

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITE DE M'SILA
FACULTE DE TECHNOLOGIE
DEPARTEMENT DE GENIE ELECTRIQUE

MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLÔME DE
MASTER EN GENIE ELECTRIQUE

SPECIALITE : AUTOMATIQUE

THEME

SEGMENTATION D'IMAGE PAR UNE APPROCHE CONTOUR ACTIFS

Proposé et dirigé par :

- Mr. HAMMOUCHE Abdelaziz

Présenté par :

- LAIFA Hamza

Année Universitaire : 2012/2013

N° d'ordre : 070

Remerciements

On remercie tout d'abord Allah, le tout puissant qui nous a facilité le chemin pour l'accomplissement de ce
Mémoire.

Nos grands remerciements vont à
Mr : HAMMOUCHE Abdelaziz.

Pour nous avoir et pour leur disponibilité et leurs précieux conseils.

A tous nos enseignants et le chef de département de
Génie Electrique

Mr : B.BEN DJAIMA

Qui nous ont aidés de proche ou de loin pour être des masters en électrotechnique option Automatique.

En fin

Nous tenons à exprimer notre reconnaissance à tous nos amis(es) et collègues pour le soutien moral et
Matériel.

Dédicace

*Avant tous, je remercie dieu le tout puissant de
m'avoir*

*Donner le courage et la patience pour réaliser ce
travail*

Je dédie ce modeste travail :

A mes très chers parents

A mes frères et ma sœur

A toute ma famille

A ma chère samrati

A tous les amis d'études surtout Ceux

****D'AUTOMATIQUE****

**Promotion 2013 **

LAIFA HAMZA

Tables de matière :

Table des matières	II
Table des figures et des tableaux.....	IV
Table des nomenclatures.....	V
Préambule	VI

Introduction Générale.....	1
----------------------------	---

I. Les méthodes de segmentation d'images

I.1. Introduction.....	4
I.2. Définition de la segmentation d'image.....	4
I.3. Système de traitement d'image.....	4
I.4. Approches de segmentation	5
I.4.1. Approches de Contours actifs	7
I.4.2. Approche région.....	10
I.4.3. Approche de type classification	13
I.5. Autres méthodes.....	15
I.6. critère de choix d'une méthode de segmentation	17
I.7. Conclusion	18

II. Segmentation d'image par la méthode de contour actif

II.1. Introduction	20
II.2. Définition d'un contour actif.....	20
II.3. Energies	21
II.3.1 Energie interne	21
II.3.2 Energie externe	22
II.3.3 Energie de contexte.....	24
II.4. Implémentations classiques	25
II.4.1 Différences finies.....	25
II.4.2 Approche variationnelle.....	27
II.4.3 Programmation dynamique.....	30
II.4.4 Algorithm « greedy »	32
II.5. Conclusion.....	34

III. Application, résultats et discussion

III.1. Introduction.....	32
III.2. Implémentation des contours actifs par l’algorithme de Greedy.....	32
III.3. Résultats.....	37
III.4. Influence et réglages des paramètres	41
III.5. Conclusion	48
Conclusion général	49
Références bibliographiques.....	50
Résumé.....	54

Liste des Figures

Figure I.1 schéma générale d'un système de traitement d'image	5
Figure I.2 les approches de segmentation.....	6
Figure I.3 Exemple de segmentation d'images	7
Figure I.4 Différents types de contours	8
Figure I.5 Dérivées premier et seconde d'un contour de type 'saut d'amplitude	9
Figure I.6 La segmentation par division - fusion.....	13
Figure I.7 Exemple d'histogrammes	15
Figure II.1 Exemple de coordonnées cartésiennes et abscisses curvilignes.....	21
Figure II.2 Variation du paramètre de gradient	23
Figure II.3 Variation du paramètre d'intensité.....	23
Figure II.4 importance du signe de l'énergie de ballon	25
Figure III.1 Exemple d'un voisinage 2.....	36
Figure III.2. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0.25, \delta=3$	38
Figure III.3. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0.25, \delta=3$	39
Figure III.4. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0.25, \delta=3$	40
Figure III.5. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0, \delta=3$	42
Figure III.6. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.65, \beta=5, \delta=3$	43
Figure III.7. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0.25, \delta=0$	44
Figure III.8. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0.5, \beta=0.25, \delta=9$	45
Figure III.9. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=0, \beta=0.25, \delta=3$	46
Figure III.10. Résultat de segmentation de l'image test cercle pour les paramètres.....	
$\alpha=5, \beta=0.65, \delta=3$	47

Introduction générale

Le traitement d'image voit son origine en physique des particules, dans les années 1950, lorsque les scientifiques cherchent à déterminer les trajectoires de particules en analysant des milliers d'images par expérience dans des chambres à bulles. Jusqu'à la fin des années 1960, les images acquises sont généralement de mauvaise qualité et très volumineuses. Leur traitement consiste, en premier lieu, à les restaurer pour corriger les défauts liés à l'acquisition; les améliorer pour rendre l'image plus belle à l'affichage; et les compresser pour réduire leur volume. Très vite, il est apparu nécessaire d'interpréter les images et l'analyse de scènes est devenue un aspect essentiel du traitement [01].

Comment comprendre les images, analyser leur contenu, stocker l'information utile, traiter le nombre grandissant d'images dans les diverses applications ?

Les premiers travaux d'analyse de scènes ont concerné l'extraction de primitives, comme les points d'intérêts de l'image. Mais pour comprendre les images, il faut avant tout reconnaître les objets présents dans celles-ci. Se pose alors la question de comment définir ces objets que l'on souhaite reconnaître. Par leur couleur, leur forme ? Est-ce suffisant ? Les premiers modèles de segmentation voient le jour. Initialement, les modèles proposés détectent les changements rapides de contraste dans l'image, les ombres et les lumières, les régions homogènes en intensité, couleur, ou texture. Mais ces informations qui sont fondées sur le niveau de gris d'un pixel, ou bien sur une texture locale ignorent l'information essentielle qu'est la géométrie de l'objet à extraire [01].

En effet, pour la tâche difficile qu'est la reconnaissance d'objets dans une image, les informations sur la couleur, le contraste, la texture, que nous qualifierons de photométriques, sont nécessaires, mais sont seulement capables de définir un objet simple dans un environnement simple. Si la scène de l'image devient complexe, ce qui est généralement le cas des images réelles, ces informations sont insuffisantes pour définir correctement l'objet recherché. Convaincu de la nécessité d'introduire dans les modèles des informations sur la forme de l'objet à détecter, il est cependant utile de réfléchir sur le type de «forme» que l'on souhaite définir dans le modèle [01]

Le travail qui nous à été attribué rentre dans le cadre de la reconnaissance de forme en utilisant la méthode des contours actifs, qui ont été largement utilisés en traitement d'image pour l'extraction d'objets. Introduite par *KASS et al* en 1985, dans *Proceedings of First International Conférence on Computer Vision*, et proposèrent, une nouvelle méthode pour la détection de contours dans une image. Ce contour est modélisé par une courbe qui se déforme en fonction de différentes énergies internes et externes pour venir se positionner sur les frontières de l'objet [02]. Le traitement s'effectuera sur les images avec un certain nombre de points chaînés, et dont le nombre varie en fonction de la précision désirée. Etant donné l'existence de différentes formes d'images, des coefficients sont attribués à chacune des énergies utilisées.

Dans ce travail, la fonctionnelle d'énergie qui représente l'ensemble des énergies, auxquelles sont appliqués leur coefficient respectif est minimisée grâce à l'utilisation de l'algorithme de Greedy. Cet algorithme est reconnu pour sa rapidité dans l'évolution du contour.

Ce travail est organisé en trois chapitres de la façon suivante : le premier chapitre contient généralité sur la segmentation d'images, dans lequel nous avons présenté quelques notions sur le traitement d'images et particulièrement les différentes méthodes de segmentation tel que les approches de type classification, les approches contour et les approches région. Le deuxième chapitre, sera consacré pour le développement de la méthode de segmentation par contours actifs «Snakes » et les méthodes d'implémentations dont les plus connues, les méthodes variationnelles, les méthodes par programmation dynamique et l'algorithme Greedy. Le chapitre dernier est réservé pour l'implémentation des contours actifs par l'algorithme de Greedy, ainsi que la discussion et interprétation des résultats. Et nous terminerons par une conclusion et quelques perspectives.

Chapitre I :

Les méthodes de segmentation d'images

I.1. Introduction :

Le traitement d'image est une technique qui a pour but l'analyse et la synthèse des images. L'analyse d'images et la reconnaissance de formes, ont pour objectif cherchent à extraire de l'image une fonction qualitative ou quantitative et à prendre des décisions en fonction des informations extraites [03].

Il existe plusieurs techniques de segmentation, chacune d'entre elles possède des avantages et des inconvénients et chaque technique est plus ou moins adaptée à un certain type d'images. Les techniques de segmentation se divisent en trois grandes approches : l'approche région l'approche frontière ou contour et l'approche de classification. Ces approches seront l'objet d'étude de ce chapitre.

I.2. Définition de la segmentation d'image

La segmentation d'image est un traitement de bas niveau qui consiste à créer une partition de l'image I en termes d'un prédicat d'homogénéité P et d'un ensemble de régions R_I vérifiant les critères suivants tel que [03]:

- ✓ $I = \cup_i R_i$
- ✓ $R_i \neq \emptyset \forall_i$
- ✓ $R_{ii} \cap R_j = \emptyset, \forall i, j; i \neq j$
- ✓ R_i est connexe $\forall i$
- ✓ $P(R_i) = vrai \forall i$
- ✓ $P(R_i \cup R_j) = faux \forall i \neq j$ et R_i et R_j sont adjacent

Donc une région est un ensemble connexe de points d'image appeler pixels ayant des propriétés communes tel que l'intensité, la texture, etc. qui les différencient des pixels des régions voisines.

I.3. Système de Traitement d'Image

Le traitement d'une image originale vers l'image résultat il passe sur plusieurs étapes comme expliquer dans la figure ci-dessus [03] :

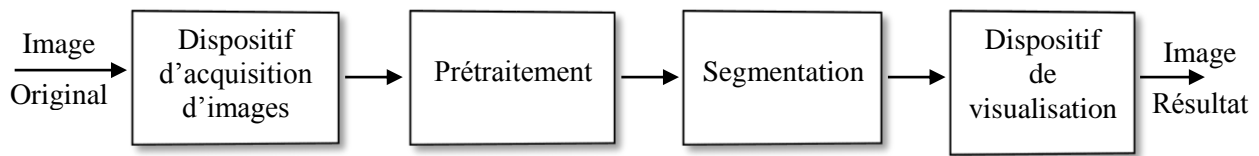


Fig.1.1. schéma générale d'un système de traitement d'image

Les différentes étapes du système de traitement d'image sont définies comme suit [03] :

Dans l'étape d'acquisition le signal analogique de l'image est converti en un tableau de valeurs numériques, chaque point est un pixel (Picture élément). La fonction est donc la numérisation de l'image en espace et en amplitude.

L'étape de prétraitements consiste à préparer les données de façon à faciliter leur traitement dans l'étape suivante (étape de segmentation). Il s'agit principalement de correction d'erreurs, de normalisation, de fenêtrage, etc.

L'étape de segmentation est une phase importante pour l'extraction d'informations principalement qualitatives de l'image.

Le Dispositif de visualisation doit répondre aux spécifications de l'utilisateur à savoir la présentation des résultats sous formes d'histogrammes, de diagrammes statistiques, d'images traitées, etc.

I.4 .Les méthodes de segmentation

Les techniques de segmentations existantes sont nombreuses mais elles sont généralement regroupées en trois principales approches. Ces approches sont comme indiquées à la figure suivante

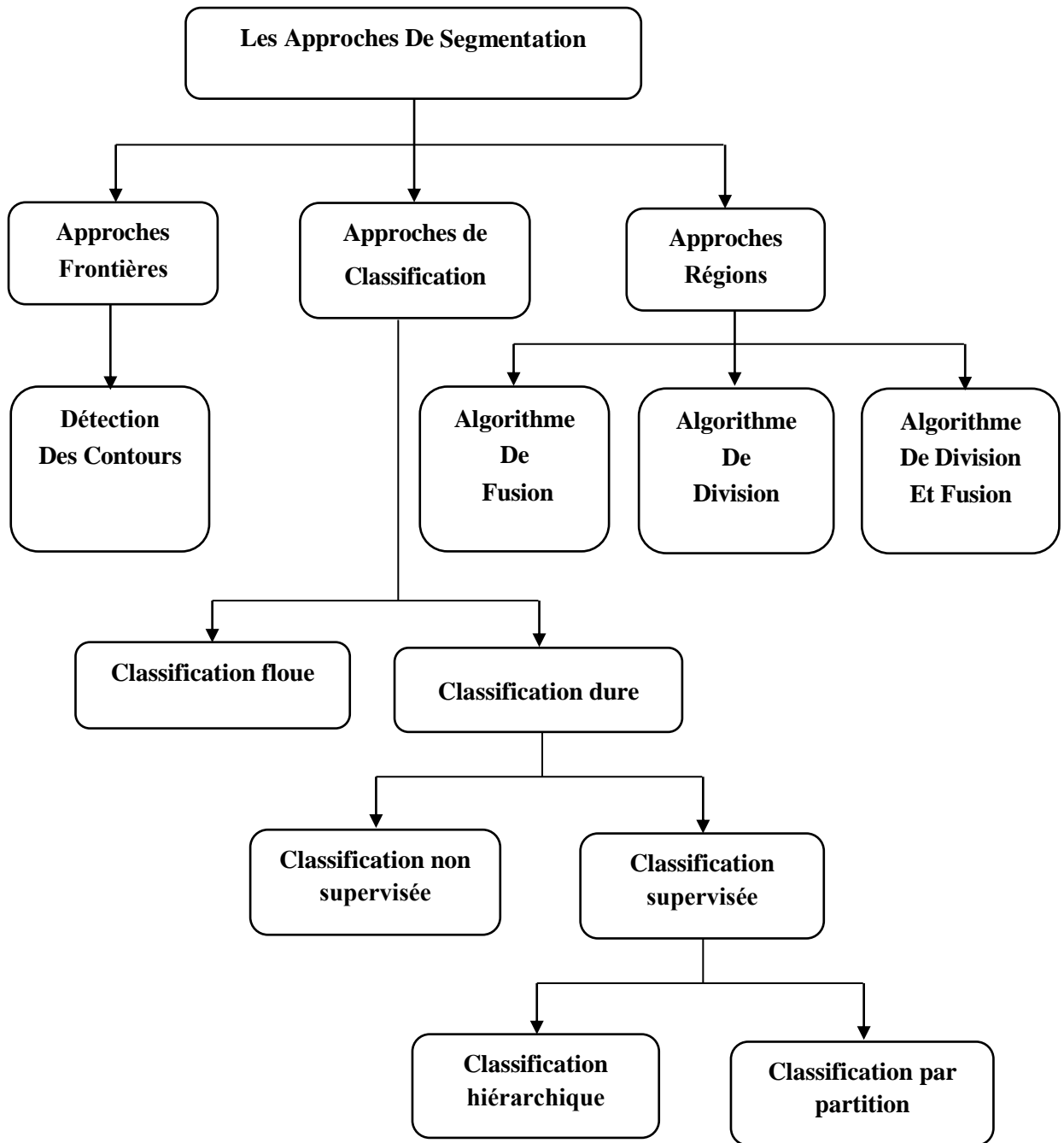


Fig.1. 2. Les approches de segmentation

Les méthodes de *l'approche contour* consistent à identifier les discontinuités du niveau de gris qui séparent les régions (fig. (I.3.b)).

Les méthodes de *l'approche région* cherchent à regrouper des pixels qui présentent une similarité et une uniformité en niveau de gris (fig. (I.3.a)). En dernier les méthodes de *classification* généralement considérée comme approche région, partitionnent l'image en des classes homogènes en considérant les propriétés colorimétriques des pixels. Les informations spatiales peuvent ou non être pris en considération.

Les approches contour et région sont duales l'une de l'autre, car une région homogène définit un contour, alors qu'un contour fermé délimite une région homogène. Cependant elles mènent à des algorithmes différents et ne donnent pas les mêmes résultats.

