

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ MOULOUD MAMMERRI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de fin d'études
Présenté en vue de l'obtention
du Diplôme d'Ingénieur d'Etat en Electronique

Option : contrôle-communication

Thème:

**Segmentation d'image IRM cérébrale par
contours actifs
(snake classique et snake GVF)**

Proposé et dirigé par :

Madame *ARAB Dalila*

Présenté par :

BOUCHENEB Chabha
BOULJDA Nacera
BOUSLENE Sihem

Année universitaire 2008/2009

Remerciements

Après une année de labeur pour la réalisation de ce mémoire, nous tenons à remercier toute personne qui nous a apporté aide de près ou de loin.

Tout d'abord nous remercions notre encadreur M^d. DALILA ARAB, enseignante au département d'électronique qui nous a apporté son expertise avec la plus grande disponibilité tout au long de l'année et ses conseils et pertinentes remarques, sans elle ce travail n'aurait pas vu le jour.

Et sans oublier madame chilali qui nous a aidés énormément.

Nous remercions également les membres du jury qui nous font honneur de juger notre travail.

Et plus généralement, Merci à tous ceux de notre entourage qui ont montrés de l'intérêt pour ce travail.

Sihem, Nacera et Chabha.

Dédicaces

Dieu merci

Je dédie ce modeste travail :

A mes très chers parents qui ont toujours été là pour moi, et qui m'ont donné un magnifique modèle de labeur et de persévérance. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour, même si je sais que je ne les rendrai jamais assez

A mon grand-père et à ma grande -mère que dieu les protège.

A mes chers frères : Farid, Ali, Samir et Lounis.

A ma chère sœur : Ghania.

A mes chers amies et amis:

Karima, Faiza, Samira , Hania, linda,aldjia,hassina, fadila,djilali,, allilou, zina ,moh, ghania, doudou, Omar, sans oublier mes binômes

Chabha et sihem et a tous mes profs

A toute ma famille ; mes cousins

A toute la promotion ELN de l'année 2008/2009.

A ceux qui m'ont poussé à aller toujours de l'avant.

Nacera

Dédicaces

Dieu merci

Je dédie ce modeste travail :

A mes très chers parents qui ont toujours été là pour moi, et qui m'ont donné un magnifique modèle de labeur et de persévérance. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour, même si je sais

que je ne les rendrai jamais assez

A mon grand-père que dieu le protège.

A mes chers frères : Farid et mahdi.

A mes chers sœurs : Zohra et ses enfants Inesse et Ziridesse.

A Hayat et sa famille surtout Ouarda.

Et aussi a Nawal et Tiziri.

A mes chers amies et amis:

Samou, Kahina, allilou, adel, zina, Omar, sans oublier mes binômes Chabha et Nacera et a tous les membres de L'association culturelle agouni ouzithouth.

A toute ma famille ; mes cousins et mes oncles en particulier : KHALTI Zola.

A toute la promotion ELN de l'année 2008/2009.

A ceux qui m'ont poussé à aller toujours de l'avant.

Sihem

Dédicaces

Dieu merci

Je dédie ce modeste travail :

A mes très chers parents qui ont toujours été là pour moi, et qui m'ont donné un magnifique modèle de labeur et de persévérance. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour, même si je sais que je ne les rendrai jamais assez

A mes chers frères : Hocine et Mokrane

En leurs espérant le plein succès dans leur vie.

A ma sœur Kahina et son mari Mustapha et leur deux fille Léa et Anaïs.

A mes chers amies et amis:

Lyes, Alilou, Omar, Fatma, Sabrina, Tassadit, Houria sans oublier mes binômes Sihem et Nacera .

A toute ma famille et tous ceux qui me sont chers

A toute la promotion ELN de l'année 2008/2009.

A ceux qui m'ont poussé à aller toujours de l'avant.

Que dieu vous protège

Chabha

Introduction générale.....	1
CHAPITRE I	<i>Notion de base sur le traitement d'image</i>
I.1 Introduction.....	4
I.2 Définitions générales	4
I.2.1 Qu'est-ce qu'une image	4
I.2.2 Le pixel.....	4
I.2.3 Les images bitmap	5
I.2.3.1 Le format GIF	5
I.2.3.2 Le format PNG	5
I.2. 3.3 Le format TIF	5
I.2. 3.4 Le format JPEG	5
I.3 Types d'image	6
I.3 1 Image binaire	6
I.3.2 Image en niveau de gris	6
I.3.3 Image couleur R.V.B	7
I.3.4 Image indexée	7
I.3.5 Image numérique	7
I.3.5.1. Balayage	8
I.3.5.2 Echantillonnage	8
I.3.5.3 Quantification	8
I.4 Caractéristiques d'une image numérique	9
I.4.1 Dimension	9
I.4.2 Résolution	9
I.4.3 Contours	9
I.4.4 Luminance	10
I.4.5 Voisinage	10
I.4.6 Contraste	10
I.4.7Le bruit	11
I.5 Histogramme	11
I.6 Filrage	12
I.7 Seuillage	12

CHAPITRE II

Méthodes de traitement d'Images

I.1 Introduction.....	15
II.2 Prétraitement	15
II.2.1 Réduction du bruit	15
II.2.1.1 Définition	15
II.2.1.2 Filtre linéaire	16
II.2.1.2.1 Filtre passe-bas	16
a) Filtre de moyenne.....	16
b) Filtre gaussien	16
II.2.1.2.2 Filtre passe haut	17
II.2.1.3 Filtre non linéaire (filtre médian).....	18
II.2.1.4 Filtre morphologique	19
a) Erosion	19
b) Dilatation	19
c) Ouverture.....	20
d) Fermeture	20
II.3 les différentes opérations appliquées sur l'histogramme	21
II.3.1 Modification d'histogramme	21
II.3.2 Expansion de la dynamique	21
II.3.3 Egalisation d'histogramme	21
II.4 La segmentation	22
II.5 Les approches de la segmentation	23
II.5.1 Approche frontière (contour)	23
II.5.1.1 Méthodes dérivatives	24
II.5.1.1.1 Utilisation du gradient	24
a) Le gradient.....	24
b) Laplacien	27
c) Le seuillage.....	28
II.5.1.1.2 Algorithme de Canny	29
II.5.1.1.3 Filtre de deriche	29
II.5.1.1.4 Filtres de Shen-Castan	30
II.5.2 Approche région	30

II.5.2.1 Segmentation par fusion (merge)	30
II.5.2.2 Segmentation par division (Split)	31
II.5.3 Approche par classification des pixels	31
II.6 Conclusion	31

CHAPITRE III

contours actifs

III.1 Introduction	34
III.2 Principe du contour actif	34
III.3 Le contour actif basé contour (snake)	36
III.3.1 L'énergie d'un contour	36
III.3.1.1 L'énergie interne	36
III.3.1.2 L'énergie externe	37
III.4 Processus de minimisation	37
III.4.1 Méthodes d'optimisation.....	38
III.4.1.1 Méthodes itératives	38
III.4.1.2 Algorithme glouton (greedy)	38
III.4.1.3 La programmation dynamique	39
III.5 Implémentation des snakes par méthode variationnelle	39
III.6 Avantages et inconvénients des contours actifs	43
III.6.1 Avantages	43
III.6.2 inconvénients	43
III.7 la force de ballon	44
III.8 flux du vecteur gradient	45
III.8.1 principe du contour GVF	45
III.8.1.1 Calcul de la carte de contours	46
III.8.2 Le calcul du champ GVF	46
III.9 flux vecteur gradient généralisée	48
III.9.1 définition	48
III.9.2 Calcul du champ GGVF	48
III.10 contours actifs géométrique	49
III.10.1 Courbe se déformant suivant sa normale	49
III.11 les contours actifs géodésique	51

III.12 Conclusion	51
-------------------------	----

CHAPITRE IV

l'approche adoptée

IV.1 Introduction.....	53
IV.2 Applimentation.....	53
IV.2.1 Transformation et affichage d'image en niveau de gris.....	55
IV.2.2 Initialisation automatique	55
IV.2. 2.1 Etude de l'histogramme	56
IV.2.2.2 Traitements de bas niveau	57
a) Le seuillage	57
b) Détection de contour	58
c) Le filtrage morphologique	59
d) Calcul de la carte contour	59
IV.2.3 Choix du snake	60
IV.2.4 Calcul du champ GVF	60
IV .2.5 Interpolation linéaire	61
IV.2.6 Déformation du snake	61
IV.2.7 Critère d'arrêt.....	62
IV.3 conclusion	62

Chapitre V

tests et résultats

V.1 Introduction	64
V.2 La tumeur	64
V.3 Aplication de l'apprche sur une image IRM	65
V.3.1 Initialisation manuelle.....	66
V.3.2 Initialisation automatique.....	68
V.3.3 Interprétation des résultats	69
V.4 Conclusion.....	70

Conclusion générale	72
---------------------------	----

Annexes	
---------------	--

Bibliographie.....	
--------------------	--

Introduction générale

Introduction Générale

Parmi les moyens qu'utilise l'homme pour communiquer et transmettre l'information, se situe l'image. Grâce aux développements des techniques récentes, cette communication a pu être ramenée même jusqu'au monde des machines.

Ces derniers temps, la vision artificielle est un domaine qui occupe une importance de plus en plus considérable. Les systèmes d'imagerie sont de plus en plus performants, de moins en moins chers et l'on remarque les progrès qu'ils induisent dans divers domaines, notamment, dans l'extraction et la reconnaissance d'objets, l'imagerie médicale, etc. Cependant, le problème généralement rencontré dans ce type de systèmes est bien l'interprétation des informations visuelles obtenues par l'analyse.

Analyser une image a pour objectif d'arriver à une description synthétique de l'information brute contenue dans l'image. Plusieurs méthodes de traitement et d'analyse ont été mises en œuvre. Les résultats attendus de chaque méthode sont fonction des traitements effectués et du type d'image traitée.

Le but de notre travail est de réaliser un outil d'analyse et de traitement qui facilite l'extraction d'objets contenus dans une image. En d'autres termes, effectuer une segmentation sur une image. La segmentation est un traitement de bas niveau qui consiste à partitionner une image en régions homogènes et significatives. Cette opération constitue la base de tous les processus d'interprétation d'images. Différentes techniques de segmentation ont été développées. Chacune d'entre elles possède des avantages et des inconvénients.

Notre étude est consacrée à la technique de segmentation d'images par contours actifs (SNAKES), basées sur la détection de contours.

Les SNAKES sont constitués de trois phases qui sont : l'initialisation, l'évolution et l'extraction des contours.

Afin d'aider cette méthode à donner de bons résultats de segmentation, l'image à traiter passe en premier lieu par une étape de prétraitement, qui consiste à préparer l'image en éliminant le bruit et en renforçant le contraste entre les régions adjacentes par l'application des filtres adéquats.

Organisation du mémoire

Ce mémoire est organisé en deux parties : la première présente un état de l'art sur les méthodes de traitement et de segmentation d'images, la seconde décrit l'implémentation et l'évaluation de la méthode des contours actifs dans la segmentation d'images.

L'objectif de la première partie *-Etat de l'art sur les méthodes de traitement et de segmentation d'images-* est de présenter les concepts de bases du traitement d'images et les différentes techniques de segmentation.

Le chapitre I-*Généralités sur le traitement d'images-* s'intéresse aux généralités et quelques définitions sur le traitement d'images.

Le chapitre 2 *-Méthodes de traitement d'images-* présente quelques techniques de prétraitement et de segmentation d'images.

Le chapitre 3 *-Contours actifs -* définit les techniques de segmentation basées sur la détection de contours, à savoir les contours actifs classiques (SNAKE), GVF et GGVf etc.

La seconde partie de ce mémoire intitulée *-Conception et réalisation-* présente une implémentation des contours actifs et leur évaluation sur différents types d'images.

Le chapitre 4 *-Conception et mise en œuvre-* présente en particulier une conception pour l'implémentation des Contours actifs(SNAKES).

Chapitre 5 *-Tests et résultats-* est dédié aux tests et résultats effectués afin d'évaluer notre application.

Pour terminer, une conclusion générale suivie d'une annexe.

Chapitre I

Notions de base sur

Le traitement d'image

I.1. Introduction

Les méthodes de traitement d'images sont appliquées suivant un système de traitement donné pour atteindre un but précis selon le domaine d'application. Elles sont constituées de différentes étapes à savoir, une étape d'acquisition, une étape de prétraitement et une étape de traitement proprement dite.

Nous présentons dans ce chapitre quelques définitions générales sur l'image. Nous commençons par donner sa composition ainsi que les méthodes de base qui s'appliquent à celle-ci. Par la suite, nous donnerons quelques caractéristiques de l'image.

I.2. Définitions générales

I.2.1. Qu'est-ce qu'une image ? [1]

L'image est définie comme étant la reproduction exacte, ou la représentation analogique d'une scène réelle.

