIE
batouche@umc.edu.dz

.....

RÉSUMÉ. Dans ce travail, nous décrivons une nouvelle approche de détection de contours, basée sur une méta-heuristique d'optimisation : l'optimisation par essaim de particules (OEP). Le problème de la segmentation est décrit comme un problème d'optimisation, et le processus se déroule en comparant une image étalon au contour obtenu par la méthode OEP. La fonction de Fitness permet de mesurer la qualité de la segmentation obtenue. Les résultats sont comparés à des algorithmes de la littérature, et s'avèrent prometteurs

ABSTRACT. In this paper, we describe a new approach of edge detection, based on optimization metaheuristics: PSO. The segmentation problem is described as an optimization problem, and the process is carried out by comparing a ground truth image with the edge obtained by PSO. A fitness function measures the quality of the obtained segmentation. Results are compared to well-known algorithms of the literature, and appears promising.

MOTS-CLÉS : Traitement d'images, détection de contours, vie artificielle, métaheuristique d'optimisation, optimisation par essaim de particules, émergence.

KEYWORDS: image processing, edge detection, artificial life, optimization metaheuristics, particle swarm optimization, emergence.

.....

1. Introduction

La segmentation d'images est l'une des opérations les plus importantes en traitement d'images. Elle permet d'extraire des informations pertinentes de l'image. Il existe une grande variété de techniques de segmentation, mais aucune n'est efficace sur tous les types d'images. D'où ce domaine de recherche reste ouvert pour trouver des méthodes plus évoluées. Dans ce travail, nous explorons le monde de la vie artificielle et des techniques bio-inspirées pour proposer une méthode de segmentation par extraction de contours. Ce travail est basé sur l'optimisation par essaim de particules.

2. L'optimisation par essaim particulaire

2.1. Origines

L'optimisation par essaim particulaire (OEP) est une méthode née en 1995 aux Etats Unis sous le nom de *Particle Swarm Optimization (PSO)*. Initialement, ses deux concepteurs, Russel Eberhart et James Kennedy, cherchaient à modéliser des interactions sociales entre des « agents » devant atteindre un objectif donné dans un espace de recherche commun, chaque agent ayant une certaine capacité de mémorisation et de traitement de l'information. La règle de base était qu'il ne devait y avoir aucun chef d'orchestre, ni même aucune connaissance par les agents de l'ensemble des informations, seulement des connaissances locales. Un modèle simple fut alors élaboré [1]. Dès les premières simulations, le comportement collectif de ces agents évoquait celui d'un essaim d'êtres vivants convergeant parfois en plusieurs sous-essaims vers des sites intéressants. Ce comportement se retrouve dans bien d'autres modèles, explicitement inspirés des systèmes naturels. Ici, la métaphore la plus pertinente est probablement celle de l'essaim d'abeilles, particulièrement du fait qu'une abeille ayant trouvé un site prometteur sait en informer certaines de ses consœurs et que celles-ci vont tenir compte de cette information pour leur prochain déplacement.

2.2. Principe de l'algorithme

L'algorithme OEP est un processus de recherche à base de populations où les individus, désignés sous le nom de particules, sont groupés dans un essaim. Chaque particule dans l'essaim représente une solution candidate au problème d'optimisation. Dans un système OEP, chaque particule est « pilotée » dans l'espace multidimensionnel de recherche, ajustant sa position dans cet espace selon sa propre expérience et celle des particules voisines [2]. Une particule se sert, donc, de la meilleure position produite par elle-même et par ses voisins pour se déplacer vers une solution optimale.

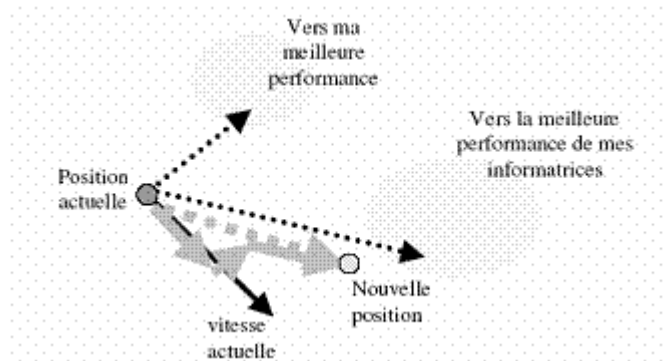


Figure 1 . Schéma de principe du déplacement d'une particule dans l'algorithme OEP

L'algorithme OEP s'exécute tant qu'un critère de convergence n'est pas atteint. Ce critère peut être: Un nombre fixe d'itérations; Une valeur de fonction de fitness nulle ou inférieure à un seuil ; Une vitesse proche de 0.