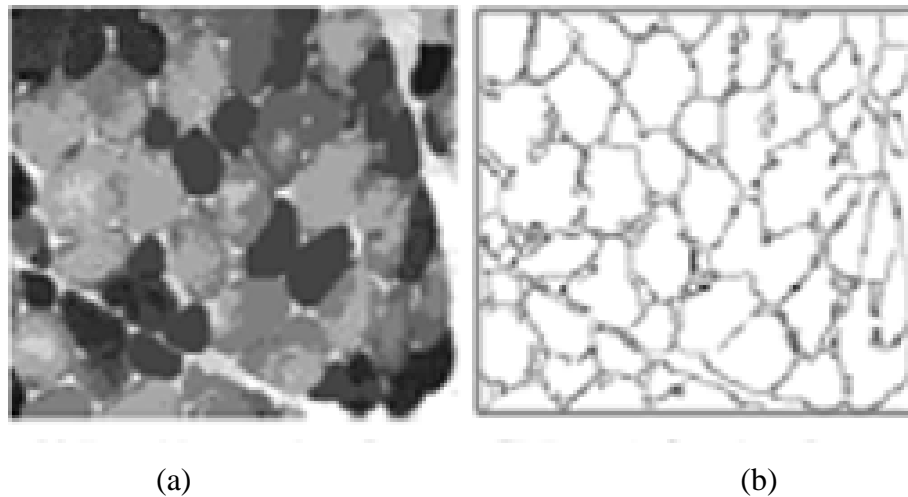


Fig.I.3. Exemple de segmentation d'images. *a) Segmentation en régions homogènes*
b) Segmentation en carte de contours

I.4.1. Approches de détection de Contours

Selon Pavlidis [04], un contour est une zone de transition séparant deux régions homogènes due à des discontinuités de profondeur et de réflectance de la scène observée. Ainsi un contour peut séparer deux régions ayant des intensités moyennes différentes. Il s'agit alors de contours de type saut d'amplitude, comme illustré sur la figure.4.3 a et b. Ou bien il peut également correspondre à une variation locale d'intensité présentant un maximum ou un minimum (figure.4.3 c et d).

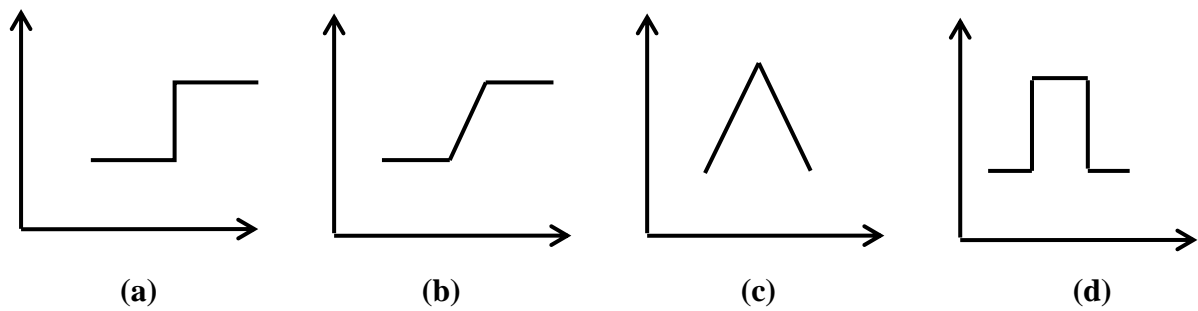


Fig.I.4. Différents types de contours

Nous présentons dans ce qui suit les principales méthodes de détection de contours.

a- Méthodes dérivatives

Les méthodes dérivatives sont les plus utilisées pour détecter des transitions d'intensité par différenciation numérique. Ce sont des méthodes locales qui balayent l'image avec un masque définissant la zone d'intérêt. A chaque position, un opérateur est appliqué afin de détecter les transitions significatives au niveau de l'attribut de discontinuité choisi.

De nombreuses techniques d'extraction de contours existent dans la littérature. Elles peuvent être divisées en deux approches :

- ✓ **Approche gradient** : calcul des premières dérivées partielles;
- ✓ **Approche laplacien** : calcul des deuxièmes dérivées partielles.

L'allure de la dérivée première et de la dérivée seconde est donnée par la figure (I.5). Une zone de transition dans le signal correspond à un maximum (ou un minimum) local de la dérivée première et un passage par zéro de la dérivée seconde. L'identification d'une zone de transition du signal peut être faite par seuillage de la norme de sa dérivée première ou du passage par zéro de sa dérivée seconde.

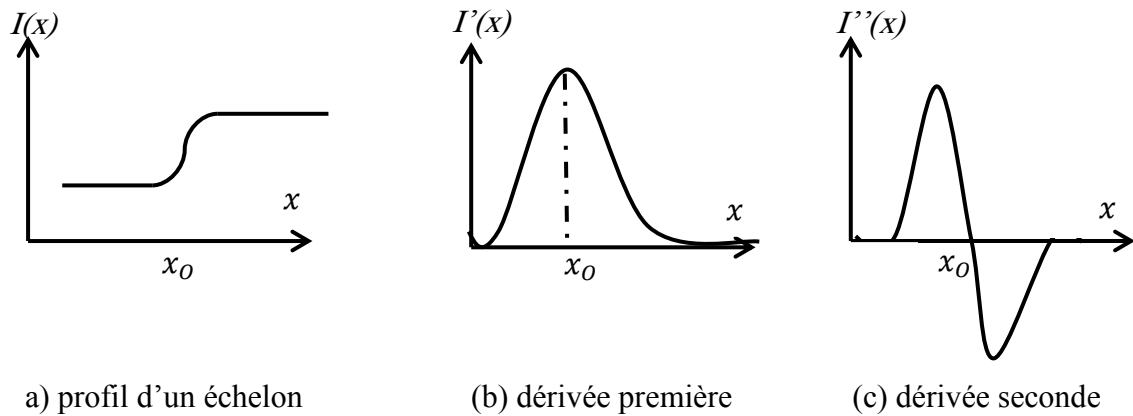


Fig.1.5. Dérivées première et seconde d'un contour de type "saut d'amplitude".

Parmi les opérateurs gradient, on trouve le masque de Robert, de Prewitt, de Sobel et l'opérateur de Kirsh. Ces opérateurs sont faciles à implémenter, rapides en temps de calcul et donnent de bons résultats pour des images non bruitées [05].

Les techniques exploitant le Laplacien possèdent un problème dû à leurs grandes sensibilités aux petites variations non significatives et qui sont dues essentiellement à la présence du bruit. Une manière de résoudre ce problème est d'utiliser un filtre gaussien.

Pour améliorer la qualité des contours et pallier aux problèmes de précision de localisation et d'efficacité de détection, sont apparus les opérateurs de dérivation avec filtrages optimaux. Le filtre optimal est un dérivateur qui permet de détecter des contours en respectant les trois critères suivants [05]:

- *Une bonne détection* : l'opérateur donne une réponse au voisinage d'un contour ;
- *Une bonne localisation* : optimisation de la précision avec laquelle le contour est détecté;
- *Unicité de la réponse* : le contour doit provoquer une réponse unique de l'opérateur.

Plusieurs opérateurs optimaux sont apparus dans la littérature. Parmi eux on trouve le filtre de Canny, le filtre de Deriche et le filtre de Shen et Castan [05].

D'autres étapes sont nécessaires pour passer des points de contours à une liste de contours exploitables par un processus d'interprétation. Très souvent la détection des points de contours est suivie d'une étape de prolongement et de chaînage de ces points et un codage des chaînes obtenues par des segments de droites [06].

b- Autres méthodes

D'autres approches d'extraction de contours font appel à d'autres modèles de l'image. Nous en citons :

- ✓ **Les approches surfaciques** : L'image des intensités est considérée comme une surface. Dans ce cadre, la transition entre deux régions est modélisée par un modèle surfacique. Le contour est présent quand il y a une bonne corrélation entre le modèle surfacique et une zone de l'image [07].
- ✓ **Les approches contours actifs** : Les contours actifs ou « Snakes » ont été introduits par Kass et al. [2]. Ils se présentent comme un modèle pour l'extraction des caractéristiques visuelles dans une image comme les contours d'objet ou les éléments de frontières. L'idée de base est de positionner, au voisinage du contour à détecter, une courbe qui sera l'initialisation du contour actif et de la déformer successivement jusqu'à ce qu'elle coïncide avec la frontière de l'objet.

I.4.2 Approches de détection de région homogène

Contrairement à la segmentation par contours dont le principe est la recherche des points essentiels qui donnent la forme des objets composant l'image, la segmentation en régions consiste à décomposer l'image en des régions homogènes. Dans ce type de méthodes de segmentation, en plus de l'information niveau de gris, l'information spatiale est automatiquement intégrée dans le processus de segmentation pour la formation de régions connexes et homogènes au sens d'un prédicat prédéfini [08],[09],[04]. Le contrôle géométrique peut être aussi utilisé pour contrôler le processus de segmentation [10]. Le problème fondamental associé à cette catégorie d'approches est justement lié à la définition du critère d'homogénéité utilisé dans le processus de segmentation de l'image.

On distingue dans cette classe trois grandes familles de techniques :

- ✓ **Les méthodes de fusion de régions ;**
- ✓ **Les méthodes de division ;**
- ✓ **Les méthodes de division/fusion.**

a) Les méthodes de Fusion de régions

Ces méthodes procèdent par agrégation (regroupement) itérative des pixels ou de régions obtenues par une simple technique de croissance de régions ou par une technique de

division. L'algorithme s'arrête quand une segmentation optimale est atteinte au sens d'un prédicat d'homogénéité portant sur les caractéristiques des régions adjacentes.

✓ *Agrégation de pixels*

Le point de départ est le choix d'un ensemble de pixels appelés « germes ». A chaque germe on associe un vecteur de caractéristiques. Chaque germe est agrégé avec le premier pixel connexe qui possède un vecteur de caractéristiques suffisamment proche du sien [11], [12]. Le processus d'agrégation est itératif de sorte que chaque région croît pixel par pixel. Divers vecteurs de caractéristiques et diverses mesures de similarité ont été proposés dans la littérature [13]. Généralement le vecteur de caractéristique est réduit à la valeur de niveau de gris et la distance euclidienne est utilisée comme mesure de similarité. Des propriétés statistiques telles que la moyenne ou la variance peuvent aussi être utilisées [14].

Les régions ainsi construites dépendent fortement du choix des germes de départ ainsi que de l'ordre de parcours des pixels agrégés. La plupart des algorithmes de croissance de régions parcourent l'image de façon prédéterminée de haut en bas et de gauche à droite [15], [16], [17].

✓ *Regroupement itératif d'ensemble de pixels*

Dans cette méthode, de petites régions-germes sont générées (elles peuvent se réduire à un pixel). Un graphe d'adjacence de régions est construit tel que chaque nœud représente une région et chaque arête représente une adjacence entre deux régions. Un coût de fusion est associé à chaque couple de régions adjacentes. La méthode consiste alors à déterminer à parti de ce graphe pondéré, les meilleures fusions qui minimisent ce coût. Cette méthode est facile à implémenter mais pose certains problèmes tels que : le choix du seuil de similarité: un seuil trop faible crée une sur-segmentation et s'il est trop fort conduit à une sous-segmentation [18].

Il existe d'autres représentations utilisées pour la segmentation d'image qui sont pyramidales parmi on peut mentionner la pyramide irrégulière de Bertolino [19] qui utilise une information de gradient lors de l'évaluation de la fusion des régions du graphe d'adjacence.

b) Les Méthodes de type division (Split)

Dans ce type de méthodes, l'image est divisée d'une manière récursive tant que le

critère d'homogénéité n'est pas vérifié. Si le critère est vérifié l'algorithme s'arrête, sinon l'image est divisée en d'autres zones, ou chaque zone est testée et redivisée si elle ne valide pas le critère. L'algorithme se termine lorsque toutes les régions sont homogènes ou bien leur taille est en dessous d'un seuil minimal fixé. La division de l'image est réalisée selon une structure géométrique. Citons deux structures possibles :

- ✓ **L'arbre quaternaire (structure Quadtree)** : dans cet arbre chaque nœud correspond à un bloc (zone de l'image de forme carrée) et possède exactement quatre nœuds fils, excepté les nœuds terminaux. L'algorithme de division consiste à tester l'homogénéité de l'image originale. Si elle n'est pas homogène, elle est récursivement découpée en des zones carrées jusqu'à ce que chaque carré soit homogène. Cette méthode présente deux inconvénients majeurs qui sont la non invariance en translation et sa rigidité.
- ✓ **Structure de Voronoi** : Au départ des germes initiaux sont choisis afin de construire les polygones de Voronoi. Ensuite de nouveaux germes sontensemencés d'une manière itérative à l'intérieur de chaque polygone non homogène, modifiant ainsi le diagramme de Voronoi. Le processus d'ensemencement de germes se poursuit jusqu'à ce que tous les polygones soient homogènes ou bien que leurs tailles atteignent un seuil préfixé. Une des différences par rapport à la structure précédente est que lorsqu'un polygone est partitionné, la topologie de polygones voisins change aussi, s'adaptant à ce nouveau partitionnement.

Beaucoup d'algorithmes de division consistent à tester l'homogénéité des régions en utilisant l'histogramme de niveau de gris. Les différents pics et vallées de l'histogramme sont recherchés et les régions sont alors définies à partir des intervalles entre les vallées.

c) Les méthodes de type division/fusion

Ces méthodes combinent les deux méthodes décrites précédemment, la division de l'image en des petites régions homogènes, puis la fusion des régions connexes et similaires au sens d'un prédicat de regroupement. Le processus de segmentation utilise la structure pyramidale du Quadtree ou la structure de voronoi pour la division de l'image jusqu'à la satisfaction du critère d'homogénéité. Ensuite, chaque fois que quatre sous-blocs satisfont un critère d'homogénéité de fusion ils sont regroupés à un niveau supérieur du Quadtree. La fusion continue tant que le critère d'homogénéité est satisfait (Figure I.6).

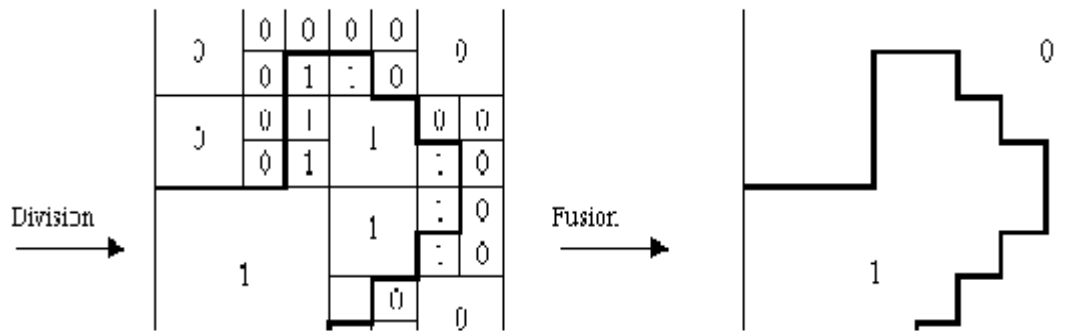


Fig.I.6. La segmentation par division-Fusion.

Un autre exemple de division/fusion est l'algorithme développé par Beveridge et al [20]. L'image est d'abord découpée en de petites fenêtres rectangulaires qui se chevauchent afin d'éviter des problèmes dus aux frontières des régions.

La division est ensuite appliquée d'une façon parallèle pour chacune de ces fenêtres en utilisant les modes et les vallées de leur histogramme de niveaux de gris. Une région est créée pour chaque mode de l'histogramme tout en tenant compte de la connexité des pixels. Les régions suffisamment proches au sens d'un critère (choisi selon la moyenne des niveaux de gris des deux régions, de leur taille et de la longueur de leur contour commun) sont ensuite fusionnées. L'inconvénient de cet algorithme est l'introduction d'un nombre important de seuils pour effectuer la segmentation.

I.4.3 Les méthodes de classification

Ces méthodes de segmentation d'images définissent une partition de l'image en un ensemble de classes (chaque classe rassemble des pixels ayant des vecteurs de caractéristiques) distinctes les unes des autres [21].

Généralement, ces méthodes de classification ne prennent pas en compte la disposition spatiale des pixels et ne considèrent que le vecteur de caractéristiques utilisé. L'identification des classes de pixels s'effectue en affectant à chaque pixel une étiquette indiquant la classe à laquelle il appartient. Ainsi les pixels appartenant à une classe peuvent former plusieurs régions non adjacentes dans l'image mais partageant les mêmes propriétés statistiques. Une segmentation en régions n'est obtenue qu'après analyse de la connexité des pixels dans l'image étiquetée. Les méthodes de classification peuvent être divisées en deux grandes méthodes *dures* et *floues* :

- a) **Les méthodes de classification floue** : dans laquelle on affecte au pixel un degré d'appartenance (indique la probabilité qu'un pixel y appartienne) pour chaque classe de la partition. La classe finale du pixel sera celle pour laquelle son degré d'appartenance est le plus élevé.
- b) **Les méthodes de classification dure** : dans laquelle un pixel est affecté qu'à une seule classe dans la partition de l'image et ce divise en deux méthode :
- ✓ **Méthodes de classification supervisée**, dans laquelle on dispose d'un ensemble de pixels étiquetés d'apprentissage ce qui implique un nombre de classes connu ; puis on associe chaque nouveau pixel à la classe la plus adaptée en se servant des pixels étiquetés.
 - ✓ **Méthodes de classification non supervisée**, L'objectif est de pouvoir regrouper automatiquement des pixels similaires dans une même classe sans la connaissance du nombre et du contenu des classes possible. Pour cela on défini une fonction de similarité entre pixels qui sera maximum entre les pixels d'une même classe et minimum avec ceux des autres classes. On dispose aussi de deux types de méthodes :
 - **Les méthodes hiérarchiques** qui produisent une hiérarchie de classes, telle que plus on descend dans la hiérarchie plus on trouve des classes homogènes. A l'opposé, **les méthodes de partitionnement** déterminent une seule partition.