Visuellement, c'est une représentation bidimensionnelle. Elle contient en chaque point l'intensité lumineuse et/ou chromatique perçue par un dispositif d'acquisition (caméra, appareil photo,...). De ce fait, l'image peut être décrite sous forme d'une fonction $I(x, y)$ de brillance analogique continue, tel que x et y sont les coordonnées spatiales d'un point et I une fonction d'intensité lumineuse ou chromatique. Sous cet aspect, l'image est inexploitable par l'ordinateur, d'où la nécessité de sa numérisation.

I.2.2. Le pixel

Le pixel est l'abréviation de « Picture élément », élément d'image. C'est la primitive de bas niveau la plus pauvre en information car ses seuls attributs sont la position dans la matrice image (n lignes, n colonnes) et la valeur numérique indiquant sa couleur, ou son niveau de gris.

Il peut être représenté en mémoire sur :

- Un bit (0/1) pour les images monochromes : 0 pour le noir et 1 pour le blanc.
- Un octet, soit 256 niveaux de gris pour une image à niveaux de gris: 0 noir et 255 blancs.

- Trois octets pour une image couleur (RVB) :

Dans ce cas, la couleur du pixel est la combinaison des trois nuances de chaque couleur.

I.2.3. Les images bitmap

Appelées aussi images raster, il s'agit des images pixellisées, c'est-à-dire un ensemble de points (pixels) contenus dans un tableau, chacun de ces points possède une ou plusieurs valeurs décrivant sa couleur. Parmi ces formats on cite :

1.2.3.1. Le format GIF

Une image GIF peut contenir de 2 à 256 couleurs (2, 4, 8, 16, 32, 64, 128 ou 256). Ainsi grâce à sa palette limitée en nombre de couleurs (et non limité en couleurs différentes), les images obtenues par ce format ont une taille généralement très faible.

1.2.3.2. Le format PNG

Il a été mis au point en 1995 afin de fournir une alternative libre au format GIF Le format PNG permet de stocker des images en noir et blanc, en couleurs réelles (True color), ainsi que des images indexées, faisant usage d'une palette de 256 couleurs.

1.2.3.3. Le format TIF

Le format TIF permet de stocker des images en noir et blanc, aux couleurs réelles (True color, jusqu'à 32 bits par pixels) ainsi que des images indexées, faisant usage d'une palette de couleurs.

1.2.3.4. Le format JPEG

Cette méthode de compression est beaucoup plus efficace sur les images photographiques comportant de nombreux pixels de couleurs différentes .Les fichiers du format JPEG ont pour extension .jpg ou .jpeg

I.3. Types d'image

I.3.1. Image binaire

L'image binaire est représentée sous forme d'une matrice dont les éléments valent 0 ou 1. Les 0 sont représentés par du noir et les 1 par du blanc. Chaque donnée d'un pixel est codée sur un bit.

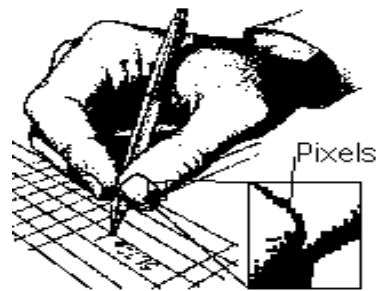


Fig. I.1 : image binaire

I.3.2. Image en niveau de gris

La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux de gris dégradés. Chaque pixel n'est donc plus représenté par un bit, mais par un octet (dans le cas général).



(a)

(b)

(c)

Fig. I.2 : (a) : image à 4 niveaux de gris.
(b) : image à 16 niveaux de gris.
(c) : image à 256 niveaux de gris.

1.3.3. Image couleur R V B

Elle consiste à utiliser 24 bits pour chaque point de l'image. Huit bits sont employés pour décrire la composante rouge (R), huit pour la verte (V) et huit pour la bleu (B). Il est ainsi possible de représenter environ 16.7 millions de couleurs différentes simultanément.



Fig. I.3 : exemple d'une image numérisée avec 16 millions de couleurs

1.3.4. Image indexée

L'image indexée est représentée par un nombre limité de couleurs, ces dernières sont mémorisées dans une table de couleurs (colormap), qui est une matrice de $n \times 3$ (ou n est le nombre de couleurs). L'image est alors une matrice contenant des nombres entiers compris entre 1 et n , chaque entier joue le rôle d'un index relatif à la table de couleurs.

1.3.5. Image numérique

L'image fournie par un capteur est généralement analogique. Pour l'exploiter comme une donnée informatique, il est nécessaire de la numériser afin qu'elle soit compréhensible par l'ordinateur. L'image numérique est constituée de pixels contenant chacun différentes informations (intensités lumineuses, couleurs...). Ces informations seront codées dans une grille échelonnée.

La numérisation de l'image analogique se fait en trois étapes :

I.3.5.1. Balayage [3]

C'est le fait de balayer, ligne par ligne, l'image tout en captant des mesures (luminosité par exemple). Ce balayage nous renvoie le spectre de l'image qui est une fonction continue (fonction de l'image). L'aspect analogique est donc toujours présent. Cette fonction est traitée, par la suite, de façon à nous fournir des valeurs numériques représentatives de l'image.

I.3.5.2. Echantillonnage [3]

Cette opération consiste à prendre des valeurs numériques discrètes par chaque intervalle défini par la taille de l'ouverture de balayage. Ainsi, à chaque intervalle (distance), on garde la valeur de la fonction spectrale résultante du balayage. Pour faciliter la compréhension de cette notion, on dira que cet échantillonnage représente le découpage de l'image d'origine en petits carrés où chaque carré est caractérisé par la mesure évaluée dans l'étape du balayage. Ainsi, l'image sera transformée en une mosaïque de ces petits carrés.

I.3.5.3. Quantification [3]

Il s'agit, durant cette étape, de codifier les valeurs des carrés en nombres entiers qui seront à leur tour codifiés sous forme binaire.

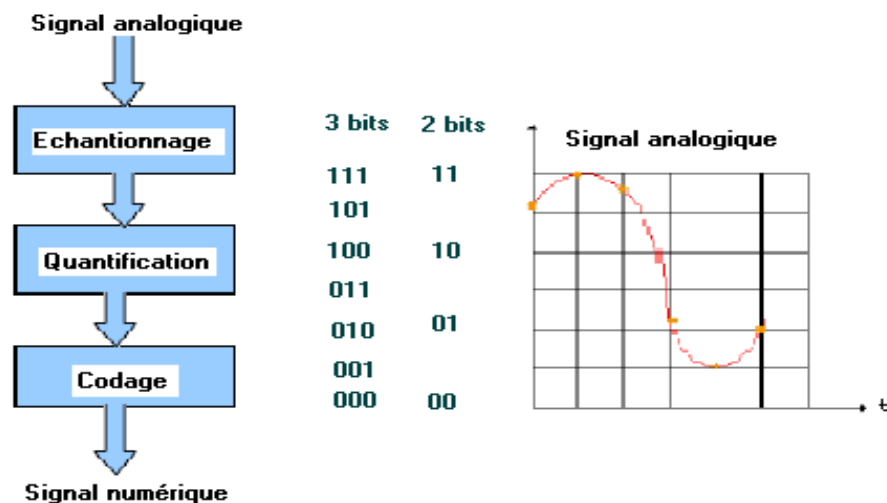


Fig. I.4: Numérisation de l'image.

I.4. Caractéristiques d'une image numérique

I.4.1. Dimension

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image.

I.4.2. Résolution

C'est l'espace ou la densité des pixels dans l'image. Elle se mesure par convention en pixel par pouce ppp (1pouce=2,54cm) ou dpi (dot per inch), plus la résolution est élevée, plus l'image a de pixels.

Ex : image de 3P/3P en 72 PPP=46656 pixels, image de 3P/3P en 300 PPP=810.000 pixels.

I.4.3. Contours

Un contour est un ensemble de pixels formant une frontière entre deux ou plusieurs régions voisines. L'épaisseur du contour est d'un pixel ou plus (un pixel dans le cas idéal).

Une frontière existe s'il y a une discontinuité du niveau de gris entre deux pixels adjacents (une brusque variation du niveau de gris ou de couleur).

Le contour peut être défini comme une marche d'escalier s'il est net, comme une rampe s'il est plus flou, ou comme un toit s'il s'agit d'une ligne sur un fond uniforme.

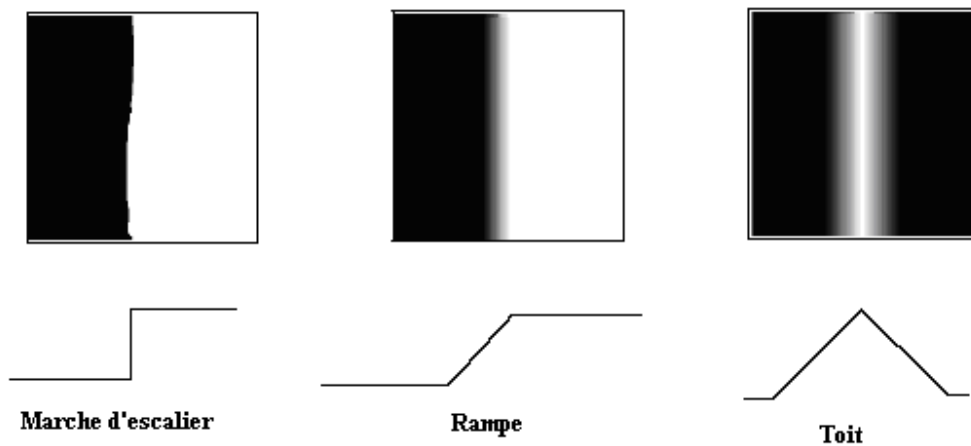


Fig. I.5: Différents type de contours

1.4.4. Luminance

C'est le degré de luminosité des points d'image. Pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet.

1.4.5. Voisinage

Le voisinage est défini comme étant l'ensemble des pixels qui entourent dans toutes les directions possibles (8 directions), un pixel donné de coordonnées (i, j) . Cette notion est utilisée souvent lors du filtrage.

$(i-1, j-1)$	$(i-1, j)$	$(i-1, j+1)$
$(i, j-1)$	(i, j)	$(i, j+1)$
$(i+1, j-1)$	$(i+1, j)$	$(i+1, j+1)$

Fig. I.6 : Tableau de voisinage d'un pixel

1.4.6. Contraste

C'est l'opposition marquée entre les régions sombres et les régions claires d'une image. Le contraste est défini en fonction de luminances de deux zones d'image.

Si $L1$ et $L2$ sont les degrés de luminosité, respectivement de deux zones voisines $A1$ et $A2$ d'une image, le contraste est défini par :

$$c = \frac{L1 - L2}{A1 - A2}$$

1.4.7. Le bruit

Le bruit est un phénomène imprévisible qui vient perturber le signal. Dans une image c'est un phénomène de brusques variations d'intensité d'un pixel par rapport à ses voisins.

Il peut être causé par :

- Les événements inattendus lors de l'acquisition comme le bougé ou une modification ponctuelle des conditions d'éclairage.
- La mauvaise qualité des capteurs ou une mauvaise utilisation de ces derniers
- Lors de l'échantillonnage, Le passage de la forme analogique à la forme numérique de l'image.
- Ou bien la nature de la scène elle même (poussières, rayures, perturbation atmosphérique,...).

1.5. Histogramme [4]

L'histogramme d'une image en niveau de gris est un graphe $H(x, y)$, tel que pour chaque niveau de gris x ($0 \leq x \leq 255$) on associe le nombre d'occurrences de celle-ci, en d'autres termes c'est une visualisation graphique de la relation entre le niveau de gris et le nombre de pixels. L'histogramme d'une image en couleur est un graphe $H(x, c, y)$ tel que pour chaque intensité x ($0 \leq x \leq 255$) de couleur c (c soit rouge, bleu ou bien vert) on associe le nombre d'occurrences de cette combinaison.