Le déplacement s'effectue suivant un trajet défini par une fonction de fitness (ou fonction objectif) qui encapsule les caractéristiques du problème d'optimisation.

Ainsi, chaque particule P est positionnée (aléatoirement ou non) dans l'espace de recherche du problème et est caractérisée par sa position courante $x(t)$ et sa vitesse courante $v(t)$.

On obtient la vitesse par la relation suivante [3] :

$$V(t+1)=v(t)+c1*(p(t) - x(t)) + c2*(g(t) -x(t)) \quad (1)$$

La vitesse $v(t+1)$ est mise à jour en additionnant la vitesse précédente de la particule, sa propre partie « cognitive » d'apprentissage $c1 * (p(t) - x(t))$ et sa partie « sociale » d'apprentissage $c2 * (g(t) - x(t))$.

La position d'une particule au pas de temps $t+1$ est la somme de sa position précédente $x(t)$ et sa vitesse actuelle $v(t+1)$.

$$x(t+1)=x(t)+vi(t+1) \quad (2)$$

Le choix des paramètres de contrôle $c1$ et $c2$ influe sur la convergence du processus. Diverses expériences ont montré que le choix de $c1 = c2 = 2$ donnait des valeurs optimales.

3. L'automate cellulaire

3.1. Définition générale

Un automate cellulaire est une grille régulière de **cellules** contenant chacune un **état** choisi parmi un ensemble fini des états et qui peut évoluer au cours du temps. L'état d'une cellule au temps $t+1$ est fonction de l'état au temps t d'un nombre fini de cellules appelé son **voisinage**. À chaque nouvelle unité de temps, les mêmes règles sont appliquées simultanément à toutes les cellules de la grille, produisant une nouvelle **génération** de cellules dépendant entièrement de la génération précédente [4].

3.2. Domaines d'application des automates cellulaires

Les applications pratiques des automates cellulaires sont nombreuses et diverses. Fondamentalement, ils constituent des univers dont on fixe les lois. Ces lois ne sont que partiellement connues et apparaissent hautement complexes. Dans un automate cellulaire, les lois sont simples et complètement connues. On peut ainsi tester et analyser le comportement global d'un univers simplifié. Parmi les domaines d'application des AC : en physique : simulation des processus de percolation et de cristallisation, étude des matériaux magnétiques selon le modèle d'Ising, en cryptographie, en mathématiques comme alternative aux équations différentielles, en électronique pour la conception d'ordinateurs massivement parallèles, simulation de la propagation des feux de forêt... En particulier, les applications des automates cellulaires en traitement d'images sont détaillées dans [6] [7].

4. La méthode proposée

Dans la littérature relative au domaine du traitement d'images, on trouve une multitude de méthodes de détection de contours qui s'appuient sur divers algorithmes. La plupart de ces méthodes se basent sur le calcul des dérivées directionnelles. D'autres méthodes abolissant le calcul des dérivées se basent plutôt sur des connaissances préalables de la nature et des caractéristiques de l'image, mais ces exigences limitent le champ d'application de la méthode. On tente ici, d'élaborer une méthode alternative pour la segmentation d'images, qui s'inspire des principes des systèmes complexes et de l'émergence. Elle utilise les automates cellulaires (ACs) comme modèle mathématique, qui est une représentation triviale de l'image. La puissance de ces ACs est décuplée par l'utilisation de l'algorithme OEP, qui leur fournit des aptitudes d'évolution et

d'adaptation. Cette hybridation de procédés donne un outil puissant, qui est les *Automates Cellulaires Evolutionnaires EvCA* [8], qu'on se propose d'appliquer au problème de détection de contour sur des images de différents types (binaires, à niveaux de gris, couleur).