Les méthodes de classification se diversifient par la fonction de similarité qui est la distance utilisée pour regrouper les pixels de l'image en des classes, la fonction objective utilisée pour évaluer la qualité de la classification obtenue et selon la stratégie de regroupement et de parcours de l'espace des classes possibles. Le processus de classification est comme suit [22]:

- Représentation des données de la classification (extraction des attributs des pixels tels que le niveau de gris, paramètres de texture, couleur, etc.) ;
- Définition d'une distance de similarité entre les pixels, tel que la distance euclidienne entre les niveaux de gris des pixels ;
- Regroupement des pixels en des classes ;
- Evaluation de la classification obtenue.

La stratégie de regroupement peut être : **Ascendante versus descendante** dans laquelle la méthode ascendante démarre avec un nombre important de classes (qui se réduit au nombre

de pixels de l'image) puis fusionner les classes proches selon un processus récursif jusqu'à ce que le critère d'arrêt soit vérifié. A l'inverse, la méthode descendante part d'une seule classe regroupant tous les pixels qu'elle divisera successivement en de nouvelles classes jusqu'à ce que le critère d'arrêt soit vérifié. Ou bien **Déterministe versus stochastique** : Cette propriété concerne en particulier les méthodes de classification par partitionnement qui tentent d'optimiser la fonction objective mesurant la qualité de la partition obtenue, qui peut être accomplie en utilisant des techniques traditionnelles ou à travers une recherche aléatoire de l'espace de solutions contenant tous les étiquetages possibles.

La détermination du nombre de classe peut être effectuée par **la méthode de Seuillage d'histogramme** qui se base sur l'étude de la distribution des niveaux de gris des pixels [01].

Dans le cas, et d'un fond visuellement différent, l'histogramme est bimodal (figure I.7.a). Par contre l'histogramme est multimodal (figure I.7.b), si l'image est constituée de régions ayant des caractéristiques différentes, ou chaque mode correspond à une région.

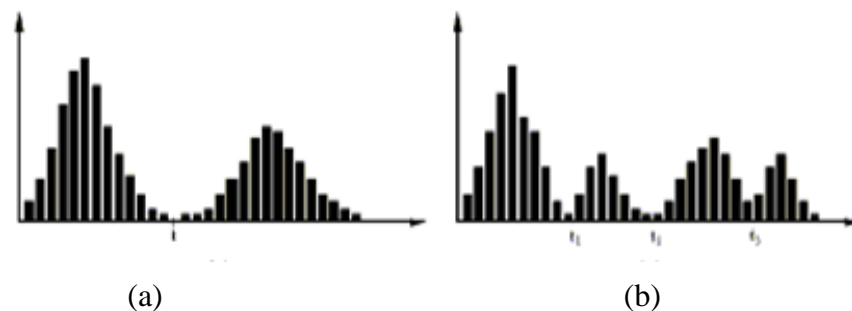


Fig.I.7. Exemples d'histogrammes. (a) Histogramme bimodal. (b) Histogramme multi-modal

D'une manière générale, le choix de seuils d'histogramme peut se faire de deux manières différentes. Soit les seuils sont déterminés globalement pour tous les points de l'image, soit les seuils sont calculés localement à partir de petites fenêtres de focalisation.

I.5. Autres méthodes de segmentation

Dans la littérature il existe des travaux qui portent sur la combinaison de deux techniques vue précédemment, on se servant des avantages de chacune d'elle pour remédier à leurs inconvénients, comme il existe aussi des méthodes basées sur le comportement sociaux

des individus, ou des organes des êtres vivant, tel que :

I.5.1. Segmentation par les réseaux de neurones

Les réseaux de neurones ont été utilisés aussi bien pour la détection de contours que pour la classification de pixels. Par exemple on peut citer les travaux de Cortes [23], ceux de Babaguchi [24] qui utilise un réseau de neurones multicouches pour l'extraction des points de contour. Dans ce type de réseau, chaque cellule est connectée à celles situées à une certaine distance par rapport à elle. Cette cellule peut être inhibée par celle de son voisinage. Dans [25], les auteurs présentent une série de tests effectués sur plusieurs types de réseaux de neurones effectuant le filtrage d'une image en éliminant le bruit qu'elle contient. Un état de l'art sur les méthodes basées réseaux de neurones pour la détection de contours peut être trouvée dans [26].

I.5.1.b Segmentation d'images par les algorithmes génétiques

Les premiers travaux utilisant les Algorithmes Génétiques pour résoudre le problème de classification sont dus à Raghavan [27]. L'algorithme démarre avec un nombre de classes K fixé à l'avance et une population de chromosomes de longueur N (taille de l'image à segmenter). Chaque chromosome associe une classe à un pixel. Les opérateurs génétiques, utilisés pour générer à chaque génération une population de partitions possibles, sont une adaptation des opérateurs génétiques binaires. Le but de l'algorithme est de minimiser une fonction fitness représentant l'inverse de la variance intra classe.

I.5.1.c Segmentation par les systèmes multi-agents

Les Systèmes Multi-Agents (SMA) sont apparus au début des années 80 et se sont développés à partir de l'organisation de la vie et de la société. Un SMA est un ensemble d'agents autonomes qui travaillent ensemble de façon coopérative ou parallèle pour résoudre un problème en s'appuyant sur les capacités et les connaissances individuelles de chaque agent [05]. Les systèmes multi-agents offrent une architecture permettant de faire travailler plusieurs entités autonomes sur un même problème (en coopération ou en parallèle) à l'aide de protocoles de communication et de processus d'échange d'informations. Le paradigme Multi-agents a été largement utilisé avec succès ces dernières années dans le domaine de traitement d'images en général et dans la segmentation en particulier.

Un système multi-agent (SMA) est défini dans [28] comme un système composé des éléments suivants :

- ✓ un **environnement** E, c'est à dire un espace disposant généralement d'une métrique ;
- ✓ un ensemble d'**objets** O. Ces objets sont situés, c'est à dire que pour tout objet, il est possible, à un moment donné, d'associer une position dans E. Ces objets sont passifs, c'est à dire qu'ils peuvent être perçus, créés, détruits et modifiés par les agents ;
- ✓ un ensemble A d'**agents**, qui sont des objets particuliers ($A \subseteq O$), lesquels représentent les entités actives du système. Ces agents possèdent les propriétés spécifiées plus haut;
- ✓ un ensemble de **relations** R qui unissent des objets (et donc des agents) entre eux ;
- ✓ un ensemble d'**opérations** Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O ;
- ✓ des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers ;

I.6. Critères de choix d'une méthode de segmentation :

Malgré la diversité des techniques de segmentation d'images, il n'existe pas de technique universelle pour toutes les images existantes. Toutes les méthodes étant valables, mais chacune donne un résultat plus ou moins acceptable. De se fait, pour choisir une méthode plusieurs critères de choix s'imposent ; on cite par exemple :

- ✓ La nature de l'image (texturée, présence de bruit, etc.).
- ✓ Primitives à extraire (contours, régions, etc.).
- ✓ Opérations situées en aval de la segmentation (mesures, interprétations, etc.).
- ✓ Contraintes d'exploitation (fonctionnement en temps réel, taille mémoire disponible, complexité algorithmique, etc.).

Reste à dire que la méthode de segmentation dépend essentiellement de la nature de l'image à segmenter, à partir de laquelle on veut distinguer ses différents éléments, pour savoir exprimer une meilleure interprétation possible de l'image [03].

I.7. Conclusion

Les méthodes de segmentation présentées dans ce chapitre, présentent quelques inconvénients. L'approche contour, par exemple, nécessite souvent un algorithme de fermeture du contour. L'approche région est, quant à elle, généralement plus gourmande en calculs.

L'approche que nous avons choisie dans ce travail repose sous le nom de contours actifs, proposées depuis 1987. Ces contours actifs permettent d'obtenir un résultat, en partant d'un contour composé de points chaînés a priori et utilisent des connaissances se bornant simplement à des propriétés de régularité et de continuité, et fera l'objet d'étude de prochain chapitre.

Chapitre II :

Segmentation d'image par la méthode de contour actif

II.1.Introduction

Un modèle de contour actif, est une structure dynamique utilisée en traitement d'image introduit de manière formelle par Kass et Witkin en 1987. Ce modèle est formé d'un ensemble de points répartis sur une courbe placée autour d'un objet à segmenter sur une image. Plusieurs équations décrivent son évolution. La courbe se déplace et épouse lentement les contours de l'objet en fonction de divers paramètres comme l'élasticité, la rigidité et la tolérance au bruit, etc.

Dans l'approche de segmentation par contour actif, la dynamique est basée sur la notion d'énergie interne et externe, le but étant de minimiser l'énergie totale présente le long de la courbe.

Dans la première section de ce chapitre, une étude des contours actifs est effectuée, dans la deuxième section les différentes méthodes de son implémentation seront développées.

II.2. Définition d'un contour actif

Les contours actifs sont définis par une courbe continue, fermée ou non, à extrémités fixes ou non. Ils se déforment à partir d'une position d'initialisation située près de l'objet d'intérêt. Le modèle est soumis à des forces qui le déforment et le déplacent dans l'image. L'énergie du modèle comprend un terme d'énergie interne de régularisation ou de lissage qui correspond à la fonctionnelle de Tikhonov et un terme d'énergie externe ou d'adéquation aux données, auquel des contraintes en terme d'énergie peuvent être ajoutées.

Le contour est défini par une représentation paramétrique.

$$V: \Omega = [0,1] \rightarrow \mathbb{R}^2 \quad (\text{II.1})$$

Soit C une courbe déformable spatialement et temporellement [10]. Le domaine de définition de la courbe est normalisé à $[0,1]$ et le paramètre est l'abscisse curviligne. Avec s le paramètre de position spatiale le long de la courbe dans l'image et t l'évolution temporelle de la courbe dans l'image la courbe se représente comme sur la Figure II.1.

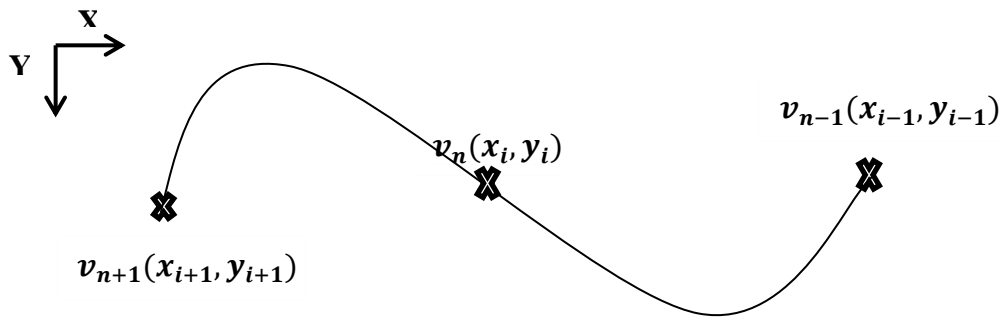


Fig.II.1. Exemples de coordonnées cartésiennes et abscisses curvilignes d'un contour

Soient $v(s, t)$ la position d'un point de la courbe à un instant t et x, y les coordonnées cartésiennes d'un point de l'image.

$$V(s, t) = (x(s, t), y(s, t))^t \quad \forall (s, t) \in [0, 1] * [0, +\infty] \quad (\text{II.2})$$

Les contours actifs permettent de résoudre un problème par minimisation d'une fonctionnelle d'énergie. Dans ce qui suit nous examinons les différentes énergies utilisées dans les contours actifs.

II.3. Energies

La fonctionnelle d'énergie attachée au contour actif est composée de trois énergie

$$\Phi(v): v \rightarrow E_{interne}(v) + E_{externe}(v) + E_{contexte}(v) \quad (\text{II.3})$$

Nous allons préciser ces trois types d'énergies.

II.3.1. Energie interne

L'énergie interne gère la cohérence de la courbe. Elle maintient la cohésion des points et la raideur de la courbe. C'est le modèle physique type « plaque - membrane mince » qui est utilisé comme terme de régularisation, c'est-à-dire les deux premiers termes régularisant de Tikhonov [29].

$$E_{interne}(v) = \int_0^1 \frac{\alpha}{2}(s) \|v'(s)\|^2 + \frac{\beta}{2}(s) \|v''(s)\|^2 ds \quad (\text{II.4})$$

Les termes v' et v'' sont les dérivées première et seconde de v par rapport à s .

Le terme du 1er ordre correspond à la tension (comportement comme une membrane mince). Il prend une valeur importante quand la courbe se distend.

Lorsque $\alpha=0$ la courbe peut présenter des discontinuités. Nous parlerons donc d'énergie de continuité

Le terme du 2^{ème} ordre correspond à la courbure (comportement comme une plaque mince). Il prend une valeur importante lorsque la courbe s'incurve rapidement c'est-à-dire pour l'obtention de coins. Lorsque $\alpha = 0$, la courbe peut prendre une forte convexité, par contre lorsque α est grand, la courbe tendra vers un cercle si elle est fermée ou une droite si elle est ouverte.

II.3.2. Energie externe

L'énergie interne gère la régularisation du contour actif, par contre l'énergie externe correspond à l'adéquation aux données et prend en compte les caractéristiques de l'image. Rappelons ici que ce sont les contours de formes qui sont recherchés donc les points de fort gradient ou des points ayant une propriété de position par rapport à une couleur donnée.

✓ Gradient

Pour la recherche des zones de fort contraste dans l'image, est introduite la fonction :

$$E_{externe} = - \int_0^1 \|\nabla I(v(s))\|^2 ds \quad (II.5)$$

Où $\nabla I(v(s))$ représente le gradient de l'image I en $v(s)$. Très souvent, c'est le gradient gaussien qui est utilisé :

$$E_{externe} = - \int_0^1 \|\nabla(g_\sigma * I)v(s)\|^2 ds \quad (II.6)$$

Où g_σ est la gaussienne centrée d'écart type σ .

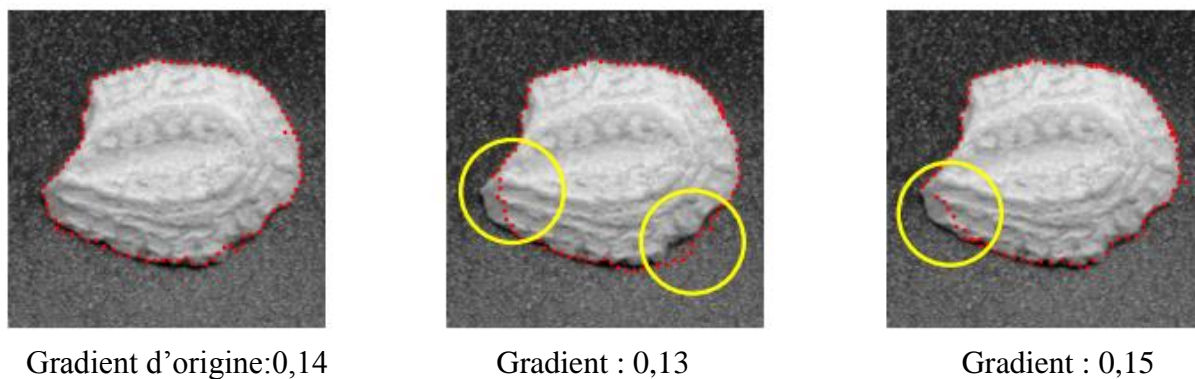


Fig. II.2. Variation du Paramètre du gradient

Cette énergie, au contraire, permet de sélectionner les zones sombres ou claires selon le signe choisi.

$$E_{intensite} = \pm \int_0^1 (I(v(s)) - i_0)^2 ds \tag{II.7}$$

La valeur i_0 introduit ou non, un certain seuillage. On peut ainsi favoriser la position du contour dans une zone donnée.