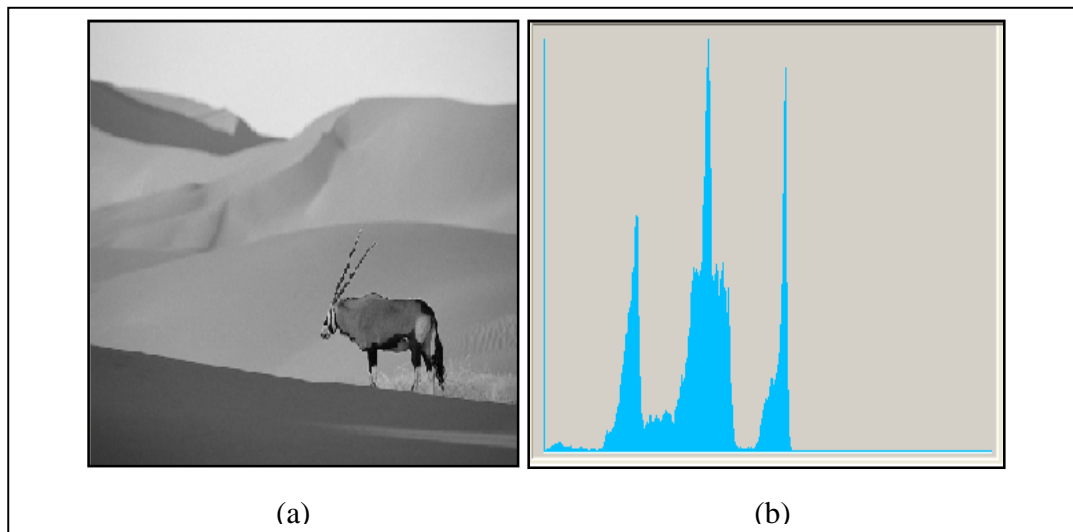


Fig. I.7 : Histogramme d'une image
(a) : Image originale, (b) : son histogramme

I.6.Filtrage

Le filtrage est une opération fondamentale en traitement d'image. Il permet d'améliorer la perception de certains détails, de réduire le bruit et de compenser les défauts du capteur.

On note trois catégories de filtrage : le filtrage linéaire, le filtrage adaptatif et le filtrage morphologique.

I.7. Seuillage

L'opération dite de "seuillage simple" consiste à mettre à zéro tous les pixels ayant un niveau de gris inférieur à une certaine valeur (appelée seuil, en anglais *threshold*) et à la valeur maximale les pixels ayant une valeur supérieure. Ainsi le résultat du seuillage est une image binaire contenant des pixels noirs et blancs, c'est la raison pour laquelle le terme de binarisation est parfois employé. Le seuillage permet de mettre en évidence des formes ou des objets dans une image. Toutefois la difficulté réside dans le choix du seuil à adopter.

Voici une image en 256 niveaux de gris et les résultats d'une opération de seuillage avec les valeurs respectives de seuil de 125 et 200 :



Fig. I.8 : image en 256 niveaux de gris



(a)

(b)

Fig. I.9 : (a) : Image à seuil de 125, (b) : Image à seuil de 200

Chapitre II
Prétraitement et
segmentation

I.1.Introduction

Le domaine du traitement d'images a connu ces dernières années un intérêt de plus en plus croissant, et suscite de part ses différentes applications la convoitise des chercheurs de différents horizons et domaines. L'introduction du traitement d'images dans différents secteurs où l'homme opère s'est avérée des plus nécessaire dans le but de substituer à l'observateur humain la machine, ou au moins l'épauler dans la réalisation de quelques tâches. Étant donné le volume énorme de données qu'un ordinateur peut traiter et en des temps record avec une efficacité déconcertante, il devient nécessaire de faire en sorte que cet outil puisse avoir des données bien formulées et exemptes de toute ambiguïté.

II.2. Prétraitement

Avant de passer à l'analyse d'une image, il est souvent nécessaire d'améliorer sa qualité, et cela dans le but de faciliter l'extraction des différents objets de la scène, en améliorant l'homogénéité des régions qui la composent et en renforçant le contraste entre les régions adjacentes.

Le prétraitement réuni :

1. Élimination du bruit: plusieurs filtres ont été implémentés comme les filtres linéaires et les filtres non linéaires.
2. Renforcement de contraste et l'homogénéité des régions grâce aux méthodes de modification d'histogramme et de rehaussement de contraste.

II.2.1. Réduction du bruit

C'est l'un des sujets les plus délicats du traitement d'images. De nombreuses méthodes lui ont été consacrées, tout d'abord très intuitives, mais progressivement de plus en plus complexes.

Nous verrons dans ce qui suit les approches linéaires, ensuite les non-linéaires

II.2.1.1. Définition

Le filtrage est une opération qui consiste à appliquer une transformation à toute l'image numérique ou à une partie d'elle.

On peut trouver comme filtrages:

- Filtrage linéaire : L'opération est linéaire (passe-bas, passe-haut).
- Filtrage non linéaire : L'opération n'est pas linéaire.
- On peut trouver aussi une combinaison de filtre passe-bas et passe-haut (sobel, perwitt,..).

II.2.1.2. Filtres linéaires

Dans de nombreux cas on considère que l'image non dégradée $f(x, y)$ est affectée d'un bruit additif $n(x, y)$ tel que : $g(x, y) = f(x, y) + n(x, y)$

Le cas d'un bruit multiplicatif est beaucoup plus difficile, il est par exemple traité dans le cas de l'imagerie radar.

Dans des conditions plus simples, on suppose le signal stationnaire, et on adopte souvent une approche par filtrage linéaire. Intuitivement, et en absence de toutes autres informations, on recherche des filtres passe-bas, éliminant les variations à très haute fréquence du signal. Expérimentalement ces filtres ne sont pas très bons car ils dégradent considérablement les contours et rendent l'image floue.

II.2.1.2.1. Filtre passe-bas

Les filtres passe bas adoucissent les détails d'une image (lissage d'images) et réduisent les bruits granuleux .On trouve entre autres :

a) Filtre de moyenne

C'est un cas particulier de filtre de convolution « passe-bas », il agit en remplaçant chaque pixel par la moyenne des valeurs des pixels adjacents et du pixel central.

b) Filtre gaussien

Le filtre Gaussien est un filtre isotrope spécial avec des propriétés mathématiques bien précises. La fonction Gaussienne est très commune dans la nature et elle est aussi souvent utilisée dans les distributions statistiques, elle est définie par la fonction $G(x, y)$

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (\text{II.1})$$

Le paramètre sigma s'appelle la déviation standard, et détermine la largeur de la cloche Gaussienne.

En général un filtre Gaussien avec un $\sigma < 1$ est utilisé pour réduire le bruit, et si $\sigma > 1$, c'est dans le but de fabriquer une image qu'on va utiliser pour faire un "masque flou". Il faut noter que plus le sigma est faible, plus la cloche Gaussienne est large et plus le flou appliqué à l'image sera marqué.

Les propriétés de réduction de bruit des filtres Gaussiens peuvent être utilisées en combinaisons avec d'autres filtres, qui au contraire génèrent du bruit comme les filtres Laplacien. On peut par exemple choisir d'appliquer un filtre Gaussien pour réduire le bruit, avant d'appliquer un filtre Laplacien pour détecter les points autour desquels les variations de luminosité sont importantes.

Exemple

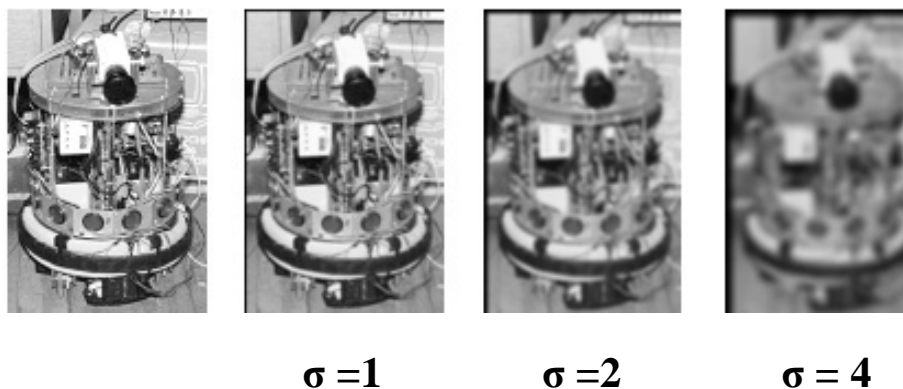


Figure II.1 : Exemple d'utilisation du filtre gaussien.

II.2.1.2.2. Filtre passe-haut

L'application principale des produits de convolution est la création des filtres « passe haut » et « passe bas ». Un filtre « passe haut » favorise les hautes fréquences spatiales, comme les détails, de ce fait, il améliore le contraste. Un filtre « passe haut » est caractérisé par un noyau comportant des valeurs négatives autour du pixel central.

Mais un examen attentif de l'image traitée montre des effets secondaires au traitement:

- **Augmentation du bruit** : Particulièrement dans les images avec un rapport Signal Bruit faible, le filtre augmente le bruit granuleux dans l'image.
- **Effet de bord** : Il est possible que sur les bords de l'image apparait un cadre. Mais cet effet est souvent négligeable et peut s'éliminer en tronquant les bords de l'image.

II.2.1.3. Filtres non-linéaires (filtre médian)

Le filtre médian est utilisé pour atténuer des pixels isolés, d'une valeur très différente de leur entourage. C'est le cas par exemple des pixels thermiques chauds ou froids qui ont une intensité très différente de leurs voisins. Le filtre médian n'est pas un produit de convolution, mais sa mise en œuvre sur l'image est assez similaire puisqu'un noyau est appliqué sur l'image et collecte les valeurs des pixels. La valeur médiane d'une série est par définition celle qui sépare l'échantillon en deux parties de population égale. Son principe est le suivant :

- c) En chaque pixel de l'image, former une liste d'intensités des pixels voisins.
- d) Ordonner cette liste par ordre croissant.
- e) Attribuer à ce pixel la valeur médiane.

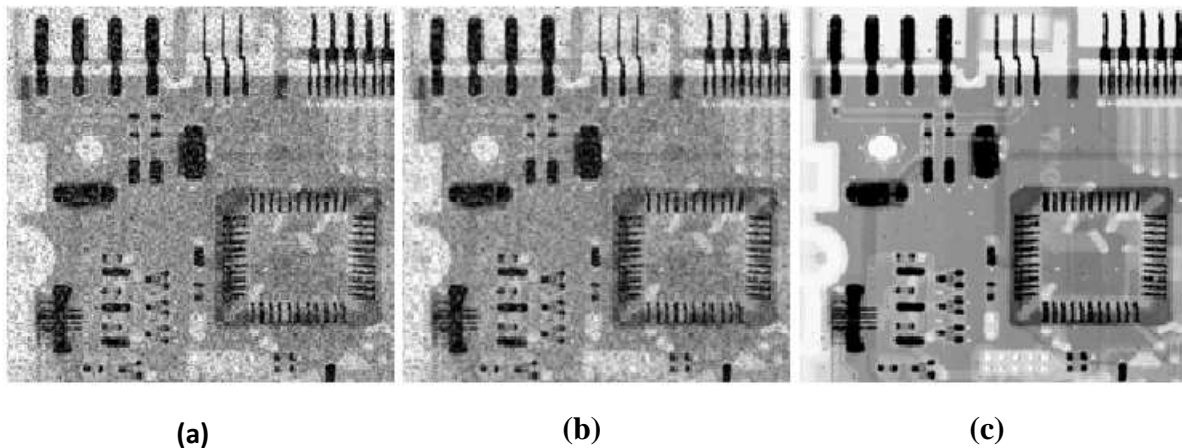


Figure II.2: résultat du filtre moyen et le filtre médian:

- (a) Image bruitée d'un circuit électronique
- (b) résultat de l'application du filtre moyen
- (c) résultat de l'application d'un filtre médian 3x3

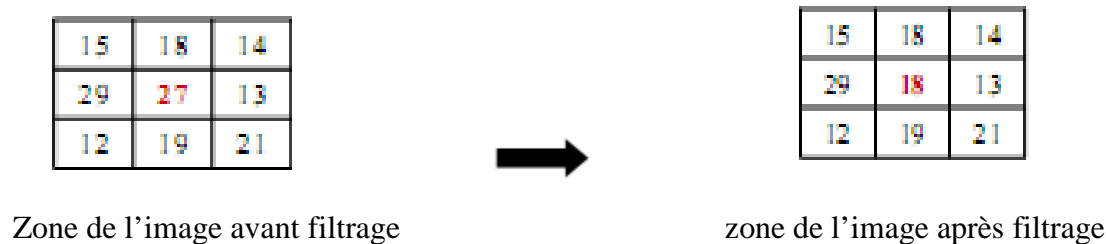


Figure II. 3: Exemple de filtrage médian

II.2.1.4. les filtres morphologiques [7]

La morphologie mathématique a comme principe de base de comparer les objets d'une image A à un certain objet B pris comme référence (forme géométrique appelée élément structurant). Le filtre morphologique est utilisé pour éliminer les pixels isolés considérés comme un bruit. Les opérations de base sont : la dilatation et l'érosion, à partir desquelles on peut construire des outils plus avancés, tels que l'ouverture et la fermeture.

Avant de définir les différents opérateurs de morphologie mathématique, nous tenons d'abord à définir l'élément structurant.

- **Elément structurant** : un élément structurant est une forme de référence qui peut être quelconque, et se déplace sur un ensemble de données.