La méthode peut être résumée comme suit :

La métaheuristique OEP est utilisée pour la recherche de la règle d'AC adéquate à la réalisation de la tâche de détection de contour. Pour ce faire, on utilise un ensemble d'images de référence, ainsi que leurs répliques segmentées. Ce sera le pool d'entraînement de l'OEP.

La fonction fitness mesure la qualité du processus de détection de contours, qui résulte de l'application de l'AC sur le pool d'images de référence. Le processus évolutionnaire commence avec un essaim initial aléatoire de règles d'ACs. Le résultat sera un ensemble évolué de règles de transition locales, qui définissent un AC détecteur de contours.

L'image représentant le contour de référence (vérité terrain ou ground-truth), est soit fabriquée à la main (hand-made), dans le cas où l'image source est une image simple (figures géométriques...), ou bien obtenue en association avec l'image testée, comme c'est le cas du benchmark de Berkeley qui fournit des paires d'images : source et ground-truth.

4.1. Format des règles

Le type de règle employé est dit règle totalistique, où l'état futur de la cellule centrale dépend uniquement de l'état des cellules voisines et non pas de leurs positions. Ce type de règles nous permet d'optimiser l'espace de recherche et le temps d'exécution.

Une cellule est dite morte si son état est égal à 0, et vivante sinon, c'est-à-dire que chaque cellule a deux états, le type de voisinage que nous allons utiliser est celui de MOORE (9 cellules).

On a le nombre de cellules vivantes qui varie de 0 à 9, et donne alors 10 valeurs possibles. Cette variable est notée NCV.

Le nombre de règle qu'on peut représenter est donc $2^{10} = 1024$ règles.

Exemple : on utilise la règle de transition 56 en binaire : 0000111000

Interprétation de cette règle est :

L'état futur de la cellule centrale est 0 si le nombre de cellules vivantes est $NCV \leq 3$ or $NCV \geq 7$. Dans les autres cas, son état futur sera égal à 1.

EF	0	0	0	0	1	1	1	0	0	0
NCV	0	1	2	3	4	5	6	7	8	9
DR	56									

EF : état futur, NCV : nombre de cellules vivantes, DR : règle de décision.

4.2. Evolution de l'Automate cellulaire par l'algorithme OEP

L'essaim est constitué par l'ensemble des règles de l'automate cellulaire. Chaque particule de l'essaim est définie par :

- Sa position : elle représente la règle de transition de l'AC. C'est une valeur codée sur 10 bits, variant de 0 à 1023.
- Vitesse : un nombre réel, initialisé à 0, il guide le processus vers la convergence.
- Fitness : c'est la fonction objectif qui mesure la qualité de l'image segmentée.

Le schéma suivant illustre le fonctionnement de notre algorithme :

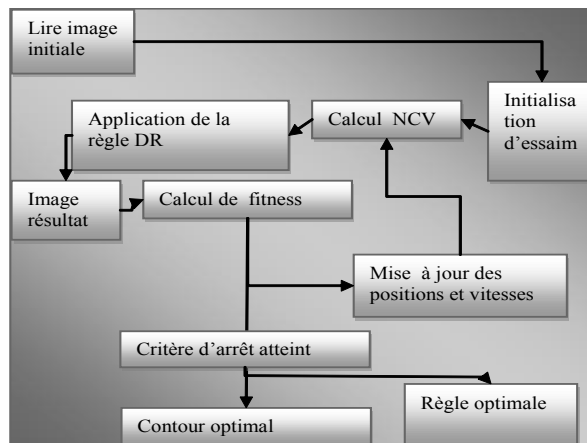


Figure 2. Etapes de traitement

4.3 La fonction Fitness

Pour évaluer la qualité de la segmentation réalisée par une règle, on applique d'abord l'AC, auquel on affecte l'image source, dont on veut extraire le contour, comme configuration initiale. On compare alors, le résultat obtenu après exécution, avec le contour étalon, et on mesure le taux d'erreur entre les deux images.