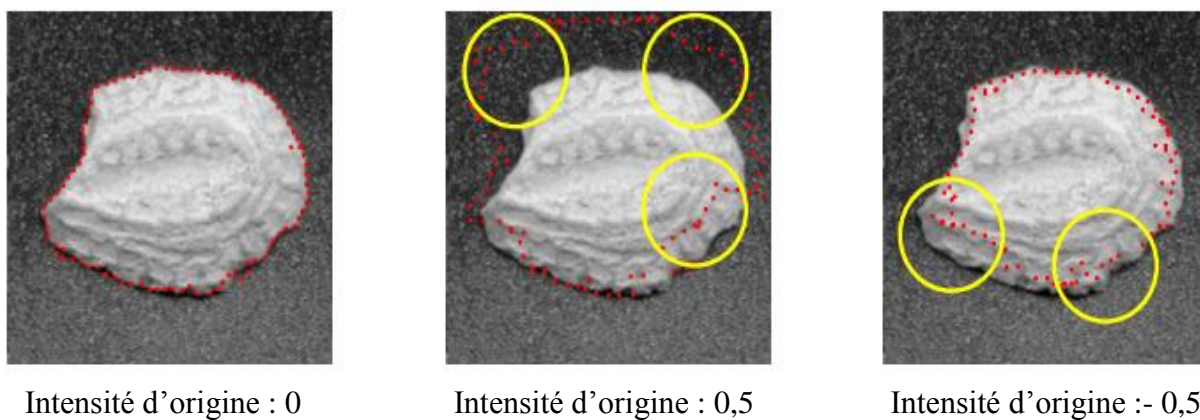


Fig.II.3. variation du paramètre d'intensité

✓ Gradient vector - flow

Cheniang Xu constatant la médiocrité de la qualité de la convergence de la courbe de contour actif vers le contour souhaité dans les zones à forte concavité, introduit un nouveau potentiel . Il s'agit d'une nouvelle force ex-terne qui traduit la diffusion isotopique d'un flux externe.

Il définit le champ "Gradient Vector Flow", le « GVF » comme le champ de vecteurs :

$$V(x, y) = [u(x, y)v(x, y)]^t \quad (\text{II.8})$$

Qui minimise la fonctionnelle d'énergie

$$\varepsilon(u, v) = \varepsilon = \iint_I \mu \cdot (u_{x^2} + u_{y^2} + v_{x^2} + v_{y^2}) + \|\nabla(\|\nabla I\|)\|^2 \cdot \|v - \nabla(\|\nabla I\|)\|^2 dx dy \quad (\text{II.9})$$

Xu et Prince proposent une généralisation de GGVF (*Generalized Gradient Vector Flow*).

Ce nouveau potentiel, d'un intérêt certain lorsque l'objet à segmenter est unique, peut poser problème dans le cas d'objets multiples dans des images réelles, la diffusion du gradient pouvant créer des interférences entre les zones d'influence des différents objets. Le temps de calcul de ce GVF peut aussi représenter un frein à cette méthode. Xu annonce, pour une image de 256 x 256 en 256 niveaux de gris, un temps de 2 s en Matlab. Delmas, en le transposant sous Linux, trouve 0,2 s par itération

II.3.3. Energie de contexte

L'énergie de contexte, parfois appelée énergie de contrainte, permet d'introduire des connaissances *a priori* sur ce que nous cherchons. Entre autres, nous plaçons, sous cette rubrique, l'énergie ballon introduite par Laurent D. Cohen.

La tendance naturelle des *snakes*, est la rétraction. La minimisation de l'énergie implique une minimisation de distance. La force ballon va tendre à gonfler le contour actif ou accélérer sa rétraction selon le signe de la force introduite.

De plus, cette force va permettre de dépasser les contours présentant un faible gradient et ainsi de sortir du bruit pour atteindre une frontière plus fortement marquée. Il s'agit d'une force normale au contour en chaque point.

$$F_{ballon}(v_s) = K\vec{n}(s) \quad (\text{II.10})$$

Où $n(s)$ est un vecteur unitaire normal à la courbe en $v(s)$.

L'intensité de l'énergie ballon est un scalaire généralement négatif (expansion du ballon) proportionnel à l'aire intérieure du contour. Elle devient :

$$E_{ballon} = \pm \delta \iint_{\Omega} dx dy \quad (\text{II.11})$$

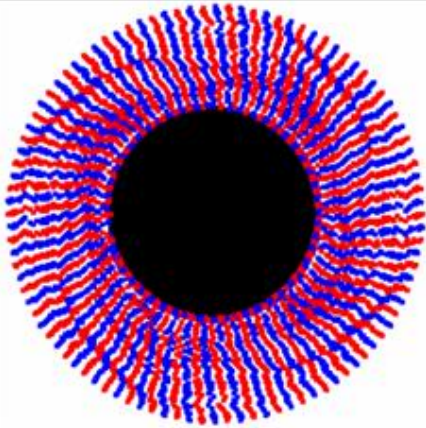
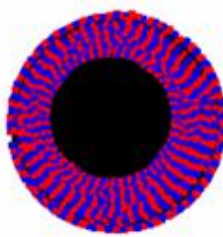
Energie de ballon positive	Energie de ballon Négative
 A large black circle surrounded by a dense ring of red and blue wavy lines. The snake is contracting towards the center.	 A smaller black circle surrounded by a dense ring of red and blue wavy lines. The snake is expanding outwards.
Le snake va contracter	Le snake va gonfler

Fig.II.4. importance du signe e l'énergie de ballon

C'est à partir de ces différentes énergies et de leur combinaison que l'on peut définir un problème dont la solution est le contour recherché. La mise en œuvre d'une telle approche a donné lieu à de nombreuses implémentations.

II.4. Implémentations classiques

A partir de cette approche théorique, trois implémentations principales peuvent être recensées : l'approche vibrationnelle classique peut-être la plus développée et la plus déclinée introduite par Kass, la méthode par programmation dynamique introduite par Amini, l'utilisation de l'algorithme glouton ou algorithme *greedy*, proposé par Williams et Shah qui est l'implémentation que nous avons privilégiée.

Avant d'examiner ces différentes approches, nous allons préciser comment les différences finies permettent de résoudre informatiquement le problème de l'implémentation de notions définies dans un espace réel [29].

II.4.1. Différences finies

Les dérivées d'une fonction par rapport à une variable peuvent être approximées par des différences finies ou des éléments finis. Laurent David Cohen et Isaac Cohen ont montré que la recherche de solutions dans un espace de Sobolev était équivalente à celle dans un espace des fonctions polynomiales de dimension finie. Ils obtiennent, avec des éléments finis, des calculs similaires aux calculs utilisant les différences finies.

Selon Laurent David Cohen, la méthode par éléments finis est moins coûteuse et

plus stable. Nous n'utiliserons, dans cette mémoire, que les différences finies.

✓ **Continuité**

Un point quelconque de la courbe est représenté par :

$$V(s_i) = v_i(x_i, y_i)^t \tag{II.12}$$

La dérivée première des coordonnées par rapport au paramètre s peut être approximée par différences finies et le carré de la norme du vecteur des dérivés premiers devient :

$$\|v'(s)\|^2 = \left\| \frac{dv_i}{ds} \right\|^2 \tag{II.13}$$

L'énergie de continuité est alors liée à

$$\|v_i - v_{i-1}\|^2 = (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 \tag{II.14}$$

✓ **Courbure**

La dérivée seconde s'approxime par :

$$\|v''_{i(s)}\|^2 = \left\| \frac{d^2v_i}{ds^2} \right\|^2 \tag{II.15}$$

L'énergie de courbure dépend alors de :

$$\|v_{i-1} - 2v_i + v_{i+1}\|^2 = (x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2 \tag{II.16}$$

En fonction de la situation géométrique du contour, le choix de la méthode de mesure de la courbure n'est pas évident. Ainsi Williams et Shah Proposent un comparatif entre $(\frac{d\theta}{ds})^2$ ou θ est un angle l'axe des abscisses et le vecteur tangent a la courbe, et quatre approximation différent de la courbure k^2 avec

$$k = \frac{|x'y'' - x''y'|}{(x'^2 + y'^2)^{\frac{3}{2}}} \cdot |v_{ss}|^2 \cdot |\vec{u}_i - \vec{u}_{i+1}|^2 \cdot \left\| \frac{\vec{u}_i}{\|\vec{u}_i\|} - \frac{\vec{u}_{i+1}}{\|\vec{u}_{i+1}\|} \right\|$$

Et avec :

$$\vec{u}_i = (x_i - x_{i-1}, y_i - y_{i-1})^t$$

Et :

$$\vec{u}_{i+1} = (x_{i+1} - x_i, y_{i+1} - y_i)^t$$

Ils examinent la variation en fonction des voisins dans cinq directions

II.4.2 Approche variationnelle

Dans la méthode des contours actifs, il s'agit de minimiser une fonctionnelle d'énergie (équation (II.3), composée d'une énergie interne, d'une énergie externe, éventuellement d'une énergie de contexte.

Comme nous l'avons déjà précisé la recherche du contour est limitée au cas d'une courbe plane

$$V: \Omega = [0,1] \rightarrow \mathbb{R}^2 \quad (\text{II.17})$$

$$S \rightarrow v(s) = (x(s), y(s)) \quad (\text{II.18})$$

On définit un modèle déformable comme un espace de déformations admissibles Ad . Nous voulons minimiser l'énergie E

$$E: Ad \rightarrow \mathbb{R} \quad (\text{II.19})$$

$$V \rightarrow E(v) = \int_{\Omega} \alpha |v'(s)|^2 + \beta |v''(s)|^2 + p(v(s)) ds \quad (\text{II.20})$$

Un minimum vérifie l'équation d'Euler

$$\begin{cases} -(\alpha v')' + (\beta v'')'' + \nabla p(v) = 0 \\ v(0), v'(0), v'(1) \text{ donne} \end{cases} \quad (\text{II.21})$$

La courbe subit, par exemple, deux types de forces.

- ✓ Des forces intérieures qui imposent une certaine régularité. Le coefficient α impose l'élasticité et le coefficient β impose la rigidité de la courbe.
- ✓ Une force extérieure - Force d'image (terme de potentiel) [29] qui pousse la courbe vers les zones qui correspondent aux attributs recherchés. Elle est définie par un potentiel de forme :

$$p(v) = -\|\nabla I(v)\|^2 \quad (\text{II.22})$$

Où I est l'image. La courbe est attirée par le minimum local du potentiel, c'est-à-dire les maxima locaux du gradient, donc des contours.

D'autres forces, définies par l'utilisateur, peuvent être ajoutées.

Cette équation peut avoir plusieurs solutions puisque l'énergie peut avoir plusieurs minima locaux. La solution que l'on cherche est localisée dans une région donnée et on suppose posséder une valeur approchée de la solution v^0

On résout alors l'équation (II.21) en lui ajoutant un terme d'évolution $\frac{\partial v}{\partial t}$ pour former

$$\frac{\partial v}{\partial t} - \frac{\partial}{\partial s} \left(\alpha \frac{\partial v}{\partial x} \right) + \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 v}{\partial s^2} \right) = f_1(v) + f_2(v) \quad (II.23)$$

Où f_1 dérive de l'image pour attirer la courbe vers les contours et f_2 est une force interne qui simule une force de pression. L'équation (II.23) est une équation parabolique de la chaleur.

Nous considérons l'équation d'évolution :

$$\gamma \frac{\partial v}{\partial t} - (\alpha v')' + (\beta v'')'' = \frac{\partial \|\nabla I\|^2}{\partial v} \quad (II.24)$$

La solution de l'équation (II.24), quand v se stabilise, est la solution du problème statique. Le coefficient γ est un coefficient d'amortissement (de viscosité).

Après avoir discrétisé la courbe avec un pas h du paramètre s , les dérivées figurant dans (II.21) sont remplacées par des différences finies. Si

$$F(v) = (f_1(v), f_2(v)) = -\nabla p(v) + \dots$$

Est la somme des forces d'images et des forces externes, l'équation

$$-(\alpha v')' + (\beta v'')'' = -f(v)$$

Peut se discrétiser :

$$\frac{1}{h} (a_i(v_i - v_{i-1}) - a_{i+1}(v_{i+1} - v_i)) + \frac{b_{i-1}}{h^2} (v_{i-2} - 2v_{i-1} + v_i) - 2 \frac{b_i}{h^2} (v_{i-1} - 2v_i + v_{i+1}) + \frac{b_{i+1}}{h^2} (v_{i+2} - 2v_{i+1} + v_i) - (f_1(v_i), f_2(v_i)) = 0 \quad (II.25)$$

Avec $v_{i=v(ih)}$, $a_i = \frac{\alpha(ih)}{h}$, $b_i = \frac{\beta(ih)}{h^2}$

Ce système peut s'écrire $AV=F$

Où A est une matrice "presque" pentadiagonale, V représente les vecteurs de positions v_i et F les forces $F(v_i)$ en ces points. V et F sont des matrices 2 colonnes,

la première pour les composantes en x et la seconde pour y .

Le terme d'évolution est aussi remplacé par des différences finies, la discrétisation s'effectuant dans le domaine temporel. L'équation (II.24) au point d'indice i devient

$$\begin{aligned} \gamma (v_i(t) - v_i(t-1)) + \frac{1}{h^2} (\alpha_i v_i(t) - v_{i-1}(t)) - \alpha_{i+1} (v_{i+1}(t) - v_i) \\ + \frac{1}{h^4} (\beta_{i-1} (v_{i-2}(t) - 2v_{i-1} + v_i(t)) - 2\beta_i (v_{i-1}(t) - 2v_i(t) + v_{i+1}(t)) \\ + \beta_{i+1} (v_i(t) - 2v_{i+2}(t))) = \frac{\partial \|\nabla I\|^2}{\partial v} \end{aligned} \quad (II.26)$$

Ce qui peut s'écrire :

$$\begin{aligned} V(t) &= (A + \gamma I)^{-1} (\gamma V(t-1)) - \frac{\partial f}{\partial v} (V(t-1)) V(t) \\ &= (A + \gamma I)^{-1} (\gamma V(t-1)) - \frac{\partial f(v(t-1))}{\partial v} \end{aligned} \quad (II.27)$$

Où $f(v) = -\nabla I(v)$

En prenant $\alpha_i = \alpha$ et $\beta_i = \beta$ constants et $h = 1$, \mathbf{A} , est une matrice circulante symétrique de taille $N \times N$. Elle n'est pas toujours réellement penta diagonale en raison des diverses conditions aux limites adoptées. Berger présente les différents cas [29] à extrémités fixes, à extrémités libres et dans le cas d'un modèle fermé, elle se présente sous la forme

$$\mathbf{A} = \begin{pmatrix} 2\alpha + 6\beta & -\alpha - 4\beta & \beta & 0 & \dots & 0 & \beta & -\alpha - 4\beta \\ -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta & \beta & \dots & \dots & 0 & \beta \\ \beta & -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta & \dots & \dots & \dots & 0 \\ 0 & \beta & -\alpha - 4\beta & 2\alpha + 6\beta & \dots & \dots & 0 & \dots \\ \dots & \dots & \dots & -\alpha - 4\beta & \dots & \dots & \beta & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & -\alpha - 4\beta & \beta \\ \beta & 0 & \dots & \dots & \dots & -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta \\ -\alpha - 4\beta & \beta & 0 & \dots & 0 & \beta & -\alpha - 4\beta & 2\alpha + 6\beta \end{pmatrix} \quad (II.28)$$

Cette implémentation est la plus courante. Elle est la plus utilisée et la plus déclinée. Elle nécessite pour atteindre la solution, à chaque itération, l'inversion de la matrice, ainsi que le réglage du coefficient d'évolution γ .

II.4.3 Programmation dynamique

La programmation dynamique est une méthode classique de résolution de problème d'optimisation. Son application aux contours actifs est due à Amini, Weymouth et Jain. Cette approche peut être une alternative intéressante au calcul variationnel. Amini considère l'équation classique

$$E_{tot} = \int_0^1 E_{ext}(v(s)) + \frac{1}{2}(\alpha(s)|v_s|^2 + \beta(s)|v_{ss}(s)|^2)ds = \int_0^1 E_{ext} + E_{int} \quad (\text{II.29})$$

En représentant la fonction à intégrer par $F(s, v_s, v_{ss})$ la solution d'Euler Lagrange donne :

$$F_v - \frac{\partial}{\partial s} F_{v_s} + \frac{\partial^2}{\partial s^2} F_{v_{ss}} = 0 \quad (\text{II.30})$$

En discrétisant avec :

$$E_{int}(v_i) = (\alpha_i |v_i - v_{v-1}|^2 + \beta_i |v_{i+1} - 2v_i + v_{i-1}|^2)/2 \quad (\text{II.31})$$

On convient :

$$E_{tot} = \sum_{i=0}^{n-1} E_{int}(v_i) + E_{ext}(v_i) \quad (\text{II.32})$$

Cette somme peut être vue comme un processus discret de décisions à plusieurs niveaux. En partant d'un premier point du contour, il est possible de traiter le problème global de minimisation comme un problème de minimisation qui, pour chaque ensemble fini d'étapes $(i_0, i_1 \dots i_{i-1})$ Prend une décision parmi un ensemble fini de solutions possibles.