Les différentes opérations de la morphologie mathématique sont définies comme suit :

a) Erosion

L'érosion est l'opération duale de la dilatation. Les points blancs d'une image sont noyés par les zones noires qui se trouvent autour. Elle est définie par l'équation suivante :

$$E_B(x, y) = \inf_{i, j \in B} I(x + i, y + j) \quad (\text{II.2})$$

b) Dilatation

Elle permet la suppression des points noirs (haute fréquence) isolés au milieu des parties blanche. Elle est définie par l'équation suivante :

$$D_B(x, y) = \sup_{i, j \in B} I(x + i, y + j) \quad (\text{II.3})$$

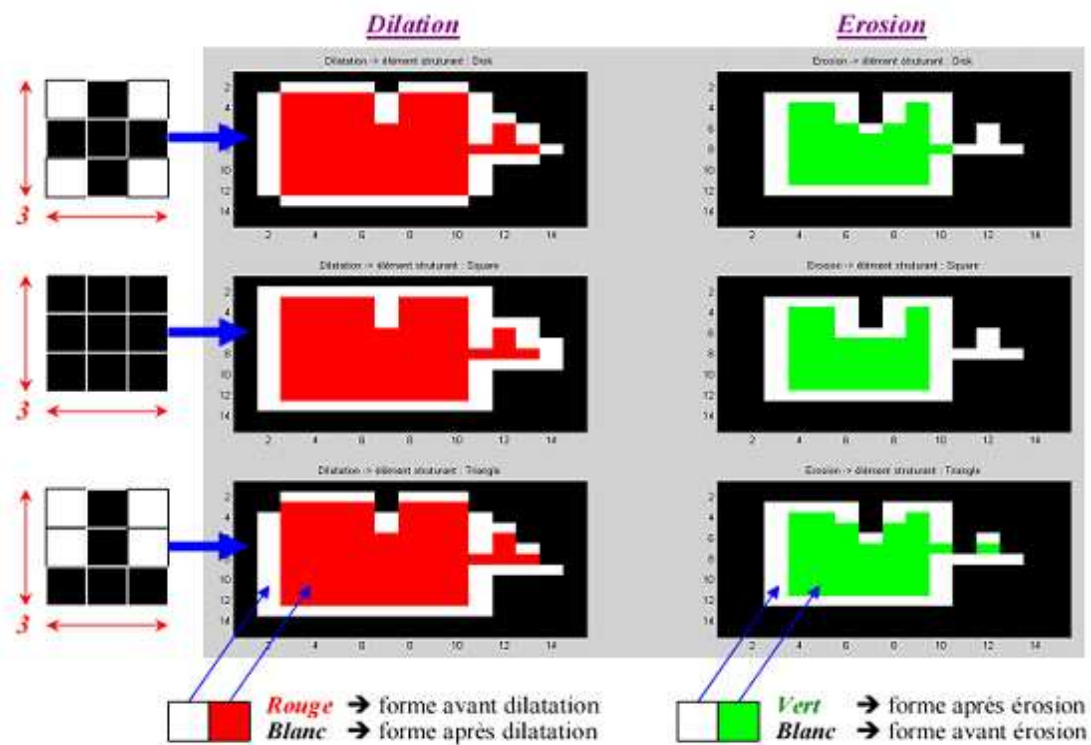


Figure II.4 : opérateurs morphologiques d'érosion et de dilatation avec les 3 éléments structurants.

A partir de ces deux opérations de base, d'autres opérations ont été définies :

c) *Ouverture*

L'ouverture est constituée par une opération d'érosion suivie d'une dilatation.

d) *Fermeture*

La fermeture est constituée par une opération de dilatation suivie d'une érosion.

$$F_B(I) = E_B(D_B(I)) \quad (\text{II.4})$$

II.3.les différentes opérations appliquées sur l'histogramme

II.3.1. Modification d'histogramme

L'histogramme est un outil qui permet d'étudier la répartition des composantes d'une image. Sa modification est généralement, représentée sur une courbe appelée une courbe tonale.

Le but de cette modification est d'optimiser la dynamique des événements à analyser en effectuant une transformation d'intensité de chaque pixel. Les méthodes de modification d'histogramme n'affectent pas la forme des régions, elles modifient uniquement l'apparence visuelle.

II.3.2. Expansion de la dynamique

Permet de distribuer selon une échelle les niveaux de gris uniformément. Pour une image I dont les niveaux de gris sont compris entre G_{min} et G_{max} , alors, après le traitement on aura I' comme suit:

$$I'(x, y) = \frac{I(x, y) - G_{min}}{G_{max} - G_{min}} \quad (\text{II.5})$$

II.3.3. Egalisation d'histogramme

Rendre le niveau du gris équiprobable donc, rendre l'histogramme des niveaux de gris dans l'image, plus plat.

On part du niveau le plus bas et on calcule la valeur cumulée des niveaux jusqu'à une valeur moyenne.

Pour chaque pixel le niveau de gris est calculé comme suit :

$$G' = \frac{255}{\text{nombre pixel total}} \text{Histocumulé}(G) \quad (\text{II.6})$$

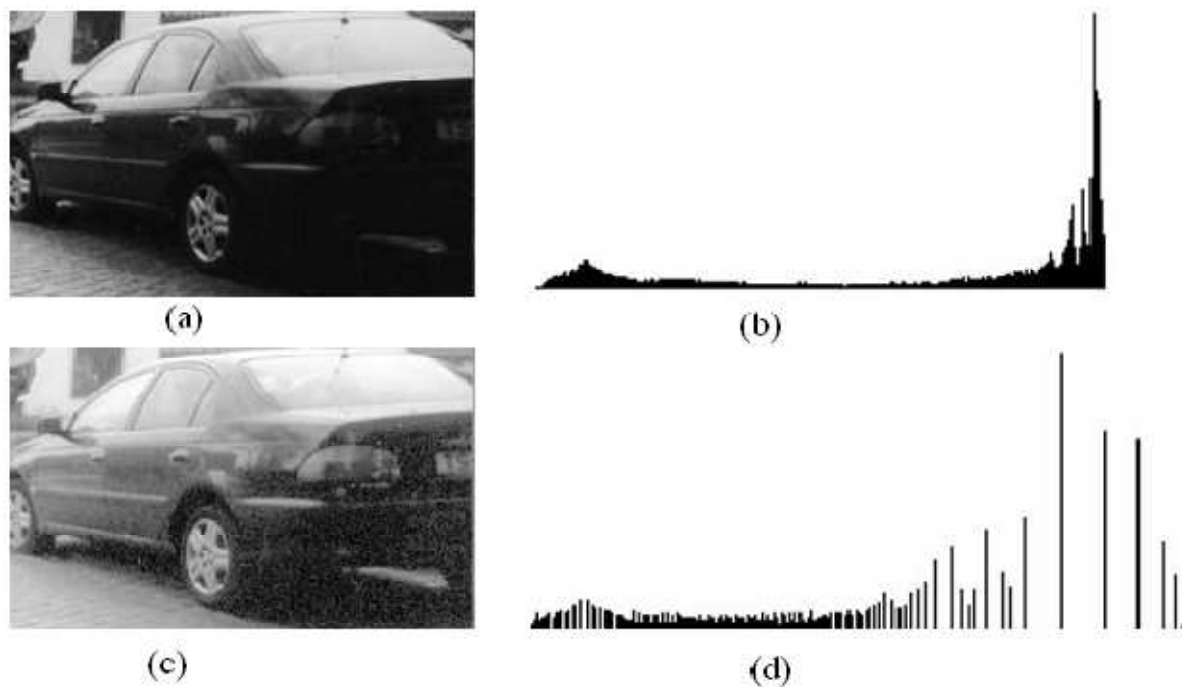


Figure. II.5: Egalisation de l'histogramme :

- (a) Image originale
- (b) histogramme
- (c) Image égalisée
- (d) histogramme égalisé.

II.4. La segmentation

La segmentation est un traitement de bas niveau qui consiste à partitionner l'image en un certain nombre de régions disjointes et de séparer les différentes zones homogènes.

Pour procéder à la segmentation d'une image, l'ordinateur traite l'image comme un ensemble de points ayant chacun un niveau de gris voir un niveau de couleur.

Il existe plusieurs méthodes de segmentation d'images, mais le choix d'une bonne méthode est lié à plusieurs facteurs :

- La nature de l'image (présence de bruit, contours fins,...)
- Les primitives à extraire
- Les opérations en aval de la segmentation

- Les contraintes d'exploitation

Une bonne méthode de segmentation permettra d'arriver à une bonne interprétation soit la quatrième étape du processus, et de simplifier l'image sans pour autant en avoir trop réduit le contenu.

Si I est une image composée de N sous ensembles (I_1, I_2, \dots, I_N) formant une partition et P un prédicat d'uniformité, alors :

- L'union de toutes les régions nous donne l'image initiale.
- Deux régions différentes sont disjointes.
- Dans une région donnée, chaque pixel satisfait le critère d'homogénéité.
- Deux régions différentes sont homogènes.

Mathématiquement elles se formulent par:

- $I = \bigcup_i^n I_i$
- $\forall i, \forall j \text{ et } i \neq j \rightarrow I_i \cap I_j = \emptyset$
- $\forall i \text{ tel que } i = 1 \dots n \rightarrow P(I_i) = \text{vrai}$
- $\forall i, \forall j : i = 1 \dots n \text{ et } j = 1 \dots n \rightarrow P(I_i \cup I_j) = \text{faux}$ I_i et I_j étant deux régions voisines est prédicat d'homogénéité.

II.5. Les approches de la segmentation

Lors d'une étude sur une image, l'objectif est de rechercher une particularité dans l'ensemble, ou dans une partie de l'image. Pour une application quelconque, il s'agit d'extraire les informations pertinentes.

II.5.1. Approche frontière (contour) [6]

Cette technique ne prend pas en considération les relations qui existent entre les régions de l'image (Les pixels sont regroupés en fonction d'un attribut global), Elle comprend les techniques de détection de contour. Mais ces derniers ne conduisent pas toujours directement à la partition recherchée. En effet, les pixels du contour mis en évidence pour une forme, généralement ne sont pas connexes. Il faut alors appliquer les algorithmes de fermeture de contours, pour que les régions apparaissent.

Plusieurs méthodes ont été développées pour ce type d'approche afin d'arriver à une détection qui fournit une représentation correcte de l'image et donner toutes les propriétés significatives de la scène.

II.5.1.1. Méthodes dérivatives

Pour déterminer la position des éventuels contours, on utilise les méthodes dérivatives. L'image dérivée met en évidence les variations de niveaux de gris ; quand il y a une forte variation en général c'est du à un changement de zone.

En imagerie à deux dimensions une scène est modélisée par une fonction de niveau gris $f(x, y)$ et les contours sont assimilés aux points de fort gradient (par la recherche des maximums de la dérivée première) ou au passage de la dérivée seconde par zéro.

Cette méthode est généralement suivi par un seuillage afin d'accentuer les zones de haute intensité des gradients. Le seuillage élimine les transitions de faible gradient qui correspond au bruit. On peut l'effectuer à l'aide de deux techniques:

II.5.1.1.1. Utilisation du gradient

a) *Le gradient*

Le gradient, en un pixel d'une image numérique, est un vecteur caractérisé par son amplitude et sa direction. L'amplitude est directement liée à la quantité de variation locale des niveaux de gris. La direction du gradient est orthogonale à la frontière qui passe au point considéré. La méthode la plus simple pour estimer un gradient est donc de faire un calcul de variation monodimensionnelle, *i.e.* en ayant choisi une direction donnée.

La détection d'un contour se fait par la recherche des maximums du gradient. Soit G la fonction de niveau de gris de l'image à traiter. Le gradient est donné par :

$$\nabla G(x, y) = \left(\frac{\delta G(x,y)}{\delta x}, \frac{\delta G(x,y)}{\delta y} \right) \quad (\text{II.7})$$

En général, on utilise sa norme au lieu d'utiliser ses coordonnées directement pour pouvoir la comparer à un seuil et déterminer où sont les contours. La norme s'exprime par :

$$|\nabla G(x, y)| = \sqrt{\frac{\delta G(x, y)^2}{\delta x} + \frac{\delta G(x, y)^2}{\delta y}} \quad (\text{II.8})$$

Ou :

$$|\nabla G(x, y)| = \max \left\{ \left| \frac{\delta G(x, y)}{\delta x} \right|, \left| \frac{\delta G(x, y)}{\delta y} \right| \right\} \quad (\text{II.9})$$

Ou encore :

$$|\nabla G(i, j)| = \frac{\left\{ \left| \frac{\delta G(x, y)}{\delta x} \right|, \left| \frac{\delta G(x, y)}{\delta y} \right| \right\}}{2} \quad (\text{II.10})$$

Tandis que son orientation est donnée par :

$$\phi(x, y) = \arctang \left[\frac{\frac{\delta G(x, y)}{\delta y}}{\frac{\delta G(x, y)}{\delta x}} \right] \quad (\text{II.11})$$

En pratique, on dispose de masques numériques à balayer sur l'image. Les plus connus sont les opérateurs de Prewitt , ceux de Sobel , Robert et Kirsh.