La fonction Fitness a une grande importance dans le processus évolutif de l'AC, guidé par l'algorithme OEP. Pour cela, on a choisi de prendre comme fitness une fonction, connue comme étant l'une des meilleures mesures de performance de la

segmentation par contour, c'est la fonction SSIM (Structural SIMilarity index) [9]. Pour deux images x et y , la fonction SSIM est calculée comme suit :

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

où μ_x , μ_y , σ_x^2 , σ_y^2 , σ_{xy} sont respectivement la moyenne de l'image x , la moyenne de l'image y , la variance de l'image x , la variance de l'image y , et la covariance des images x et y . les calculs précédents sont effectués entre deux pixels correspondants, entre les deux images. Suivant Wang et al [9], C_1 est initialisé à $(0.01 \times 255)^2$ et $C_2 = (0.03 \times 255)^2$.

5. Résultats expérimentaux

Dans ce paragraphe, on va évaluer le détecteur de contours réalisé en prenant en considération deux critères : La qualité visuelle des contours obtenus et la valeur de la fonction Fitness.

Pour cela, notre algorithme est testé sur des images de types différents (binaires, à niveaux de gris, réelles...). L'évaluation se fait par comparaison des résultats obtenus avec d'autres méthodes : des méthodes basées sur les automates cellulaires [10], [11] et des méthodes standards (détecteurs de Canny, Sobel, Prewitt, Roberts).

Le processus OEP a permis, par émergence, d'extraire une règle qui a donné les meilleurs résultats : c'est la règle 112. Le calcul de la fonction fitness SSIM a donné une valeur de l'ordre de 0.98, ce qui est excellent.

5.1. Résultats sur images de synthèse



Figure 3 . résultats de l'extraction de contours sur des images de synthèse (a) image originale (b) résultat [9] (c) résultat AC-OEP

On constate que les contours obtenus par [9] sont épais et entourés de bruit, alors que ceux de notre méthode sont nets, fins et sans bruit.

5.2. Résultats sur image réelle

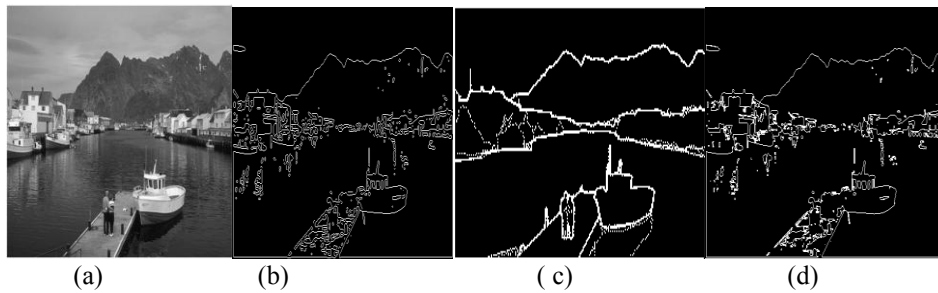


Figure 4. résultats de l'extraction de contours sur une image réelle (Benchmark Berkeley) (a) image originale (b) résultat Canny (c) contour étalon (d) résultat AC-OEP

On constate que notre méthode a permis d'extraire des contours corrects, fins, aucun faux contour ne figure sur l'image résultat, et que notre résultat est de performance égale ou supérieure à celui de Canny.

5.3. Résultats sur image médicale

On a testé notre algorithme sur des images médicales (IRM cérébrale). Le résultat est illustré ci-dessous :

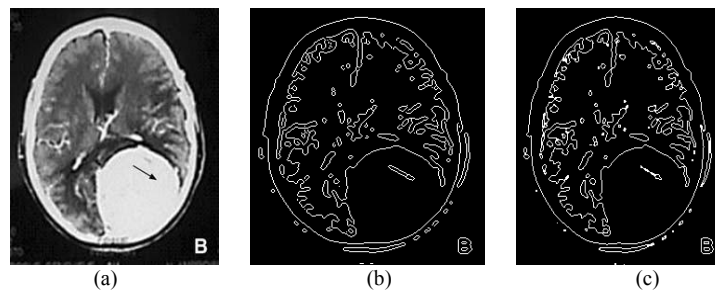


Figure 5. Résultats de l'extraction de contours sur IRM cérébrale (a) image initiale (b) résultat Canny (c) résultat AC-OEP

La même constatation que pour les images précédentes s'impose : malgré la complexité des images IRM, et les textures différentes qui les caractérisent, notre méthode a extrait les contours de manière correcte, en éliminant quelques bruits apparus dans le résultat de Canny. Les contours sont fins et continus.