Le terme d'énergie interne est composé du terme du premier degré et du terme du second degré. Après la discrétisation, cette énergie interne met en jeu un élément du contour, son prédécesseur et son successeur. Il est possible de l'exprimer :

$$E_{total}(v_1, v_2, \dots, v_n) = E_1(v_1, v_2, v_3) + E_2(v_2, v_3, v_4) + \dots + E_{n-2}(v_{n-2}, v_{n-1}, v_n) \quad (\text{II.33})$$

Ou

$$E_{i-1}(v_{i-1}, v_i, v_{i+1}) = E_{ext}(v_i) + E_{int}(v_{i-1}, v_i, v_{i+1}) \quad (\text{II.34})$$

On se ramène donc à un problème d'optimisation d'une fonction numérique de plusieurs variables. Les variables seront ici les positions des différents points du *snake*. La formulation standard sous forme récurrente de la programmation dynamique peut s'écrire :

$$s_i(v_{i+1}, v_i) = \min_{v_{i-1}} \{s_{i-1}(v_i, v_{i-1}) + \alpha|v_i - v_{i-1}|^2 + \beta|v_{i+1} - 2v_i + v_{i-1}|^2 + E_{ext}(v_i)\} \quad (\text{II.35})$$

Chaque itération donne un contour optimal. La convergence de la minimisation de l'énergie est garantie, mais la complexité est élevée. Pour un voisinage de taille m , et un contour de n points, la complexité est de $O(nm^3)$, la taille de la mémoire nécessaire est $O(nm^2)$. La procédure est, par contre, parallélisable. Cette implémentation, d'autre part autorise l'introduction de différentes contraintes comme l'énergie ballon ou d'au très énergie.

II.4.4 Algorithme "Greedy"

L'utilisation de l'algorithme *greedy* pour minimiser l'énergie d'un contour actif, comme l'ont proposé Williams et Shah, est devenue une alternative assez fréquente à l'approche variationnelle, Williams et Shah, de manière classique discrétisent l'expression :

$$E_{tot} = \int_0^1 E_{ext}(v(s)) + \frac{1}{2}(\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2) ds \quad (\text{II.36})$$

Par différences finies et utilisent pour la continuité (comme en (II.23))

$$\left\| \frac{dv_i}{ds} \right\|^2 = \|v_i - v_{i-1}\|^2 = (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 \quad (\text{II.37})$$

$$\left\| \frac{d^2v_i}{ds^2} \right\|^4 = \|v_{i-1} - 2v_i + v_{i+1}\|^4 = (x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2 \quad (\text{II.38})$$

Il est à remarquer que ces deux expressions supposent deux hypothèses. La première est que les points soient placés, le long de la courbe, à une distance unité les uns des autres. Si les points sont équirépartis, alors l'équation (II.37) doit être divisée par d^2 et l'équation (II.38) par d^4 . Dans le cas contraire, la continuité est incorrecte d'un facteur d_i^2 où d_i correspond à la distance entre les points i et $i-1$.

Cela impliquera une valeur supérieure de la continuité, de même pour la courbure. La seconde hypothèse suppose que le paramètre soit une longueur d'arc, pour que la courbure soit donnée par $\|v_{ss}\|$

L'algorithme du *greedy*, comme la programmation dynamique autorise l'introduction d'autres contraintes. Il est plus rapide que la méthode précédente puisqu'il est, à chaque itération en $O(nm)$ au lieu de $O(nm^3)$ pour un contour comportant n points et un voisinage de taille m .

L'expression à minimiser, sans ajout de contraintes particulières, est :

$$E_{tot} = \int (\alpha(s)E_{cont} + \beta(s)E_{cour} + \gamma(s)E_{image}) \quad (II.39)$$

L'algorithme est itératif comme celui de Kass et d'Amini. On examine, à chaque itération, le voisinage de chaque point. On choisit le point du voisinage qui donne la plus faible valeur de l'énergie totale

Après discrétisation (équation (II.37) et équation (II.38)), la minimisation de la distance entre les points implique une rétraction naturelle du contour. Williams et Shah [29] proposent une variante qui évite cette rétraction trop importante.

Ils utilisent la différence de distance entre les deux points, $\|v_i - v_{i-1}\|$ par rapport à la distance moyenne du point de contour \bar{d} .

$$E_{cont} = \bar{d} - \|v_i - v_{i-1}\| \quad (II.40)$$

Les différents membres de l'expression sont normalisés avant minimisation. Les grandeurs utilisées sont le maximum, le maximum moins le minimum et le maximum moins le minimum avec saturation. L'énergie de continuité et l'énergie de courbure sont normalisées par division par la valeur du maximum dans le voisinage, donnant ainsi une valeur appartenant à l'intervalle $[0,1]$. La valeur de l'énergie externe ou d'image (val) est normalisée, sur le voisinage, par $(\min - val) / (\max - \min)$. Afin d'éviter de trop grandes variations dans des zones relativement homogènes, c'est-à-dire avec une faible variation du gradient, si $(\max - \min) < 5$ alors le min sera mis à cinq. A chaque itération, une étape de calcul de courbure est effectuée. Lorsque la courbure devient supérieure à un seuil alors le coefficient β est mis à zéro, autorisant ainsi la présence de coins. L'algorithme est présenté ce dessus

Une variante, permettant de rendre l'algorithme encore plus rapide, est proposée par Lam et Yan. Il s'agit, par exemple, pour un voisinage de 3 x 3 pixels sur les 8 voisins, de n'en examiner que quatre. Si l'un de ces quatre améliore l'énergie totale, alors il n'est pas nécessaire d'aller plus loin. Sinon, les quatre qui restent sont examinés. Cela augmente le nombre d'itérations pour atteindre la convergence mais décroît le temps de calcul de chaque itération.

II.5. Conclusion

Dans ce chapitre nous avons présenté les méthodes d'implémentations des contours actifs qui évolue dans le sens de la minimisation de la somme d'énergies (interne et externe) parmi ces méthodes on trouve les méthodes variationnelles, les méthodes par programmation dynamique et la méthode utilisant l'algorithme de Greedy que nous avons choisit et fera l'objet de notre étude dans le chapitre prochain.

Chapitre III :

Application, résultats

Et

Discussion

III.1.Introduction

Après avoir étudié les contours actifs et défini les énergies interne et externe qui seront étudiées en détail par la première section. Les méthodes d'implémentations des contours actifs les plus connues, les méthodes variationnelles, les méthodes par programmation dynamique et l'algorithme Greedy.

Dans ce chapitre nous passant à l'expérimentation et à la programmation de la méthode des contours actifs et nous avons choisit l'implémentation reposant sur un algorithme Greedy, qui est reconnu pour être l'un des algorithmes les plus rapides pour les contours actifs.

III.2.Implémentation des contours actifs par l'algorithme de Greedy

III.2.1.Fonctionnement du contour actif

KASS *et al.* proposèrent dès 1985, dans *Proceedings of First International Conference on Computer Vision*, une nouvelle méthode pour la détection de contours dans une image : les contours actifs. Ce contour est modélisé par une courbe qui se déforme en fonction de différentes énergies pour venir se positionner sur les frontières de l'objet. Pour les images numériques, le traitement s'effectuera avec un certain nombre de points chaînés, et dont le nombre varie en fonction de la précision désirée.

Etant donné l'existence de différentes formes d'images, des coefficients sont attribués à chacune des énergies utilisées. La fonctionnelle d'énergie représentera alors l'ensemble des énergies, auxquelles sont appliqués leur coefficient respectif. L'algorithme de greedy tendra à minimiser cette fonctionnelle d'énergie.

Il existe trois sortes d'énergies :

- ✓ les énergies internes
- ✓ les énergies externes
- Les énergies internes gèrent la cohérence de la courbe. Elles définissent la raideur de la courbe et la cohésion des points. Cette catégorie est composée de deux énergies : l'énergie de continuité (la dérivée première de la courbe), et l'énergie de courbure (deuxième dérivée).
- Les énergies externes prennent en compte les caractéristiques des images traitées. Nous avons choisi uniquement l'énergie du gradient (la dérivée première de l'image).

III.2.2 Programmation par L'Algorithme Greedy

Dans le cas de l'algorithme *greedy* utilisé, l'énergie globale (ou fonctionnelle globale d'énergie) du contour associée aux N points est :

$$E(c) = \sum_{i=1}^f \alpha E_{\text{continuté}}(M_i) + \beta E_{\text{courbure}}(M_i) + \gamma E_{\text{gradient}}(M_i) + \delta E_{\text{intensité}}(M_i) \quad (\text{III.1})$$

avec: M_i $i = 1 \dots N$, un ensemble de points ordonnés

Il s'agit de trouver l'ensemble des N points M_i qui définissent la courbe C . Pour cela, il existe plusieurs méthodes et parmi celles-ci, l'algorithme *greedy*.

a- Principe

La méthode du *greedy* consiste à faire évoluer le contour actif en minimisant la fonctionnelle d'énergie : pour chaque point du *snake*, on choisit un nombre de voisins pour lesquels on va calculer l'énergie, le point se déplacera alors sur le voisin qui possède l'énergie la plus faible. On cherche donc l'ensemble M des points pour laquelle l'énergie est minimale.

C'est donc un algorithme itératif qui déplace un point unique pour constituer un nouveau contour actif à chaque itération. Tous les points sont traités successivement lors de chaque itération.

Dans l'algorithme *greedy* utilisé, la fonctionnelle d'énergie est une somme de 3 énergies (continuité, courbure, gradient) normalisées et pondérées chacune par un coefficient (α , β et δ).

Pour déterminer le meilleur jeu de coefficients, on tire aléatoirement un jeu de paramètres, pour tous les points composant le snake.

a- L'algorithme

Le fonctionnement de l'algorithme de greedy, est représenté comme suit :

```

Faire
  Pour tous les points du snake
    Pour tous les points du voisinage
      Calculer les énergies
    Fin Pour
  Pour tous les points du voisinage
    Normalisation
  Fin Pour
  Minimiser pour obtenir le nouveau point
Fin Pour
Jusqu'au Critère d'arrêt
  
```

Nous allons dans la suite de cette partie tenter d'expliquer quel est le rôle de chaque énergie dans la décision du déplacement du snake. Nous allons parcourir aussi les autres paramètres propres à l'algorithme Greedy.

✓ L'énergie de Courbure

L'énergie de courbure définit l'angle formé par trois points adjacents du snake. Plus le coefficient attribué à cette énergie sera fort, et plus le snake aura la forme connue pour être la forme la moins « coûteuse » en énergie : la ligne droite ou le cercle pour des objets fermés. L'énergie de courbure est donnée par la relation suivante :

$$E_{\text{courbure}} = 0.5 * ((\text{point}_{i-1}.X - 2 * \text{point}_i.X + \text{point}_{i+1}.X)^2 + (\text{point}_{i-1}.Y - 2 * \text{point}_i.Y + \text{point}_{i+1}.Y)^2) \quad (\text{III.2})$$

✓ L'énergie de Continuité

L'énergie de continuité fait partie des énergies dites internes au contour actif. Cette énergie régit la distance entre les différents points du snake. Ainsi, lorsque le coefficient de l'énergie de continuité est nul, les points du snake pourront se déplacer aussi loin les uns des autres que l'image leur permettra. Dans le cas contraire, c'est-à-dire lorsque le coefficient de continuité est très élevé, le snake sera rigide.

L'énergie de continuité se calcule par rapport à la moyenne des distances entre les points du snake, telle que :

$$E_{\text{continuité}} = 0.5 * \text{abs} \left(\text{Distance Moyenne} - \sqrt{(\text{point } 2.X - \text{point } 1.X)^2 + (\text{point } 2.Y - \text{point } 1.Y)^2} \right) \quad (\text{III.3})$$

Nous remarquons que pour minimiser cette expression, le point *Point1* doit se positionner à une distance égale à la distance moyenne (*DistanceMoyenne*) du point *Point2*.

✓ **L'énergie de Gradient**

L'énergie de Gradient est la deuxième énergie du contour actif qui dépende de l'image. Cette énergie externe est d'une importance première pour la détection du contour. En effet, un contour est généralement caractérisé par une forte différence entre les valeurs de plusieurs pixels. Ainsi, si la dérivée d'une fonction représente les pentes d'une courbe, le gradient montrera les fortes différences, les contours de l'image.

En mathématique, le gradient d'une fonction spatiale (trois dimensions au minimum) représente simplement la dérivée de celle-ci suivant toutes les directions du plan (*O, i, j*) :

$$\overrightarrow{\text{grad}}(f(x, y)) = \frac{\partial f}{\partial x} \vec{i} + \frac{\partial f}{\partial y} \vec{j} \quad (\text{III.4})$$

En informatique, les fonctions sont discrètes, et il devient difficile voir impossible d'appliquer cette formule. Aussi, pour obtenir le gradient d'une image, donc d'une matrice de pixels, nous pouvons utiliser la méthode de Sobel.

Pour obtenir le gradient d'une image, chaque pixel, chaque case de la matrice image sera calculée en fonction de ses voisins pondérés par une matrice de coefficients déterminés

Matrices de Sobel :

1	2	1
0	0	0
-1	-2	-1

1	0	-1
2	0	-2
1	0	-1

$$\text{Grad}(m_{ij}) = \sqrt{\begin{aligned} &((m_{i-1j-1} + 2 * m_{ij-1} - m_{i+1j-1} - m_{i-1j+1} - 2 * m_{ij+1} - m_{i+1j+1})^2 \\ &+ \\ &(m_{i-1j-1} + 2 * m_{i-1j} + m_{i-1j+1} - m_{i+1j-1} - 2 * m_{i+1j} - m_{i+1j+1})^2 \end{aligned}} \quad (\text{III.5})$$

III.2.3. Parcours des différents paramètres supplémentaires réglables

a) Le voisinage

Pour déplacer un point, l'algorithme Greedy calcule toutes les énergies des pixels présents dans le voisinage du pixel de chaque point composant le snake.

Pour un voisinage de deux par exemple, l'algorithme va calculer les énergies d'une matrice de 5x5 pixels pour choisir son déplacement

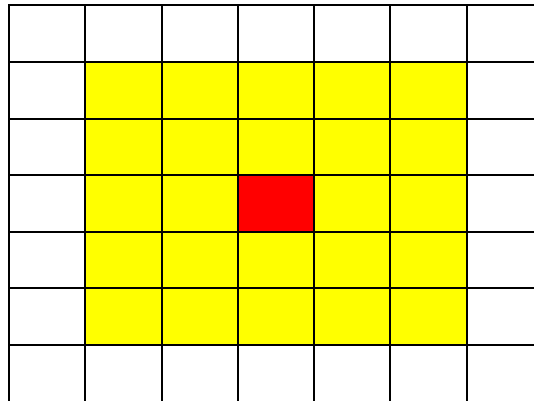


Fig. III.1. Exemple d'un voisinage de distance 2

b) . La normalisation

Afin d'éviter une trop grande dispersion des valeurs des énergies (certaines énergies varient entre 0 et 500 alors que d'autres entre -1 et 0), il est absolument nécessaire de normaliser, c'est-à-dire de diviser la valeur calculée par la plus grande valeur du voisinage.

Cela nous permet ainsi, d'obtenir des variations comprises entre -1 et 1, pour toutes les énergies.

c) . Le critère d'arrêt

Le critère d'arrêt est très important, notamment lorsque l'on travaille avec des images réelles. En effet, il faut parfois préciser au snake quand s'arrêter car, dans certains cas, le contour actif s'arrête pendant un moment sur les contours de l'image puis dépasse ceux-ci, attiré par des voisins dont l'énergie est plus faible.

Le critère d'arrêt que nous avons choisi, est le nombre d'itérations maximales. Le Snake s'arrête lorsqu'il atteint le nombre d'itération maximale

III.3. Résultats de simulation du contour actif implémenté par l'algorithme de greedy

Pour l'implémentation du contour actif nous avons choisi l'environnement Matlab, reconnu pour sa robustesse dans les calculs matricielle, sachant que les images sont représentées par des matrices de dimension élevée. La version de Matlab est R2010a sur machine de caractéristique : intel CORE i3, une mémoire RAM de 4 Go.

Les résultats obtenus sur des images tests qui sont

- L'image Cercle de taille : 256 x256 ;
- L'image carrée de taille : 256 x256 ;
- L'image avec deux carrées de taille : 256 x256.

a- Réglage des paramètres du contour actif

La paramètres sont choisit comme suit :

- Le paramètre alpha = 0.5 ;
- Le paramètre Beta =0.25 ;
- Le paramètre Gamma = 3 .

Et les résultats sont comme indiqué ci-après.