- **Opérateurs de Sobel et Prewitt**

Les opérateurs de Sobel et de Prewitt permettent d'estimer localement la norme du gradient spatial bidimensionnel d'une image en niveau de gris. Ils amplifient les régions de fortes variations locales d'intensité correspondant aux contours. Ces opérateurs sont constitués d'une paire de masques de convolution 3×3 . L'application séparée par chacun des masques, donne une estimation des composantes horizontales et verticales du gradient par un simple filtrage linéaire avec un masque 3×3 . Il est ensuite possible de calculer la norme et la direction du gradient en chaque point à partir des composantes de ∇_x et ∇_y . La norme du gradient ainsi estimée correspond à l'intensité attribuée au pixel courant.

Les opérateurs de Sobel et Prewitt combinent un léger filtrage passe-bas le long des lignes et des colonnes et un filtrage passe-haut du type gradient.

Nous donnons ici les masques de sobel et perwitt :

Matrices de Roberts	Matrices de Prewitt						Matrices de Sobel																																										
<table border="1"> <tr><td>1</td><td>0</td></tr> <tr><td>0</td><td>-1</td></tr> </table>	1	0	0	-1	<table border="1"> <tr><td>0</td><td>1</td></tr> <tr><td>-1</td><td>0</td></tr> </table>	0	1	-1	0	<table border="1"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-1</td><td>-1</td></tr> </table>	1	1	1	0	0	0	-1	-1	-1	<table border="1"> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table>	1	0	-1	1	0	-1	1	0	-1	<table border="1"> <tr><td>1</td><td>2</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	1	2	1	0	0	0	-1	-2	-1	<table border="1"> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>2</td><td>0</td><td>-2</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table>	1	0	-1	2	0	-2	1	0	-1
1	0																																																
0	-1																																																
0	1																																																
-1	0																																																
1	1	1																																															
0	0	0																																															
-1	-1	-1																																															
1	0	-1																																															
1	0	-1																																															
1	0	-1																																															
1	2	1																																															
0	0	0																																															
-1	-2	-1																																															
1	0	-1																																															
2	0	-2																																															
1	0	-1																																															

Fig. II.6 : Masques de convolution

Exemple de segmentation par l'opérateur de sobel :

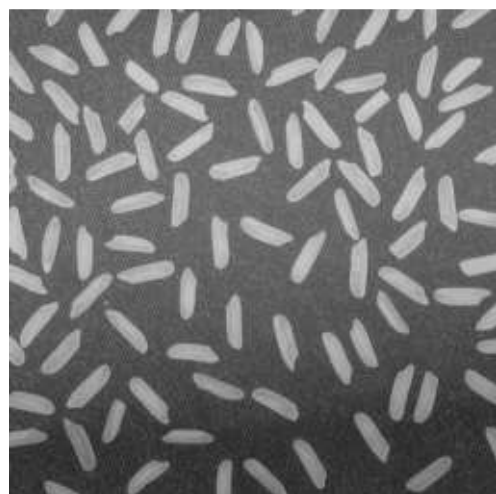


Image originale

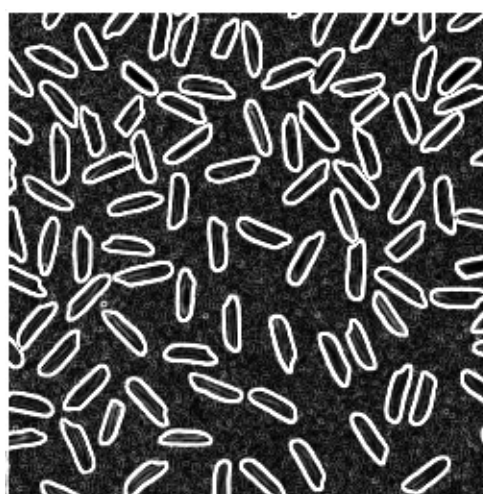


image après traitement

Fig. II.7: Exemple de segmentation par l'opérateur de Sobel

- **Opérateur de Roberts [4]**

Le détecteur de Roberts permet de calculer le gradient bidimensionnel d'une image de manière simple et rapide. Il amplifie les zones (contours) où la norme du gradient spatial est importante. Ce principe ne diffère pas beaucoup de celui des opérateurs de Prewitt et Sobel. Ces masques répondent maximalemt aux contours orientés à 45° par rapport à la grille de

pixels. Par conséquent, l'opérateur de Roberts effectue une estimation locale du gradient selon les deux diagonales.

b) Laplacien

Les opérateurs à dérivée seconde, aboutissent à la détection de brusques variations d'intensité dans les images. Ils permettent de réduire les erreurs de calcul en rejetant les gradients de faible intensité. Les points du contour, qui correspondent à un maximum local, correspondent à des passages à zéro pour la dérivée seconde.

$$\nabla^2 I(i, j) = \frac{\partial^2 I}{\partial i^2} + \frac{\partial^2 I}{\partial j^2} \quad (\text{II.12})$$

c) Seuillage

Après avoir sélectionné les maxima locaux de la carte d'amplitude du gradient, il reste à réduire le nombre de fausses détections de contours. Pour cela, des techniques de seuillage sont mises en œuvre.

- **Seuillage classique**

Le seuillage aide l'utilisateur à faire la différence entre vrai et faux contours dans une image. Toutes les valeurs inférieures à un certain seuil τ donné sont annulées. Le seuillage donne de bons résultats si l'image est de bonne qualité, mais engendre des erreurs lorsque elle est bruitée. Il donne de faux contours lorsque la valeur du seuil τ est trop bas (fausses détections), et des contours incomplets lorsque τ est trop haut.

Pour déterminer un seuil τ adéquat, il faut souvent procéder par tâtonnement.

- **Seuillage par hystérésis**

Le seuillage par hystérésis consiste à appliquer deux seuils s et s' à l'image obtenue après traitement, en faisant en sorte que $s = 2s'$. On obtient ainsi deux images **I1** et **I2** seuillées. Puisque **I2** a été générée avec un seuil élevé, elle contient moins de fausses détections que **I1**, mais présente des contours souvent incomplets. L'objectif de l'algorithme du seuillage par hystérésis est de compléter les contours de l'image **I2** en utilisant les données de **I1**. Lorsqu'un contour se termine en un point (i, j) dans **I2**, on recherche s'il ne peut être

complété en examinant les 8-voisins de ce point (i,j) dans I1. De cette manière, les contours de I1 complètent les contours fragmentés de I2.

II.5.1.1.2. Algorithme de Canny [3]

Pour détecter les contours d'objets présents dans une image, l'algorithme de Canny utilise les cartes de gradient de celle ci. Il permet de satisfaire les trois conditions à respecter lors de la conception des filtres de détection :

- i) qualité de la détection des vrais points de contour,
- ii) localisation des points détectés,
- iii) unicité de la solution (en cas de détection multiple).

La prise en compte de ces trois critères par l'opérateur, fournit une première approximation de la fonction de Gauss $g(i)$, proposée par Canny :

$$G(x, y, \sigma) = \left(1/\sqrt{2\pi\sigma}\right) e^{-(x^2+y^2)/2\sigma^2} \quad (\text{II.13})$$

La dérivée de l'image $I(x, y)$ filtrée est :

$$I_f(x, y) = \nabla(G(x, y) \otimes I_i(x, y)) \quad (\text{II.14})$$

Avec :

$I_i(x, y)$: Image d'entrée

$G(x, y)$: Fonction de Gauss

$I_f(x, y)$: Image de sortie

$$\begin{aligned} I_f(x, y) &= \left(\partial/\partial x + \partial/\partial y\right)(G(x, y) \otimes I_i(x, y)) \\ &= G'_x(x, y) \otimes I_i(x, y) + G'_y(x, y) \otimes I_i(x, y) = f_x + f_y \end{aligned} \quad (\text{II.15})$$

Avec :

$$G'_x(x, y) = \left(-x/\sqrt{2\pi\sigma^3}\right) e^{-(x^2+y^2)/2\sigma^2}, \quad G'_y(x, y) = \left(-y/\sqrt{2\pi\sigma^3}\right) e^{-(x^2+y^2)/2\sigma^2} \quad (\text{II.16})$$

Les filtres étant séparables, nous pouvons réaliser séparément les convolutions suivant x et y:

$$G'_x(x, y) = G'_x(x) \otimes G(y) \text{ et } G'_y(x, y) = G'_y(y) \otimes G(x) \quad (\text{II.17})$$

$$\text{D'où } f_x = G'_x(x, y) \otimes I_i(x, y) = G'_x(x) \otimes G(y) \otimes I_i(x, y) \quad (\text{II.18})$$

$$\text{Et } f_y = G'_y(x, y) \otimes I_i(x, y) = G'_y(y) \otimes G(x) \otimes I_i(x, y) \quad (\text{II.19})$$

La direction et l'amplitude sont alors données par les relations suivantes :

$$A = \sqrt{f_x^2 + f_y^2} \quad \text{et } Dir = \arctan\left(\frac{f_y}{f_x}\right) \quad (\text{II.20})$$

Afin d'obtenir les cartes de la norme des gradients et de leurs orientations locales, on suit ces étapes :

- Filtrage de l'image par un filtre Gaussien dont la variance σ^2 fixe le degré de lissage.
- Estimation de l'amplitude et de l'orientation du gradient en appliquant un filtre de Sobel
- Sélection des maxima locaux.
- Utilisation du seuillage par hystérésis pour détecter et relier les contours.

II.5.1.1.3. Filtre de Deriche [3]

Deriche a cherché une réalisation de l'opérateur sous la forme d'un filtre à réponse impulsionnelle infinie. Pour améliorer l'immunité au bruit, un lissage au préalable est effectué. Le principe consiste à effectuer un lissage qui est la combinaison de deux filtres monodimensionnels dans les directions x et y.

Le calcul du gradient se fait à partir des dérivées selon x et y du produit de convolution de l'image par le filtre de lissage $f(x, y)$.

L'avantage de ce filtre, est qu'il peut être implémenté de manière récursive et qu'il nécessite très peu d'opérations par pixel.

II.5.1.1.4. Filtres de Shen-Castan

Le filtre de Shen et Castan est obtenu par optimisation d'un critère incluant la détection et la localisation. La solution obtenue est assez proche de celle de Canny

$$h(x) = ce^{-\alpha|x|} \quad (\text{II.21})$$

C est choisi de façon à ce que le filtre soit normalisé :

$$c = \frac{1 - e^{-\alpha}}{1 + e^{-\alpha}} \quad (\text{II.22})$$

Le filtre de dérivation s'écrit alors comme suit :

$$h'(x) = \begin{cases} de^{-\alpha|x|} & \text{si } x \geq 0 \\ -de^{-\alpha|x|} & \text{si non} \end{cases} \quad (\text{II.23})$$

d est choisi de façon à ce que le filtre soit normalisé : $d = 1 - e^{-\alpha}$

Le paramètre α intervient dans la détermination de la largeur du filtre, sa diminution a pour effet d'augmenter le lissage de l'image. Le filtre de Shene et Castan donne une localisation très précise des contours, mais il est sensible aux bruits.

II.5.2. Approche région

Pour faire de la reconnaissance de formes à l'aide d'un système de vision, on doit structurer l'image en régions. Une image numérique est en effet avant tout un ensemble de pixels liés les uns aux autres par de simples relations de voisinage. Il est fondamental de pouvoir regrouper au sein d'une même entité tous les pixels voisins les uns des autres (nous parlerons de relation de connexité) de façon à ce que ce groupe de pixels soit homogène.

II.5.2.1 Segmentation par fusion (merge)

On part du principe que chaque pixel représente à lui seul une région. Deux régions seront fusionnées si elles répondent aux doubles critères de similarité des niveaux de gris et d'adjacence de régions. On s'arrête quand le critère de fusion n'est plus vérifié.

II.5.2.2 Segmentation par division (Split)

On peut initialiser tout le processus en considérant que la première région est composée de toute l'image. Si celle-ci ne vérifie pas le critère d'homogénéité, alors on divise cette région en blocs et l'on réitère le processus sur chaque sous-région prise individuellement.

II.5.2.3. L'algorithme Split and Merge [4]

L'algorithme Split and merge (littéralement "diviser et réunir") a été proposé par Horowitz et Pavlidis en 1974. L'image initiale peut être une première partition résultant d'une analyse grossière ou bien l'image brute. Ce processus est décomposé en deux étapes :

Dans la première étape, ou division, on analyse individuellement chaque région X . Si celle-ci ne vérifie pas le critère d'homogénéité, alors on divise cette région en blocs et l'on réitère le processus sur chaque sous-région prise individuellement. On peut tout à fait initialiser le processus en considérant que la première région est composée de toute l'image.

Dans la deuxième étape, ou réunion, on étudie tous les couples de régions voisines (X, Y) . Si l'union de ces deux régions vérifie le critère d'homogénéité, alors, on fusionne les régions. La principale difficulté de cette approche réside bien sûr dans le parcours de l'ensemble de tous les couples de régions voisines.