5.4. Les valeurs de la Fitness

Dans le tableau ci-dessous, nous illustrons les valeurs de la fonction Fitness, obtenues entre les images originales ci-dessus, et leurs répliques ground-truth. Pour les images binaires simples, l'image contour référence est faite à la main (lapin,). Pour l'image ile, le contour référence est disponible sur le site du benchmark de Berkeley. Toutes les valeurs notées dans le tableau sont le résultat du calcul de la fonction SSIM entre l'image contour référence et respectivement l'image contour de Canny, la méthode Ev-CA et la méthode AC-OEP.

Table 1. Valeurs de fitness

Méthode	CANNY	EV-CA	CA-PSO
lapin	0.98775422	0.9726516	0.99547786
IRM	0.9858974		0.9974553
Ile	0.98392303		0.99944197

Dans tous les tests effectués, aussi bien que dans ceux illustrés ci-dessus (et d'autres non mentionnés dans ce papier), et sur divers types d'images, la méthode AC-OEP a prouvé son efficacité et sa robustesse face à d'autres méthodes connues de la littérature (Canny), ainsi qu'en comparaison avec d'autres méthodes basées sur les AC évolutionnaires. La méthode AC-OEP a engendré des résultats satisfaisants, et pourrait être améliorée pour traiter des images plus complexes (texturées...).

6. Conclusion

Dans ce travail, nous avons exposé une nouvelle approche de détection de contours basée sur un algorithme évolutionnaire qui tire son efficacité et sa puissance du paradigme : optimisation par essaim particulaire. Le phénomène d'émergence permet de faire apparaître une règle capable d'extraire des contours corrects et précis, sur différents types d'images. La qualité visuelle des résultats obtenus est appréciable ; le

temps de calcul est réduit en comparaison avec d'autres algorithmes. Ce travail ouvre des perspectives d'avenir pour explorer les capacités fascinantes des systèmes bio-inspirés et leurs applications dans divers domaines pratiques.

Bibliographie

- 1 J. Kennedy, R.C. Eberhart, Y. Shi, Swarm intelligence, San Francisco, *Morgan Kaufmann*, 2001.
- 2 Maurice Clerc, Patrick Siarry, «une nouvelle métaheuristique pour l'optimisation difficile: la méthode des essais particuliers »
- 3 Maurice Clerc « l'optimisation par essaim de particules. « tutoriel pour OEP 2003, carré des sciences, Paris, France, octobre 2003.
- 4 Ganguly N., Sikdar B., Deutsch A., Canright G., and Chaudhuri P., "A Survey on Cellular Automata," Centre for High Performance Computing, Dresden University of Technology, Tech. Rep. 9, 2003.
- 5 S.Wolfram , "A New Kind of Science", Wolfram media. ISBN 1-57955-008-8, 2002.
- 6 S. Djemame, O.Djidel, M.Batouche,"Image Segmentation Using Continuous Cellular Automata", proceeding of the IEEE 11th international Symposium of Programming and Systems ISPS'11, Algiers, Algeria, ISBN 978-1-4577-0905-0, pp 94-99.
- 7 S.Djemame, M.Batouche, "Image Segmentation Using an Emergent Complex System: Cellular Automata", proceeding of the IEEE 7th International Workshop on Systems, Signal Processing and Their Applications, Tipaza, Algeria, ISBN: 978-1-4577-0689-9, pp 207-210.
- 8 M.Mitchell, J.P.Crutchfield, "Evolving Cellular Automata with Genetic Algorithms: a Review of Recent Work", Proceedings of the First International Conference on Evolutionary Computation and its Applications, Moscow 1996
- 9 Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "[Image Quality Assessment: From Error Visibility to Structural Similarity](#)," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004.
- 10 Abbassene A., Meshoul S., Batouche M. "Segmentation d'Images par Emergence, Basée sur les Automates Cellulaires Evolutives », IEA/AIE, pp. 800–808 , 2006
- 11 Slatnia S., Batouche M., Melkemi K.E. « Evolutionary Cellular Automata Based-Approach for Edge-Detection », WILF 2007, LNAI 4578, pp. 404-411, 2007