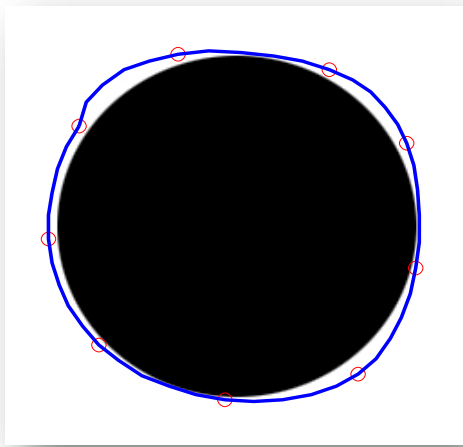


Image avec initialisation de contour

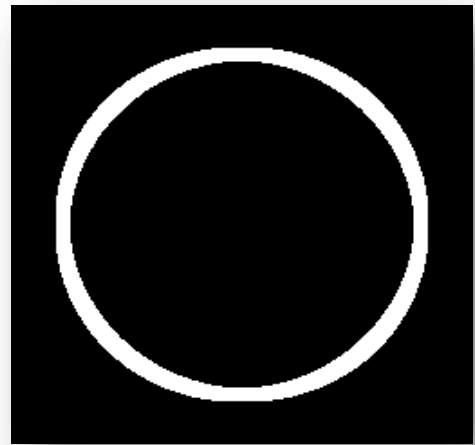
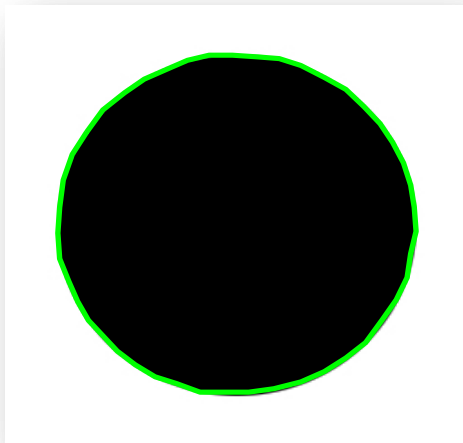
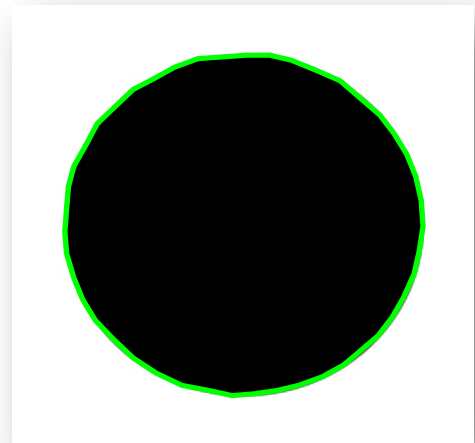


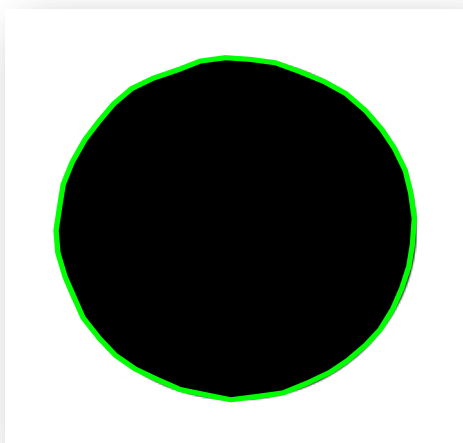
Image gradient



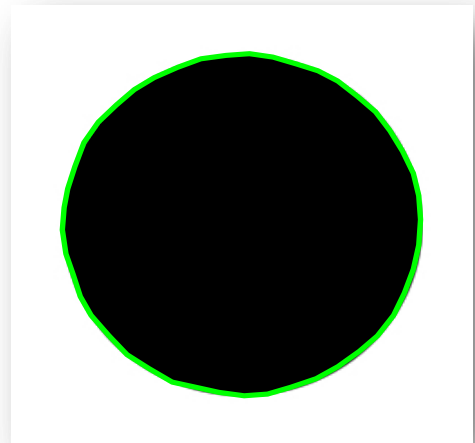
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III. 2. Résultat de segmentation de l'image test cercle pour les paramètres $\alpha=0.5$, $\beta=0.25$, $\delta=3$

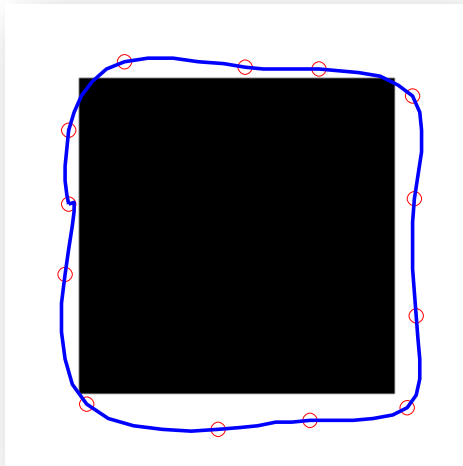


Image avec initialisation de contour

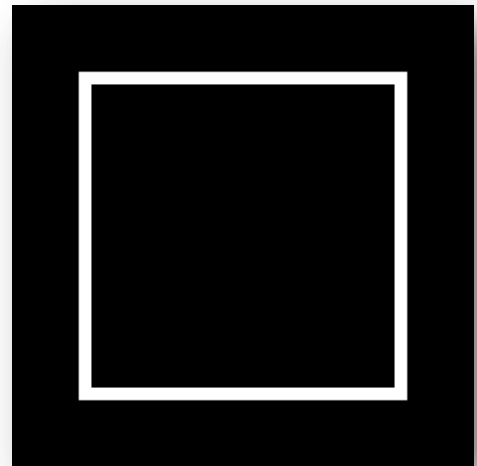
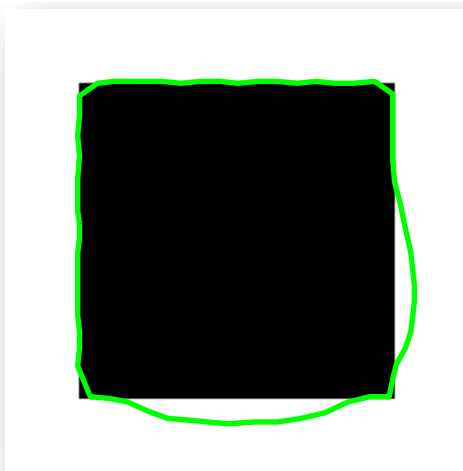
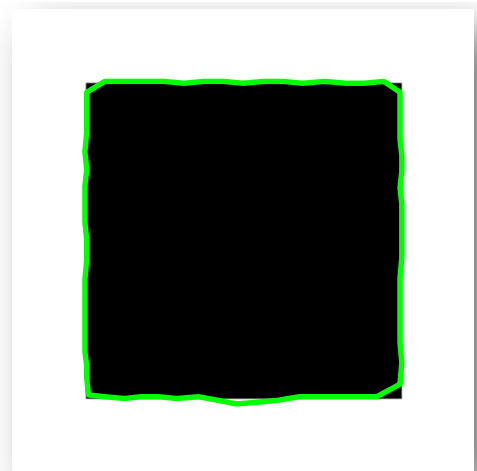


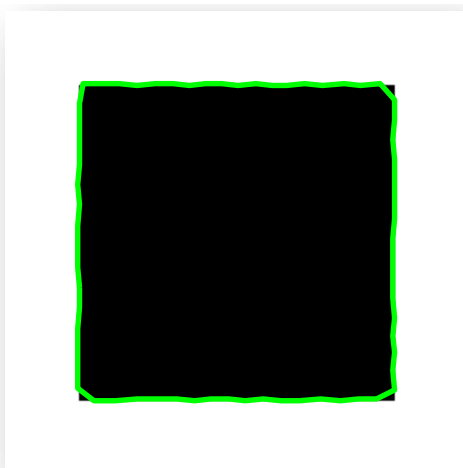
Image gradient



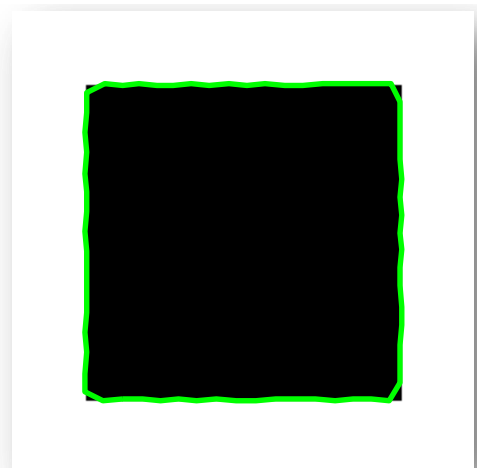
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.3. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.5$, $\beta=0.25$, $\delta=3$

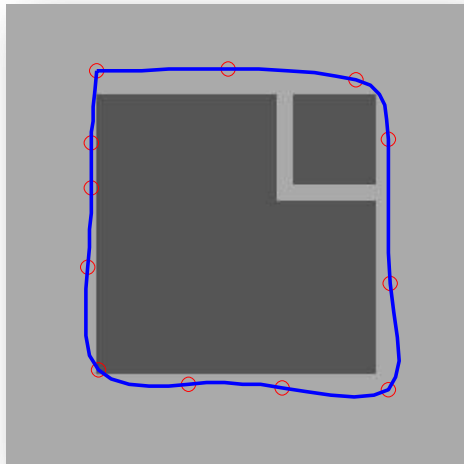


Image avec initialisation de contour

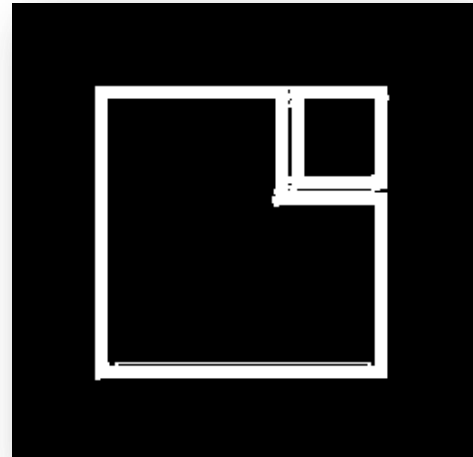
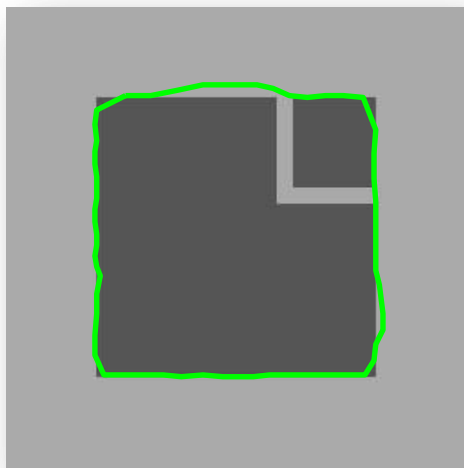
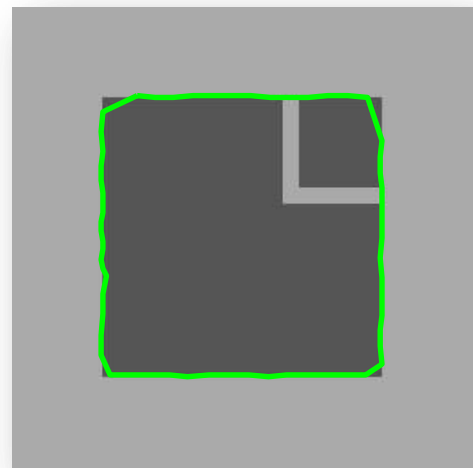


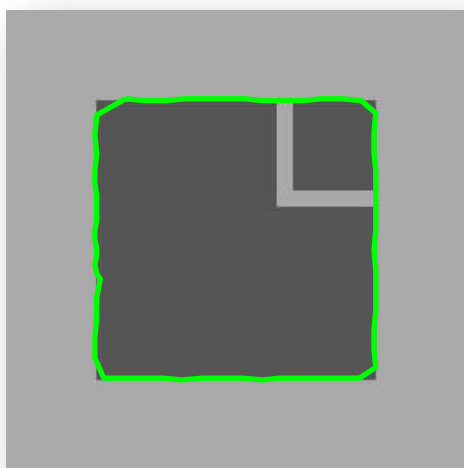
Image gradient



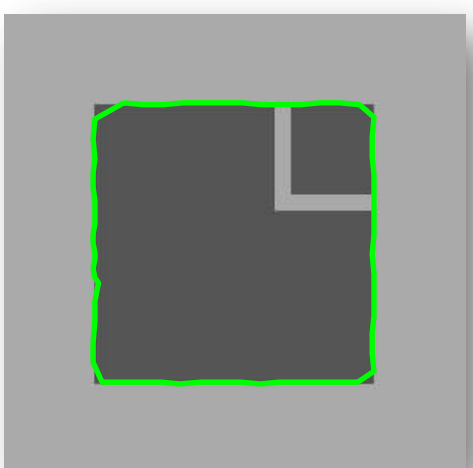
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.4. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.5$, $\beta=0.25$, $\delta=3$

a- Interprétation des résultats :

D'après les résultats des figures (figure III.2, figure.III.3 et figure.III.4) en remarque que le contour converge vers la rapidement, moins de dix itérations ce qui est expliqué par la rapidité de l'algorithme de greedy.

Les teste effectués sur les images précédente ne sont pas probants et sont d'une qualité moyenne parce que les snake sont sensible aux bruit, car un gradient fort du bruit peut bloquer la convergence du snake vers la solution recherchée ;

Le choix des coefficients aussi a pour but l'amélioration de la rapidité de la convergence du snake. Nous allons voir après l'influence de ces paramètres pour la détection de la forme des objets recherchés

La figure.III.4 montre aussi l'inconvénient majeur du snake, qui est la distinction ou la séparation des objets recherchés. Par contre dans la littérature on trouve d'autres méthodes qui sont performantes et permettent la sélection de plusieurs objets, tel que la méthode des ensemble de niveaux (level-set).

b- Influence des paramètres Alpha, Betta et Gamma

Pour montrer l'influence des paramètres sur l'évolution du contour actif, nous avons effectué une série de simulation durant lesquelles nous avons changé un seul paramètre à la fois entre la valeur zéro donc l'annulation de l'énergie associée à ce dernier et une valeur élevée :

Les résultats sont donnés comme suit :

- **Influence du paramètre Beta :**

Voici un test effectué avec un coefficient d'énergie de courbure et nul

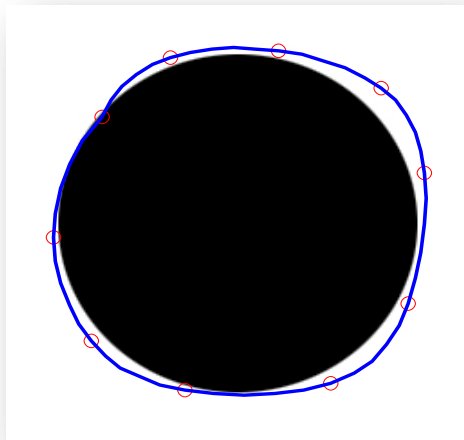


Image avec initialisation de contour

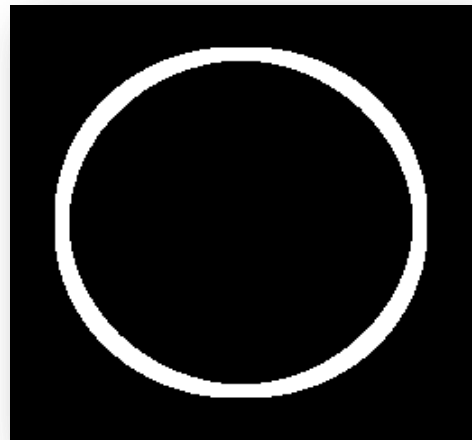
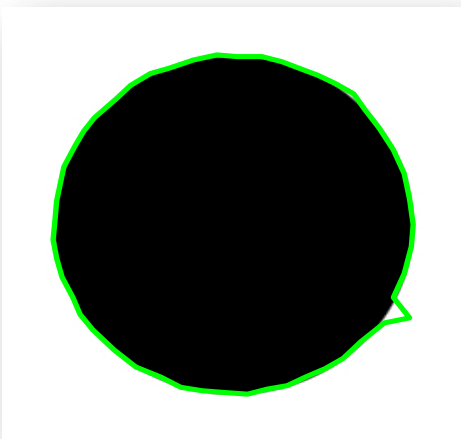
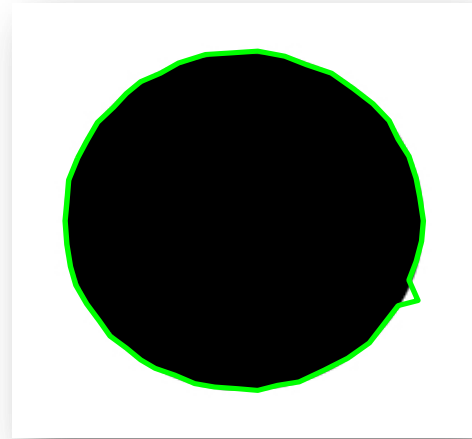