II.5.3. Approche par classification des pixels

La classification des pixels se fait sur la base de leur ressemblance. On définit un ensemble fini de classes de pixels et on affecte chaque pixel de l'image à l'une de ces classes. Des régions se forment par le regroupement des pixels connectés par la propriété d'appartenance à une même classe.

II.6. Conclusion

Afin de détecter des contours, nous venons de décrire des opérations de bas niveau telles que le filtrage linéaire, mais les résultats sont imparfaits pour diverses raisons. Ce qui rend cette détection difficile à ce niveau, provient de la présence du bruit, de la difficulté de régler les paramètres du détecteur de manière optimale et la présence de contraste variable. Par conséquent, on ne peut réaliser une segmentation sur la base des seuls contours car ceux-

ci sont le plus souvent discontinus, incomplets. Notons que de nombreux contours parasites sont détectés à cause du bruit ou de textures macroscopiques. Différentes méthodes ont été proposées afin d'y remédier. Parmi ces dernières, les contours actifs qui ont devenu de plus en plus populaires, et c'est ce que nous allons étudier dans le chapitre suivant.

Chapitre III
contours actifs

III.1. Introduction

Les contours actifs (appelés aussi snake) constituent une méthode de segmentation introduite en 1988 par Kass et son équipe.

Un contour actif est une forme continue (fermé ou non), composé d'un ensemble de points qui se déplace vers un objet d'intérêt en minimisant une fonctionnelle d'énergie. Le snake commence par une position initiale définie par l'utilisateur (cercle, ellipse, polygone, ou carré) et se déforme.

Les contours actifs peuvent être divisés en trois catégories :

- Les contours actifs fermés (le point de départ et le point d'arrivée sont les mêmes).
- Les contours actifs à extrémités libres (tous les points sont susceptibles de bouger).
- Les contours actifs à extrémités fixes, où les positions du point de départ et du point

D'arriver sont fixes dans le temps.

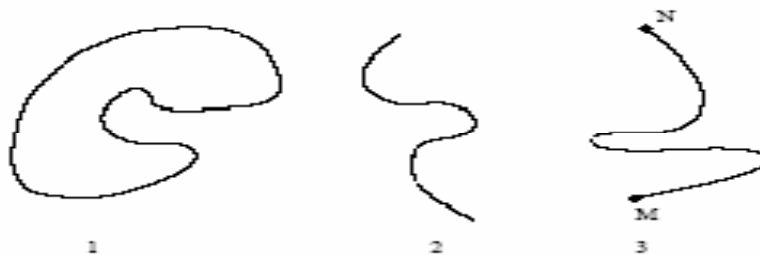


Figure III.1: Types des contours actifs.

III.2. Principe du contour actif [2]

Le principe des SNAKES est de faire évoluer un contour fermé initial autour de l'objet d'intérêt vers une position d'équilibre, c'est à dire en direction des bords de l'objet à détecter. Entre chaque deux position différente du contour (deux itérations), le mouvement des points est fait par une équation impliquant des forces qui agissent sur le contour. Ces forces dépendent des données de l'image (tel que le gradient et l'intensité) et des propriétés du contour (rigidité, élasticité). La position d'équilibre du contour peut être prise comme étant le minimum d'une fonction d'énergie dont la dérivée correspond aux forces à appliquer.

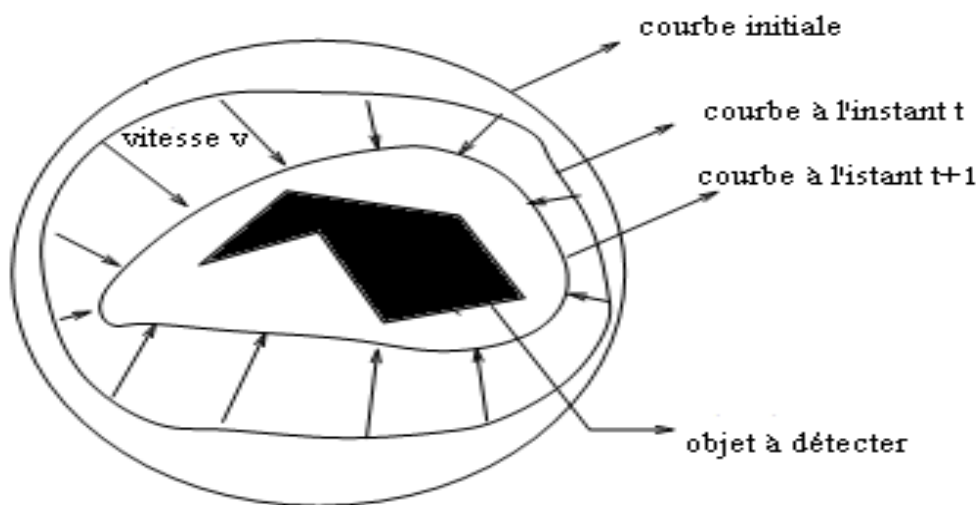


Figure III.2 : principe des contours actif

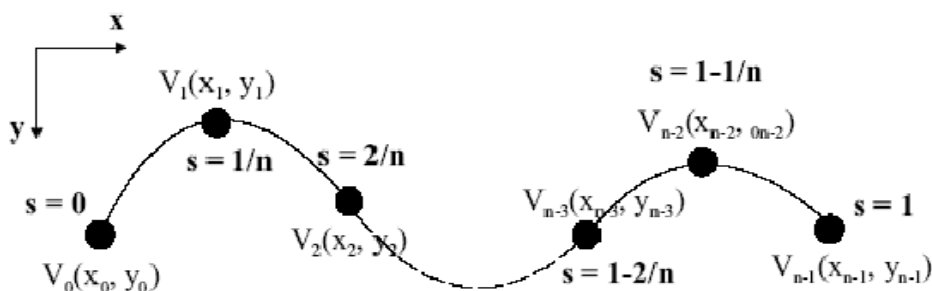


Figure. III.3 : coordonnées cartésiennes et abscisses curvilignes pour un *snake* de n points.

- L'équation d'évolution générale régissant les contours actifs est la suivante :

$$R^2 \rightarrow R^2 c(s, t): [a, b] * [0, T]$$

$$\frac{\partial c(s, t)}{\partial t} = v(s, t) \tag{III.1}$$

La courbe $c(s, t)$ évolue par la vitesse v . Cette dernière dépendra d'une énergie qui déduit l'équation d'évolution.

Donc le problème revient à déterminer v de telle façon à ce que le contour évolue vers le minimum de cette énergie.

Afin d'utiliser les contours actifs pour une application donnée nous devons :

- Définir une fonctionnelle à minimiser pour un problème de donnée.
- Dédire de la fonctionnelle l'équation d'évolution du contour.
- Implémenter cette fonctionnelle avec des méthodes appropriées.

III.3. Le contour actif basé contour (snake)

Le contour actif est défini mathématiquement par l'équation suivante :

$$c(s, t) = \{v(s, t)\} = \{x(s, t), y(s, t)\} \quad 0 \leq s \leq 1: R^2 \rightarrow R^2 \quad (\text{III.2})$$

Avec : $v(s, t)$ est la position d'un point sur la courbe à l'instant t et s l'abscisse curviligne.

III.3.1. L'énergie d'un contour

Cette énergie est composée de deux termes telle que :

$$E_{snake}(v) = E_{int}(v) + E_{ext}(v) \quad (\text{III.3})$$

Avec :

E_{int} : est l'énergie interne qui contrôle l'aspect de la courbe.

E_{ext} : est l'énergie externe qui permet d'attirer la courbe C vers les frontières de l'objet.

III.3.1.1. L'énergie interne

Appelée aussi énergie de régularisation, elle permet un contrôle de la forme de notre courbe. Elle se compose de deux termes :

$$E_{int} = \alpha \int_a^b \left| \frac{\partial v(s, t)}{\partial s} \right|^2 ds + \beta \int_a^b \left| \frac{\partial^2 v(s, t)}{\partial s^2} \right|^2 ds \quad (\text{III.4})$$

Terme 1 : il est en fonction de la première dérivée de la courbe et représente l'élasticité du contour. Sa formulation mathématique est la suivante : $\left| \frac{\partial v(s, t)}{\partial s} \right|^2$

Terme 2 : il est en fonction de la seconde dérivée de la courbe et représente la rigidité du contour. Il s'écrit : $\left| \frac{\partial^2 v(s, t)}{\partial s^2} \right|^2$

Les deux termes sont multipliés par deux coefficients de pondération, respectivement de lissage ($\alpha \geq 0$) et de rigidité ($\beta \geq 0$).

Lorsque la valeur de α est grande, le contour se distend. En revanche pour ($\alpha=0$), le contour peut présenter des discontinuités.

De son côté, lorsque β est grande, le contour tendra vers un cercle si il est fermé, ou vers une droite si il est ouvert. Cependant, lorsque ($\beta=0$), le contour peut prendre une forte convexité.

III.3.1 .2. L'énergie externe

L'énergie externe (ou de contrainte) est définie par l'utilisateur. Elle permet de déterminer les critères d'évolution du contour en fonction de l'image $I(x, y)$. C'est l'énergie potentielle liée à l'image. Elle est donnée par l'équation suivante.

$$E_{ext} = \int_0^1 P(v(s)) ds \quad (III.5)$$

Où P est une fonction d'attraction de contour définie sur le plan image. Généralement, elle est donnée par :

$$P(x, y) = -\lambda |\nabla G_\sigma * I(x, y)| \quad (III.6)$$

Où G_σ est le filtre de Gauss de variance, σ , λ une constante convenablement choisie et ∇ est

l'opérateur de gradient défini par:
$$\nabla = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}. \quad (III.7)$$

III.4. Processus de minimisation [6]

La formulation du problème des contours proposée par Kass ne prend en considération que les énergies de lissage et de rigidité et le gradient comme attaché aux données.

$$E = \alpha \int_a^b \left| \frac{\partial v(s,t)}{\partial s} \right|^2 ds + \beta \int_a^b \left| \frac{\partial^2 v(s,t)}{\partial s^2} \right|^2 ds - k \int_a^b |\nabla I(v(s,t))|^2 ds \quad (III.8)$$

L'évolution du contour revient à la minimisation de l'énergie totale de snake :

$$E_{snake} = \frac{1}{2} \int_0^1 \left[\alpha(s) \left| \frac{dv}{ds} \right|^2 + \beta(s) \left| \frac{d^2v}{ds^2} \right|^2 \right] ds - \lambda \int_0^1 |\nabla G_\sigma * I(x, y)| ds \quad (III.9)$$

Cette équation intégrale est résolue par plusieurs méthodes.

III.4.1. Méthodes d'optimisation [7]

À partir de cette approche théorique, trois implémentations principales peuvent être recensées : l'approche variationnelle classique, peut être la plus développée, introduite par Kass et all en 1987, la méthode par programmation dynamique, introduite par Amini en 1988 et l'utilisation de l'algorithme de glouton ou algorithme de greedy, proposé par william et shah en 1992.

III.4.1.1. Méthodes itératives

Dites aussi méthode variationnelles. C'est la classe de méthodes d'optimisation la plus connue et la plus utilisée dans les modèles déformables. Elle consiste à trouver une courbe qui minimise la fonctionnelle d'énergie décrite dans l'équation (III.9). La courbe résolvant ce problème doit satisfaire l'équation d'Euler-Lagrange suivante :

$$\frac{\partial}{\partial s} \left(\alpha \frac{\partial v}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 v}{\partial s^2} \right) - \frac{\partial E_{ext}}{\partial s} = 0 \quad (\text{III.10})$$

Pour trouver une solution, le contour est rendu dynamique, en traitant $v(s)$ comme une équation du temps $v(s, t)$, représentant la position. La dérivée partielle par rapport au temps est posée égale à l'équation précédente :

$$\begin{cases} \tau \frac{\partial v}{\partial s} = \frac{\partial}{\partial s} \left(\alpha(s) \frac{\partial v}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta(s) \frac{\partial^2 v}{\partial s^2} \right) - \frac{\partial E_{ext}}{\partial s} \\ v(0, s) = v_0(s), \frac{\partial v}{\partial s}(t, 0) = \dot{v}_0(0) \end{cases} \quad (\text{III.11})$$

A l'équilibre, le terme à gauche de l'équation, tend vers zéro puis s'annule. Donnant, ainsi, une solution à l'équation voulue. C'est cette méthode qui fera l'objet de notre travail.

III.4.1.2. Algorithme glouton (greedy) [8]

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.

III.4.1.3. La programmation dynamique [9]

C'est une méthode classique de résolution de problème d'optimisation. Son application aux contours actifs est due à Amini et Al en 1988. La convergence de la minimisation est garantie, mais la complexité est augmentée. .