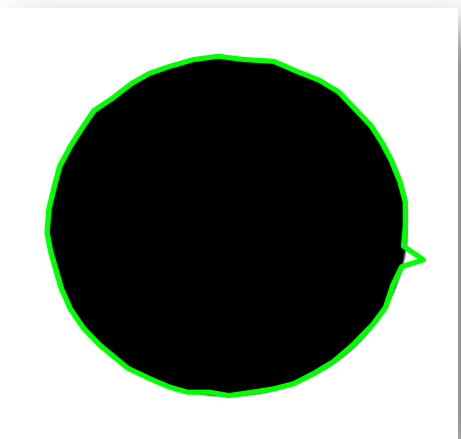
Image gradient



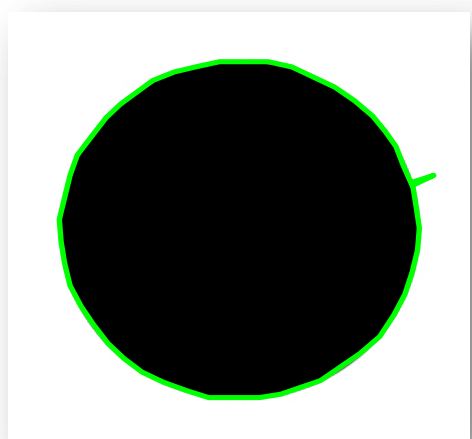
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.5. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.5$, $\beta=0$, $\delta=3$

Voici un test effectué avec un coefficient d'énergie de courbure élevé :

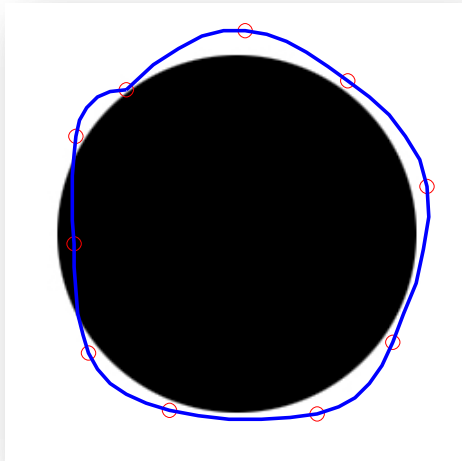


Image avec initialisation de contour

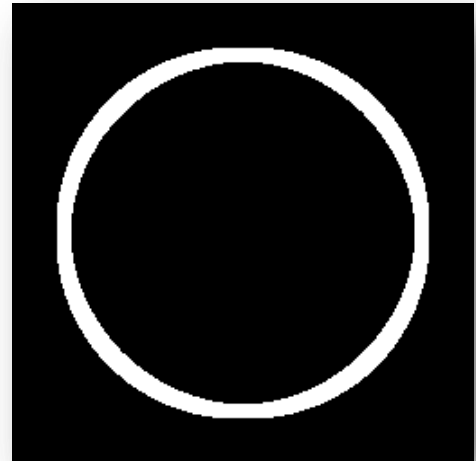
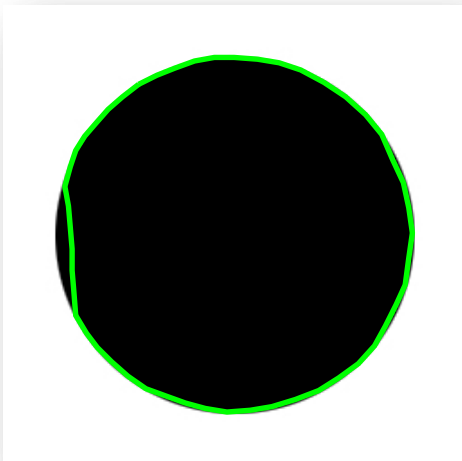
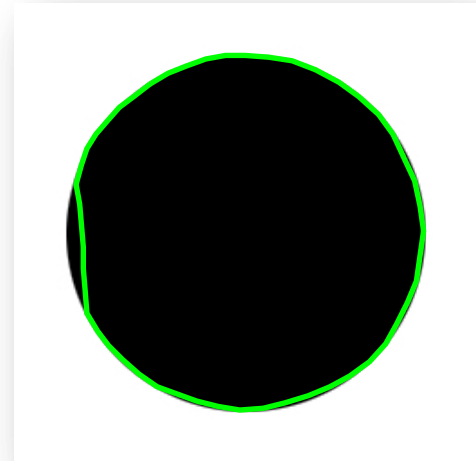


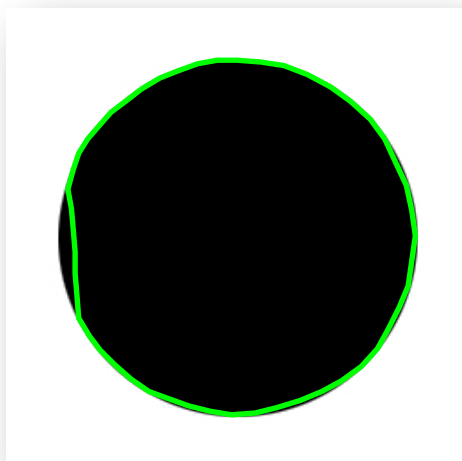
Image gradient



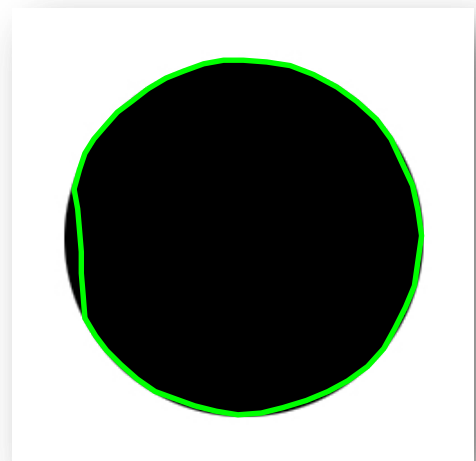
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.6. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.65$, $\beta=5$, $\delta=3$

- **Influence du paramètre Gamma :**

Voici un test effectué avec un coefficient d'énergie de gradient élevé

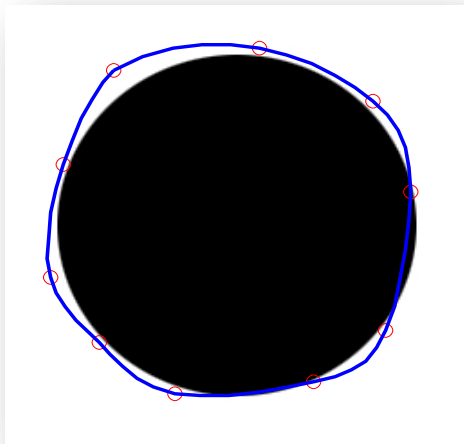


Image avec initialisation de contour

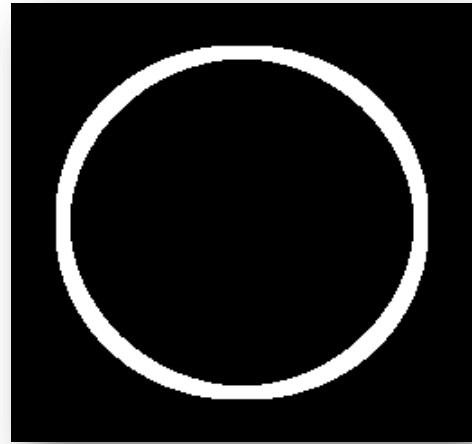
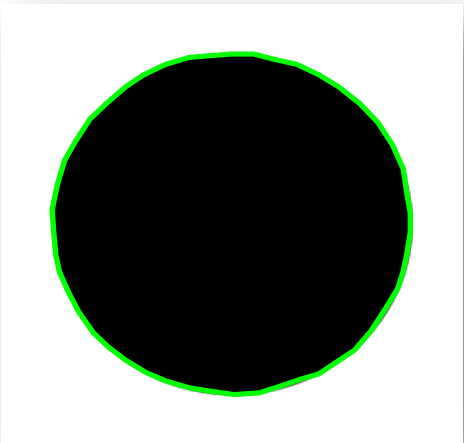
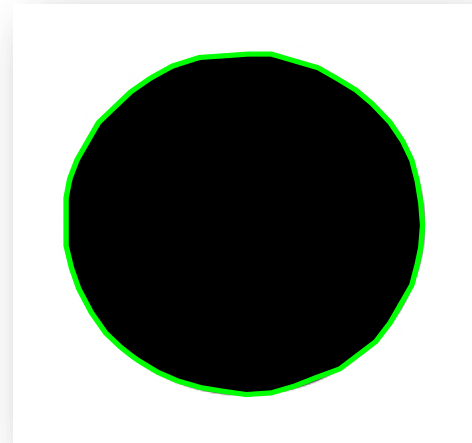


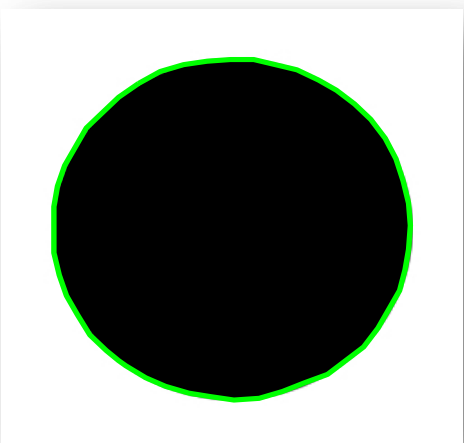
Image gradient



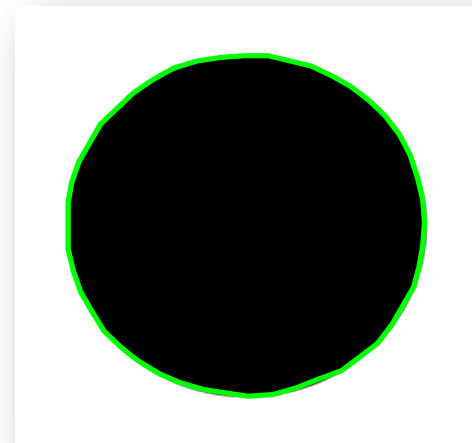
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.7. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.5$, $\beta=0.25$, $\delta=9$

Voici un test effectué avec un coefficient d'énergie de gradient nul

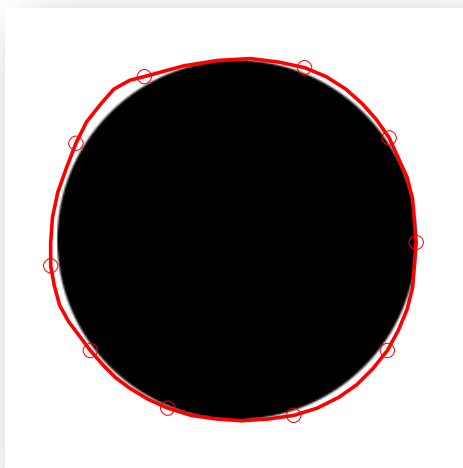


Image avec initialisation de contour

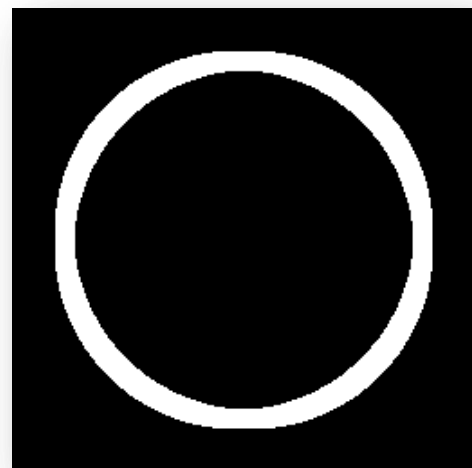
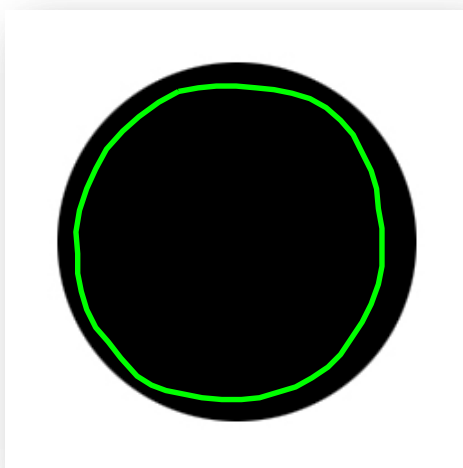
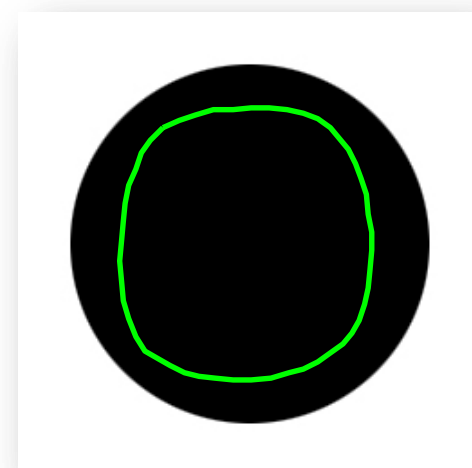


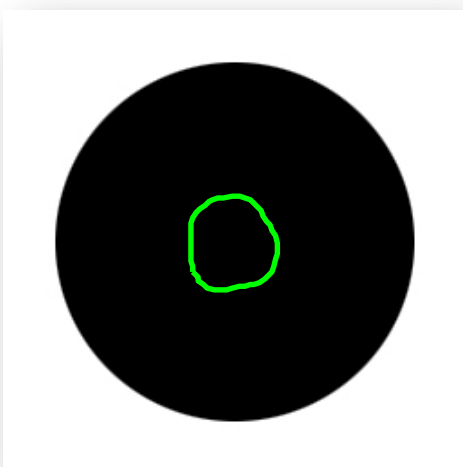
Image gradient



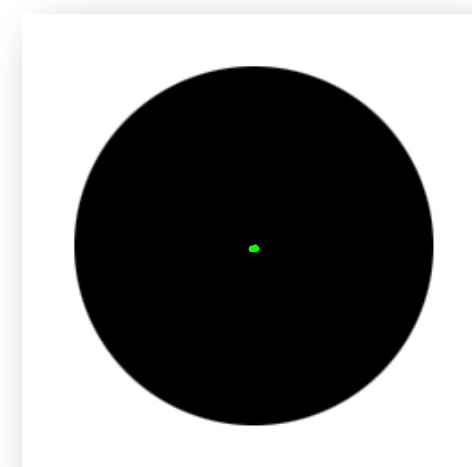
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.8. *Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0.5$, $\beta=0.25$, $\delta=0$*

- **Influence du paramètre Alpha :**

Voici un test effectué avec un coefficient d'énergie de continuité nul

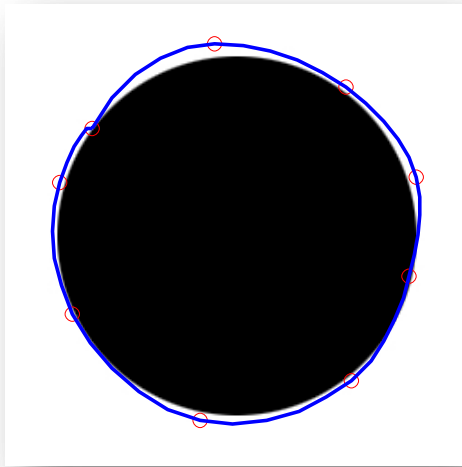


Image initialisation de contour

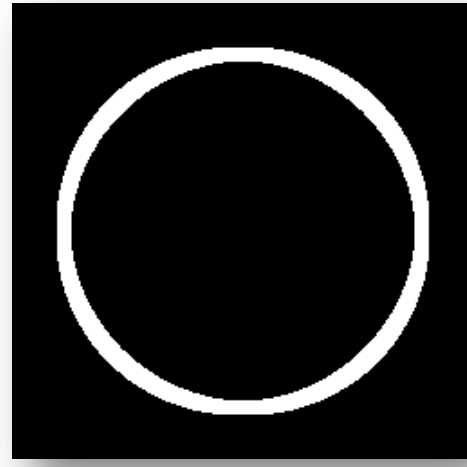
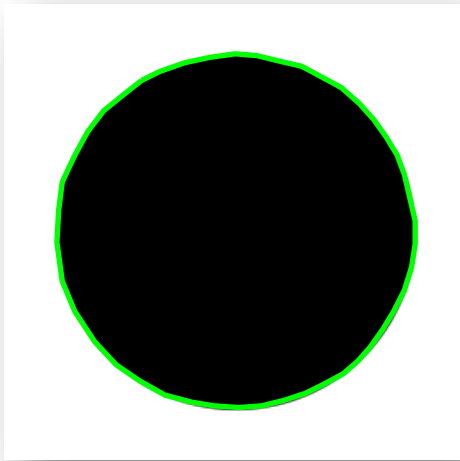
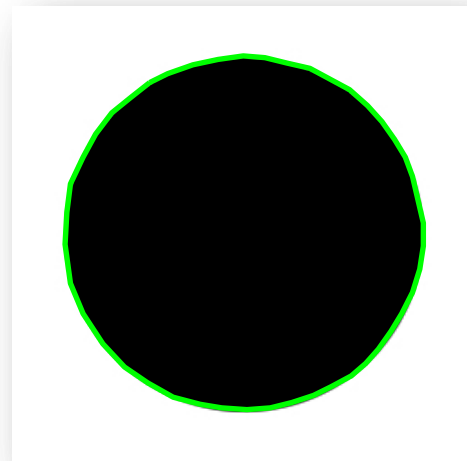