III.5.Implémentation des snakes par méthode variationnelle [6]

Dans la méthode des contours actifs, il s'agit de minimiser une fonctionnelle d'énergie (équation (III.9)). Comme nous l'avons déjà précisé, le contour correspond à une courbe C^n , définie par un ensemble de vecteur position :

$$v^n(s) = \begin{pmatrix} x^n(s) \\ y^n(s) \end{pmatrix} \quad (\text{III.12})$$

Avec : $\lim_{n \rightarrow \infty} v^n(s) = v^f(s)$

$x^n(s)$ et $y^n(s)$ sont les coordonnées d'un point de C^n dans le plan image. et n est le nombre d'itérations.

- On considère dans ce qui suit que : $\alpha(s) = \alpha = \text{cste}$, $\beta(s) = \beta = \text{cste}$ et $\lambda(s) \approx 1$. En utilisant les coordonnées cartésiennes $x(s)$ et $y(s)$ tel que $v(s) = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix}$, on aura :

$$\dot{x} = \frac{dx}{ds} \quad \dot{y} = \frac{dy}{ds} \quad \dot{v} = \frac{dv}{ds} \quad (\text{III.13})$$

$$\ddot{x} = \frac{d^2x}{ds^2} \quad \ddot{y} = \frac{d^2y}{ds^2} \quad \ddot{v} = \frac{d^2v}{ds^2} \quad (\text{III.14})$$

L'équation (III.10) s'écrit, alors :

$$\begin{cases} \tau \frac{\partial x}{\partial t} = \alpha \ddot{x} - \beta x^{(4)} - \frac{\partial E_{ext}}{\partial x} \\ \tau \frac{\partial y}{\partial t} = \alpha \ddot{y} - \beta y^{(4)} - \frac{\partial E_{ext}}{\partial y} \end{cases} \quad (\text{III.15})$$

Ainsi :

$$\|\dot{v}\|^2 = (\dot{x})^2 + (\dot{y})^2 \quad (\text{III.16})$$

$$\|\ddot{v}\|^2 = (\ddot{x})^2 + (\ddot{y})^2 \quad (\text{III.17})$$

L'énergie interne d'un point du contour actif s'écrit donc :

$$E_{int}(v(s)) = \frac{1}{2}(\alpha \|\dot{v}\|^2 + \beta \|\ddot{v}\|^2) \quad (\text{III.18})$$

Des équations (III.16) et (II.17) on obtient :

$$E_{int}(v(s)) = \frac{1}{2}(\alpha[(\dot{x})^2 + (\dot{y})^2] + \beta[(\ddot{x})^2 + (\ddot{y})^2]) \quad (\text{III.19})$$

Par définition le gradient de l'énergie s'écrit :

$$\nabla E_{snake} = \begin{pmatrix} \frac{\partial E_{snake}}{\partial x} \\ \frac{\partial E_{snake}}{\partial y} \end{pmatrix} \quad (\text{III.20})$$

En introduisant les équations d'Euler $\nabla E_{snake} = 0$, pour le calcul du gradient, on obtient :

$$\begin{cases} \frac{\partial E_{snake}}{\partial x} = \frac{\partial E_{ext}}{\partial x} - \frac{d}{ds} \left[\frac{\partial E_{int}}{\partial \dot{x}} \right] + \frac{d^2}{ds^2} \left[\frac{\partial E_{int}}{\partial \ddot{x}} \right] \\ \frac{\partial E_{snake}}{\partial y} = \frac{\partial E_{ext}}{\partial y} - \frac{d}{ds} \left[\frac{\partial E_{int}}{\partial \dot{y}} \right] + \frac{d^2}{ds^2} \left[\frac{\partial E_{int}}{\partial \ddot{y}} \right] \end{cases} \quad (\text{III.21})$$

D'autre part, de l'équation (III.19), on a :

$$\frac{\partial E_{int}}{\partial \dot{x}} = \alpha \dot{x} \Rightarrow \frac{d}{ds} \left[\frac{\partial E_{int}}{\partial \dot{x}} \right] = \alpha \ddot{x} \quad (\text{III.22})$$

$$\frac{\partial E_{int}}{\partial \ddot{x}} = \beta \ddot{x} \Rightarrow \frac{d^2}{ds^2} \left[\frac{\partial E_{int}}{\partial \ddot{x}} \right] = \beta x^{(4)} \quad (\text{III.23})$$

On fait le même calcul pour la dérivée par rapport à y.

On remplaçant, dans l'équation (III.21), les termes de dérivées par rapport à x et y, on aura :

$$\begin{cases} \frac{\partial E_{snake}}{\partial x} = \frac{\partial E_{ext}}{\partial x} - \alpha \ddot{x} + \beta x^{(4)} \\ \frac{\partial E_{snake}}{\partial y} = \frac{\partial E_{ext}}{\partial y} - \alpha \ddot{y} + \beta y^{(4)} \end{cases} \quad (\text{III.24})$$

De l'équation (III.15) et de l'équation (III.24), on tire :

$$\begin{cases} \tau \frac{\partial x}{\partial t} = - \frac{\partial E_{snake}}{\partial x} \\ \tau \frac{\partial y}{\partial t} = - \frac{\partial E_{snake}}{\partial y} \end{cases} \quad (\text{III.25})$$

Dans ce qui suit, on considère le contour actif comme un ensemble de p points. On choisit une courbe fermée, c'est-à-dire : $V(0) = V(p - 1)$ En utilisant **les différences finies** [Annexe B], les dérivées sont calculées de façon approchée, tel que :

$$\dot{x} = \frac{x_{i+1} - x_i}{h} \simeq x_{i+1} - x_i \quad (\text{III.27.a})$$

$$\ddot{x} = \frac{x_{i-1} - 2x_i + x_{i+1}}{h^2} \simeq x_{i-1} - 2x_i + x_{i+1} \quad (\text{III.27.b})$$

$$x^{(4)} = \frac{x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} + x_{i+2}}{h^4} \simeq x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} - x_{i+2} \quad (\text{III.27.c})$$

Où h est le pas entre deux points (on suppose $h=1$, un pas d'un pixel) et $i = 0, \dots, p-1$.

$$\text{Posant :} \quad \begin{cases} f_x(i) = \left(\frac{\partial E_{ext}}{\partial x} \right)_{x_i} \\ f_y(i) = \left(\frac{\partial E_{ext}}{\partial y} \right)_{y_i} \end{cases} \quad (\text{III.28})$$

Après la discrétisation, les équations d'Euler deviennent :

$$\begin{cases} \left(\frac{\partial E_{snake}}{\partial x} \right)_{x_i} = \frac{\partial E_{ext}}{\partial x} + \alpha(-x_{i-1} + 2x_i - x_{i+1}) + \beta(x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} - x_{i+2}) \\ \left(\frac{\partial E_{snake}}{\partial y} \right)_{y_i} = \frac{\partial E_{ext}}{\partial y} + \alpha(-y_{i-1} + 2y_i - y_{i+1}) + \beta(y_{i-2} - 4y_{i-1} + 6y_i - 4y_{i+1} - y_{i+2}) \end{cases} \quad (\text{III.29})$$

En développant on aura :

$$\begin{cases} \left(\frac{\partial E_{snake}}{\partial x} \right)_{x_i} = f_x(i) + \beta x_{i-2} + (-\alpha - 4\beta)x_{i-1} + (2\alpha + 6\beta)x_i + (-\alpha - 4\beta)x_{i+1} + \beta x_{i+2} \\ \left(\frac{\partial E_{snake}}{\partial y} \right)_{y_i} = f_y(i) + \beta y_{i-2} + (-\alpha - 4\beta)y_{i-1} + (2\alpha + 6\beta)y_i + (-\alpha - 4\beta)y_{i+1} + \beta y_{i+2} \end{cases} \quad (\text{III.30})$$

Ces dernières peuvent être mises sous forme matricielle suivante :

$$\begin{cases} AX + f_x(x, y) = \frac{\partial E_{snake}}{\partial x} \\ AY + f_y(x, y) = \frac{\partial E_{snake}}{\partial y} \end{cases} \quad (\text{III.31})$$

Avec :

$$A = \begin{bmatrix} 2\alpha + 6\beta & -\alpha - 4\beta & \beta & 0 & \dots & 0 & \beta & -\alpha - 4\beta \\ -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta & \beta & 0 & \dots & 0 & \beta \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ -\alpha - 4\beta & \beta & 0 & \dots & 0 & \beta & -\alpha - 4\beta & 2\alpha + 6\beta \end{bmatrix} \quad (\text{III.32})$$

- X est le vecteur composé des abscisses des points du contour actif et Y le vecteur composé des ordonnées :

$$X = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{p-1} \end{bmatrix}, \quad f_x(x, y) = \begin{bmatrix} f_x(0) \\ \cdot \\ \cdot \\ \cdot \\ f_x(p-1) \end{bmatrix}, \quad Y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_{p-1} \end{bmatrix}, \quad f_y(x, y) = \begin{bmatrix} f_y(0) \\ \cdot \\ \cdot \\ \cdot \\ f_y(p-1) \end{bmatrix} \quad (\text{III.33})$$

En écrivant l'équation (III.25) sous la forme discrète, on aura :

$$\begin{cases} \tau \frac{X^{i+1} - X^i}{h} = -\frac{\partial E_{snake}}{\partial x} \\ \tau \frac{Y^{i+1} - Y^i}{h} = -\frac{\partial E_{snake}}{\partial y} \end{cases} \Rightarrow \begin{cases} X^{i+1} = X^i - \frac{h}{\tau} \frac{\partial E_{snake}}{\partial x} \\ Y^{i+1} = Y^i - \frac{h}{\tau} \frac{\partial E_{snake}}{\partial y} \end{cases} \quad (\text{III.34})$$

On posant $\frac{h}{\tau} = \frac{1}{\gamma}$, on aura :

$$\begin{cases} \frac{\partial E_{snake}}{\partial x} = -\gamma(X^{i+1} - X^i) \\ \frac{\partial E_{snake}}{\partial y} = -\gamma(Y^{i+1} - Y^i) \end{cases} \quad (\text{III.35})$$

En remplaçant les coordonnées du gradient de l'énergie dans l'équation (II.31), on obtient :

$$\begin{cases} AX^{i+1} + \lambda f_x(X^{i+1}, Y^{i+1}) = -\gamma(X^{i+1} - X^i) \\ AY^{i+1} + \lambda f_y(X^{i+1}, Y^{i+1}) = -\gamma(Y^{i+1} - Y^i) \end{cases} \quad (\text{III.36})$$

Avec λ le coefficient de pondération introduit afin de compenser des valeurs élevées de l'énergie externe.

Supposant que les valeurs de f_x et f_y varient peu, entre deux itérations. Ce qui permet de faire les approximations suivantes:

$$\begin{cases} f_x(X^{i+1}, Y^{i+1}) \simeq f_x(X^i, Y^i) \\ f_y(X^{i+1}, Y^{i+1}) \simeq f_y(X^i, Y^i) \end{cases} \quad (\text{III.37})$$

L'équation (III.36) devient :

$$\begin{cases} AX^{i+1} + \lambda f_x(X^i, Y^i) = -\gamma(X^{i+1} - X^i) \\ AY^{i+1} + \lambda f_y(X^i, Y^i) = -\gamma(Y^{i+1} - Y^i) \end{cases} \quad (\text{III.38})$$

D'où :

$$\begin{cases} (A + \gamma I_d)X^{i+1} = \gamma X^i - \lambda f_x(X^i, Y^i) \\ (A + \gamma I_d)Y^{i+1} = \gamma Y^i - \lambda f_y(X^i, Y^i) \end{cases} \quad (\text{III.39})$$

Avec I_d la matrice identité. Après inversion de la matrice $(A + \gamma I_d)$, on aura une représentation matricielle déterminant la position des pixels du *snake* discrétisé à implémenter :

$$\begin{cases} X^{i+1} = (A + \gamma I_d)^{-1} (\gamma X^i - \lambda f_x(X^i, Y^i)) \\ Y^{i+1} = (A + \gamma I_d)^{-1} (\gamma Y^i - \lambda f_y(X^i, Y^i)) \end{cases} \quad (\text{III.40})$$

A partir de la formulation (III.40), la prise en compte d'énergie externe, autre que le simple gradient utilisé, se fait en remplaçant, simplement, le terme $f(X^i, Y^i)$ par sa nouvelle expression.

III.6. Avantages et Inconvénients des contours actifs

III.6.1. Avantages [3]

Comme méthode de segmentation, les contours actifs présentent plusieurs avantages. On trouve entre autre :

- Segmenter efficacement des images de différentes natures sans connaissances à priori sur la géométrie des objets à extraire.

- Offrir à l'utilisateur la possibilité d'ajouter à tout moment des potentiels d'attraction afin de contrôler la convergence du contour.

En résumé, le contour actif à seulement besoin d'être initialisé étant donnée que sa convergence est régit par un model mathématique dont le but est la minimisation de l'énergie globale.

III.6.2. Inconvénients [10]

La segmentation par contour actif n'est pas exempte d'inconvénients, on peut en citer trois inconvénients majeurs :

- les contours actifs dépendent de l'initialisation. Si le contour est initialisé loin de l'objet, il a peu de chance, de rejoindre le contour souhaité. De plus, si l'image est bruitée, il a de forte chance de s'arrêter sur des contours parasites.