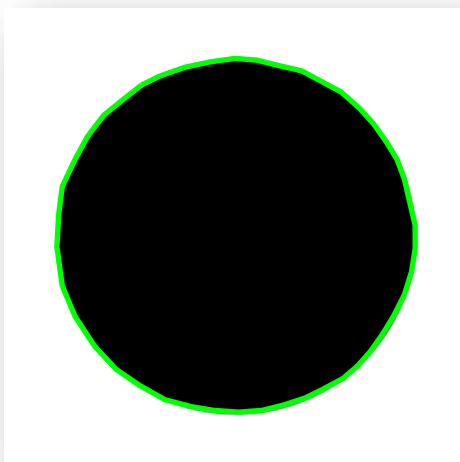
Image gradient



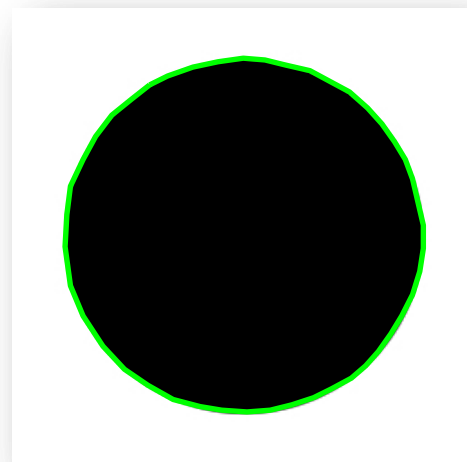
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.9. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=0$, $\beta=0.25$, $\delta=3$

Voici un test effectué avec un coefficient d'énergie de continuité élevé :

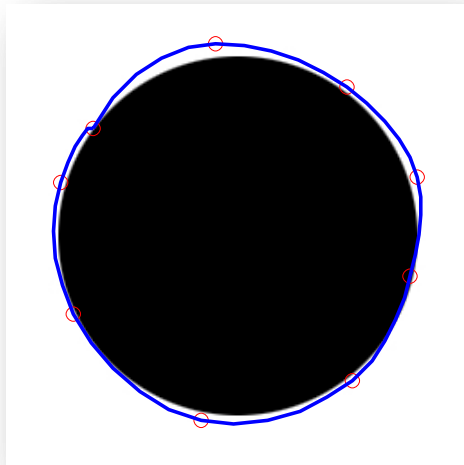


Image avec initialisation de contour

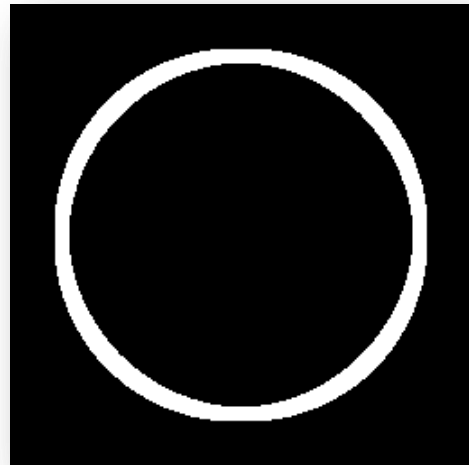
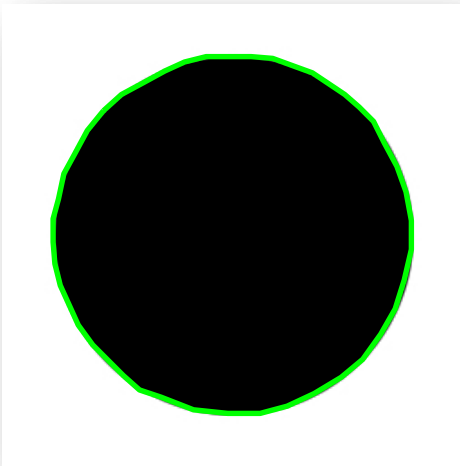
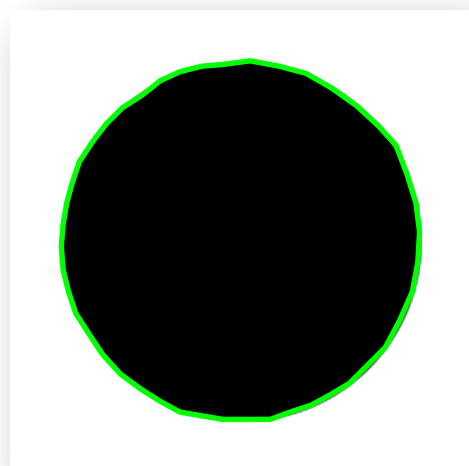


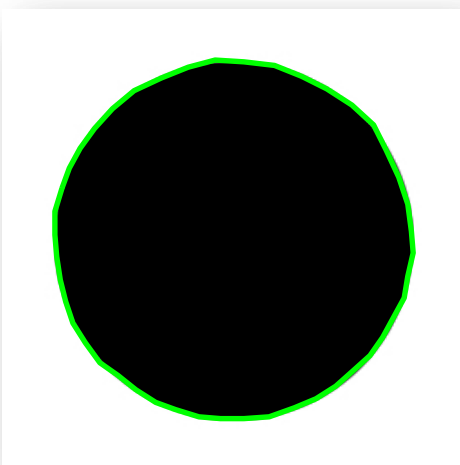
Image gradient



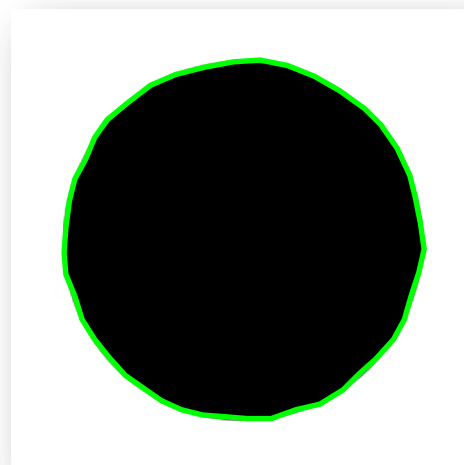
Résultat après 10 itérations



Résultat après 20 itérations



Résultat après 70 itérations



Résultat après 300 itérations

Fig. III.10. Résultat de segmentation de l'image test carrée pour les paramètres $\alpha=5$, $\beta=0.65$, $\delta=3$

b- Interprétation des résultats

D'après les résultats précédents en remarque que :

- Lorsque le coefficient β associé à l'énergie de courbure est nul, ceci permet d'observer des variations brusques sur le contour final, ceci est dû angles très obtus qui existent entre trois points du snake (figure.III.5).

Par contre un coefficient β de valeur élevée attribué à l'énergie de courbure, permet au snake d'avoir la forme connue pour être la forme la moins « coûteuse » en énergie comme montré en figure.III.6 (ligne droite ou le cercle pour des objets fermés).

- Lorsque le coefficient d'énergie de gradient δ est nul, veut dire que le snake n'a capté aucun contour, donc il ne s'arrête pas résultat de la figure.III.7. Ce qui implique que sans gradient, le contour actif ne pourrait pas connaître les emplacements des contours.

A l'inverse, le coefficient élevé du gradient bloque le snake. En effet, le gradient étant une dérivée spatiale de l'image, plus la différence de couleur entre deux pixels est élevée, plus le gradient sera fort, et plus le snake pensera qu'il s'agit d'un contour. Un fort gradient va donc stopper le snake sur une faible variation de couleur de l'image ceci n'est pas observable sur nos résultats parcequ'il n'y a pas une grande variation de couleurs (figure.III.8)

- Lorsque le coefficient de l'énergie de continuité α est nul (figure.III.9), les points du snake pourront se déplacer aussi loin les uns des autres que l'image leur permettra.

Dans le cas contraire, c'est-à-dire lorsque le coefficient de continuité est très élevé (figure.III.10), le snake sera rigide. Du fait de forte valeur du coefficient de continuité, la distance entre les points du snake est fixe. Cela oblige les points du snake à se déplacer à une distance fixe de ses points adjacents.

III.4. Conclusion

Le problème majeur des contours actifs réside dans la nécessité de régler, d'ajuster les coefficients de pondérations des différentes énergies mises en jeu. Les résultats précédents ont montré que le snake est très dépendant de ces coefficients.

Par contre, l'implémentation des contours actifs par l'algorithme de greedy à montrer la convergence rapide vers une solution optimale.

Conclusion générale

Dans ce travail, nous nous sommes intéressés aux méthodes de segmentation d'images particulièrement la méthode de segmentation par contours actifs. Dans une première partie, nous avons abordé les différentes approches de segmentation tel que l'approche contour, l'approche région et l'approche de classification.

Les contours actifs, les snakes qui font l'objet de recherche depuis plus de trente ans, présentent plusieurs variantes. Nous nous sommes intéressé à l'implémentation par l'algorithme Greedy. Cet algorithme est itératif et nous avons procédé à la minimisation d'une somme d'énergies composé de trois énergies pondérées qui sont l'énergie de courbure, de continuité et de gradient, et le déplacement s'effectue en comparant les différentes fonctionnelles d'énergies des pixels voisins pour chaque point composant le Snake.

La qualité des résultats obtenus montrent que l'algorithme Greedy converge vers la solution rapidement (après 10 itérations), et donne des résultats satisfaisant, cette rapidité est fortement lié au choix des paramètres qui interviennent dans la fonctionnelle d'énergie à minimiser. Sachant que le choix des paramètres qui interviennent dans la fonctionnelle d'énergie est une opération délicate et qui varie selon le type d'image utilisée.

Comme perspectives nous pouvons améliorer l'algorithme par :

- L'association d'autres types d'énergies, tel que l'énergie de contexte (force Ballon) et quelques énergies externes lier aux propriétés de l'image (énergie intensité).
- L'introduction d'autres critères d'arrêt.
- Estimation itérative des paramètres de pondération aux énergies.

Références bibliographie

[01] : Thèse ROCHERY. M, «contours actifs d'ordre supérieur et leur application a la détection de linéiques dans les images de télédétection » thèse de docteur université de Nice 2005

[02] : NEBEL Fabien « Amélioration des paramètres des contours actifs par une technique issue de la théorie de l'apprentissage ».thèse de docteur, Laboratoire d'informatique de TOURS. **2004.**

[03] : LOUASSA. R, LAICHI M, « La Segmentation Des Images Application : Image à Résonance Magnétique (IRM) » thèse d'ingénieur d'état université m'sila.**2006.**

[04] : T. Pavlidis and Y.-T. Liow. Integrating region growing and edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence, 12(3):225--233, 1990.

[05] : OUADFEL. S «Contributions à la Segmentation d'images basées sur la résolution collective par colonies de fourmis artificielles » thèse de docteur Université Hadj Lakhdar de Batna 2006.

[06] : G.Giraudon, Chaînage efficace de contours, Rapport interne INRIAn°605, janvier 1987.

[07] : M.F Hueckel An operator which locates edges in digitized pictures, J. Ass.Comput. Mach, vol. 18, n° 1, pp. 113-125, 1971.

[08] : A. Lopès, R. Fjørtoft, D. Ducrot, P. Marthon, and C. «Le maréchal. Segmentation of SAR images in homogeneous regions ». In C. H. Chen, editor, Information Processing for Remote Sensing. World Scientific Publishing Co., Singapore, 1999.

[09] : M.Chang,I. Sezan. I et M. Tekalp."Adaptive Bayesian Segmentation of color images". Journal of electronic imaging, vol. 3, no 4, pp. 404-414, 1994.

[10] : J.M. Chassery , Montonvert A., «Géométrie discrète en analyse d'image »,

[11] : R.C. Gonzalez and R.E, Woods digital image processing Addison- Wesley, Reading, MA.

[12] : A. Trémeau et N. Borel. A« region growing and merging algorithm to color segmentation». Pattern Recognition, 30:1191–1203, 1997.

[13] : J.P.Cocquerez and S. Philipp. « Analyse d'images, filtrage et segmentation » «Comparaison de méthodes de segmentation d'images, conf. Invité ». Masson, Paris, 1995.

- [14] : J.Bryant On «the clustering of multidimensional pictorial data » In Pattern Recognition, number 11, pages 115-125, 1979.
- [15] : R. Kara-Falah «*Segmentation d'images : Coopération, fusion, évaluation* ». Thèse de doctorat, Université de Savoie, Juin 1995.
- [16] : R. Adams, et L. Bischof Seeded region growing IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 6, 1994.
- [17] : Mehnert and P. Jackway, An improved seeded region growing algorithm, Pattern Recognition Letters, vol. 18, pp. 1065-1071, 1997.
- [18] : R.H. Turi *Clustering-Based Color Image Segmentation*, Phd These. Monash University, Australia 2001.
- [19] : P. Bertolino. Contribution des pyramides irrégulières en segmentation d'images multi resolution. Thèse de doctorat. Institut National Polytechnique de Grenoble. France. Novembre 1995.
- [20] : J.R. Beveridge, J.G. Riffith. R. R. Kohler A.R. Hanson et E. M. Riseman Segmenting images using localized histograms and region merging IJCV2 pp. 311-347, Janvier, 1989
- [21] : A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering : a review. ACM Computing Surveys, 31(3) :264*322, 1999.
- [22] : A.K Jain et R.C Dubes Algorithms for clustering Data. Prentice hall advanced Reference series, 1988.
- [23] : C. Cortes et A. Hertz, A Network System for Image Segmentation, Proceedings of the International Joint Conference on Neural Networks, vol. 1,pp. 121-125, Washington DC, USA, June 1989.
- [24] : N. Babaguchi, et Yamada, K. K. Kise, and T. Tezuka, "Connectionist Model Binarization," Proceedings of the 10th International Conference on Pattern Recognition, vol. 1, pp. 51-56, Atlantic City, New Jersey, USA, June 1990.
- [25] : L. Yin, J. Astola, A new Class of Nonlinear Filters-Neural Filters, IEEE Transaction on Signal Processing, pp 1201-1222, 1993.
- [26] : F. Devaux, *Filtrage d'images par réseaux de neurones*. Maîtrise en informatique Université Paris 8, 1997.

[27] : V.V. Raghavan et K. Birchard. A clustering strategy based on a formalism of the reproductive process in natural systems. In Information Implications into the Eighties, Proceedings of the Second International Conference on Information Storage and Retrieval, pages 10–22. ACM, 1979

[28] : Ferber Jacques, Les Systèmes Multi-Agents : vers une Intelligence Collective, Inter Editions, 1995.

[29] : Thèse de J.J. Rousselle, « Les contours actifs, une méthode de segmentation », thèse de docteur, Laboratoire d'informatique de TOURS, **2003**.

MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLOME
DE MASTER EN GENIE ELECTRIQUE
SPECIALITE : AUTOMATIQUE

Proposé et dirigé par : Mr. HAMMOUCHE Abdelaziz

Présenté par : LAIFA Hamza

Thème

Segmentation D'image Par Une Approche Contour Actif

Résumé :

Les méthodes de segmentation d'images sont nombreuses, toutes présentent des avantages mais ne donnent pas entière satisfaction. Toutes doivent être adaptées en fonction des applications que l'on se propose de réaliser.

Dans l'approche de segmentation par contour actif, la dynamique est basée sur la notion d'énergie interne et externe, le but étant de minimiser l'énergie totale présente le long de la courbe. Des contraintes permettent de conserver une courbe lisse avec des points équidistants tout en laissant un certain champ libre pour les déformations. L'énergie interne correspond à la morphologie et aux caractéristiques de la courbe (courbure, longueur, etc.). L'énergie externe provient des informations propres à l'image, les critères sont variables (l'image intensité, Gradient de l'image, etc.). L'évolution se fait de manière itérative et les algorithmes peuvent faire l'objet de diverses optimisations et techniques numériques, Le contour actif que nous avons étudiés est implémenté par un algorithme de Greedy.

Mots clés :

Traitement d'images, segmentation, contours actifs, Energies, algorithme *greedy*,