- Les paramètres d'élasticité et de courbure adaptés à une forme particulière ne sont donc pas utilisables sur une autre forme. On peut remarquer qu'une forte valeur de α induit des contraintes de tension importante et empêche le snake de suivre les contours souhaités. Une forte valeur de β efface les détails du contour. Il est donc nécessaire de procéder à des essais préalables pour déterminer le meilleur jeu de paramètres,

- La faible convergence du contour actif vers les régions de forte concavité. La figure (III.4) montre le problème de convergence à une concavité d'une zone d'intérêt, où on voit que par la méthode des contours actifs traditionnels, ne peut pas aller vers la zone concave.

Lorsque v^t et v^{t-1} sont très proches, on considère que la convergence est réalisée et le processus s'arrête.

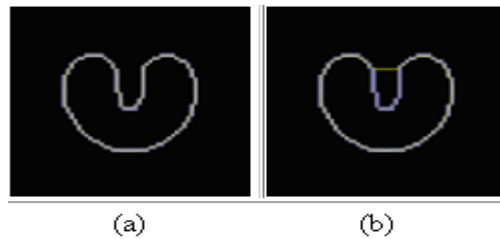


Figure.III.4 :- (a) Image du contour a détecté, (b) Segmentation par le contour actif traditionnel

C'est alors particulièrement sur le problème de l'initialisation et la convergence que des solutions ont été proposées, citons : la force de ballon, la méthode du Champ flux vecteur gradient (GVF) et la méthode du champ flux vecteur gradient généralisé (GGVF).

III.7.la force de ballon

La force de ballon est l'une des méthodes proposées pour améliorer le modèle des snakes, elle a été développée par Laurent D. Elle exerce sur la courbe une force à deux sens (vers l'intérieur ou vers l'extérieur) en chaque point du contour.

Son équation est sous la forme :

$$F_{ballon}(v(s)) = k_1 \vec{n}(v(s)) - k_2 |\nabla I(v(s))|^2 \quad (III.41)$$

Elle est composée de deux forces :

- $fg = k_2 |\nabla I(v(s))|^2$: est une force de gradient de l'image qui a tendance à coller les snakes sur les contours.
- $fb = k_1 \vec{n}(v(s))$: est une force de ballon qui oblige le contour actif à se gonfler par

Le calcul des normales au contour.

III.8. flux du vecteur gradient

La méthode du flux du vecteur gradient (*Gradient Vector Flow GVF*) a été développée dans le cadre d'une approche plus générale en traitement d'image reposant sur le concept du contour actif, et ce dans le but de ramener une solution plus ou moins satisfaisante à certaines limitations de l'approche telles que l'initialisation du snake et sa faible convergence vers les régions concaves.

III.8.1. principe du contour GVF

Pour résoudre le problème des points du *snake* qui se retrouvent piégés par les points des murs de la concavité. *Xu* et *Prince* ont proposé l'utilisation d'une équation de diffusion des vecteurs gradients d'une carte de contours dans des régions distantes des frontières. Cette équation permet d'agrandir le rayon de capture du *snake*. Ainsi, l'information contour est diffusée dans les régions homogènes, ce qui permet une initialisation éloignée du contour. Ceci donne lieu à un nouveau champ de force appelé Flux de Vecteur Gradient (*Gradient Vectors Flow*) qui est un champ vectoriel $w(x, y) = (u(x, y), v(x, y))$ qui minimise la fonctionnelle d'énergie suivante :

$$\varepsilon = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |w - \nabla f|^2 dx dy \quad (\text{III.42})$$

Ou :

- $f = f(x, y)$: Représente une carte de contours de l'image que nous prendrons égale à l'énergie externe.
- μ , est un paramètre de régularisation commandant la différence entre les deux termes de l'intégrale. Ce paramètre se règle en fonction du bruit de l'image, plus le bruit est important plus ce paramètre doit être grand.
- $W = \nabla f$: Est équivalent au gradient de la carte de contour.

Le nouveau champ de force extérieure, W , remplace l'énergie externe de l'équation

$$\tau \frac{\partial v}{\partial t} = \frac{\partial}{\partial s} \left(\alpha(s) \frac{\partial v}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta(s) \frac{\partial^2 v}{\partial s^2} \right) + W \quad (\text{III.43})$$

La méthode GVF procède en deux étapes pour le calcul du champ GVF :

- le calcul de la carte de contours
- le calcul du champ GVF.

III.8.1.1. Calcul de la carte de contours

Pour définir le champ GVF, on commence par présenter la carte de contour (edge map) $f(x, y)$ qui est dérivée de l'image $I(x, y)$.

Trois propriétés générales des cartes de contours sont importantes dans le contexte actuel :

1. Le gradient d'une carte de contours a des vecteurs qui se dirigent vers les frontières, et qui sont normaux aux bords.
2. Ces vecteurs ont de grandes valeurs seulement à proximité des frontières.
3. Dans des régions homogènes, où $I(x, y)$ est presque constant.
 - En raison de la première propriété, si l'initialisation est proche de la frontière à détecter, le contour actif convergera vers une configuration stable près de la frontière.
 - En raison de la deuxième propriété, la gamme de capture sera très petite.
 - Enfin, en raison de la troisième propriété, les régions homogènes n'auront aucune force externe.

Remarque

Les deux dernières propriétés sont indésirables. L'approche de Xu a proposée de garder la première propriété, en rendant l'initialisation loin de la frontière à détecter et dans les régions homogènes, en utilisant un processus de diffusion. Ce dernier va créer des vecteurs qui se dirigent dans les régions concaves.

La carte de contour est définie par :

$$f(x, y) = |\nabla G_\sigma * I(x, y)| \quad (\text{III.44})$$

II.8.2 Le calcul du champ GVF

Le champ GVF est déterminé par le calcul variationnel. On obtient le système d'équations d'Euler suivantes à résoudre :

$$\begin{cases} \mu \nabla^2 u - (u - \nabla_x f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) = 0 \\ \mu \nabla^2 v - (v - \nabla_y f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) = 0 \end{cases} \quad (\text{III.45})$$

Où ∇_x et ∇_y sont respectivement les opérateurs gradient selon x et y .

Ce système est résolu en traitant u et v comme des fonctions du temps et en le résolvant, de la même manière que pour la minimisation de l'énergie du modèle déformable (la méthode variationnelle).

On obtient alors :

$$\begin{cases} \frac{\partial u(x,y,t)}{\partial t} = \mu \nabla^2 u(x,y,t) - (u(x,y,t) - \nabla_x f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) \\ \frac{\partial v(x,y,t)}{\partial t} = \mu \nabla^2 v(x,y,t) - (v(x,y,t) - \nabla_y f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) \end{cases} \quad (\text{III.46})$$

Ces équations peuvent être résolues séparément en u et v . Elles sont connues sous le nom d'équations de diffusion généralisées.

On discrétisant en temps et en espace u et v on obtient :

$$\begin{cases} u(x,y,t+1) = u(x,y,t) + \Delta t \left[\mu \nabla^2 u(x,y,t) - (u(x,y,t) - \nabla_x f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) \right] \\ v(x,y,t+1) = v(x,y,t) + \Delta t \left[\mu \nabla^2 v(x,y,t) - (v(x,y,t) - \nabla_y f) \left((\nabla_x f)^2 + (\nabla_y f)^2 \right) \right] \end{cases} \quad (\text{III.47})$$

Pour simplifier, on réécrit les équations sous la forme suivantes :

$$\begin{cases} u(x,y,t+1) = (1 - b\Delta t)u(x,y,t) + \Delta t \mu \nabla^2 u(x,y,t) + \Delta t c^1(x,y) \\ v(x,y,t+1) = (1 - b\Delta t)v(x,y,t) + \Delta t \mu \nabla^2 v(x,y,t) + \Delta t c^2(x,y) \end{cases} \quad (\text{III.48})$$

Avec :

$$\begin{cases} b(x,y) = (\nabla_x f)^2 + (\nabla_y f)^2 \\ c^1(x,y) = \nabla_x f b(x,y) \\ c^2(x,y) = \nabla_y f b(x,y) \end{cases} \quad (\text{III.49})$$

Le laplacien de u et v peut s'exprimer sous la forme suivante :

$$\begin{cases} \nabla^2 u = \frac{1}{\Delta x \Delta y} (u(x+1,y) + u(x,y+1) + u(x-1,y) + u(x,y-1) - 4u(x,y)) \\ \nabla^2 v = \frac{1}{\Delta x \Delta y} (v(x+1,y) + v(x,y+1) + v(x-1,y) + v(x,y-1) - 4v(x,y)) \end{cases} \quad (\text{III.50})$$

Pour garantir la convergence, le pas en temps doit vérifier :

$$\Delta t \leq \frac{\Delta x \Delta y}{4\mu} \quad (\text{III.51})$$

On voit bien, sur cette condition, que lorsqu'on applique un lissage important (c'est-à-dire, Choisir μ grand) le pas du temps diminue par conséquent, on aura une convergence lente. Lorsqu'on travaille sur une image à faible résolution (c'est-à-dire, Δx et Δy importants) la convergence devienne plus rapide.

III.9.flux vecteur gradient généralisée

III.9.1. définition

Le champ GVF est déterminé par le calcul variationnel de d'équations d'Euler (III. 10). Le lissage excessif de ce champ, engendré par le coefficient μ , près des contours, peut ne pas permettre au *snake* de converge vers les pics fins et longs des contours. Pour remédier à ce problème XU et al ont proposés deux fonctions de poids convenable, à la place des coefficients μ et $|\nabla f|^2$ dans l'équation (III.45), ce qui donne le nom de Flux du vecteur gradient généralisé(GGVF), défini par l'équation suivante :

$$\frac{\partial W}{\partial t} = g(|\nabla f|)\nabla^2 W - h(|\nabla f|)(W - \nabla f) \quad (\text{III.52})$$

$g(|\nabla f|)$: est un **terme de lissage**, permettant l'adoucissement des variations du champ de vecteur.

$h(|\nabla f|)$: est un **terme de donnée**, permettant de rapprocher le champ de vecteur W vers le ∇f .

Ou :

$$g(|\nabla f|) = e^{-(|\nabla f|/k)} \quad \text{Et} \quad h(|\nabla f|) = 1 - g(|\nabla f|) \quad (\text{III.53})$$

III.9.2Calcul du champ GGVF

Les composantes du champ GGVF sont calculées par un calcul variationnel en résolvant l'équation (III.54), qui s'écrit sous la forme développée en fonction de $u(x, y, t)$, et $v(x, y, t)$:

$$\begin{cases} \frac{\partial u}{\partial t} = g(|\nabla f|)\nabla^2 u - h(|\nabla f|)(u - \nabla_x f) \\ \frac{\partial v}{\partial t} = g(|\nabla f|)\nabla^2 v - h(|\nabla f|)(v - \nabla_y f) \end{cases} \quad (\text{III.54})$$

En discrétisant ces équations on aura :

$$\begin{cases} \frac{u(x,y,t+1)-u(x,y,t)}{\Delta t} = g(|\nabla f|)\nabla^2 u(x, y, t) - h(|\nabla f|)(u(x, y, t) - \nabla_x f) \\ \frac{v(x,y,t+1)-v(x,y,t)}{\Delta t} = g(|\nabla f|)\nabla^2 v(x, y, t) - h(|\nabla f|)(v(x, y, t) - \nabla_y f) \end{cases} \quad (\text{III.55})$$

D'où :

$$\begin{cases} u(x, y, t + 1) = u(x, y, t) + \Delta t [g(|\nabla f|)\nabla^2 u(x, y, t) - h(|\nabla f|)(u(x, y, t) - \nabla_x f)] \\ v(x, y, t + 1) = v(x, y, t) + \Delta t [g(|\nabla f|)\nabla^2 v(x, y, t) - h(|\nabla f|)(v(x, y, t) - \nabla_y f)] \end{cases} \quad (\text{III.56})$$

On pose :

$$\begin{cases} b(x, y) = h(|\nabla f|) \\ c^1(x, y) = \nabla_x f b(x, y) \\ c^2(x, y) = \nabla_y f b(x, y) \end{cases} \quad (\text{III.57})$$

En remplaçant dans l'équation (II.58) on obtient :

$$\begin{cases} u(x, y, t + 1) = (1 - b\Delta t)u(x, y, t) + \Delta t g(|\nabla f|) \nabla^2 u(x, y, t) + \Delta t c^1(x, y) \\ v(x, y, t + 1) = (1 - b\Delta t)v(x, y, t) + \Delta t g(|\nabla f|) \nabla^2 v(x, y, t) + \Delta t c^2(x, y) \end{cases} \quad (\text{III.58})$$

Pour garantir la convergence, le pas du temps doit satisfaire la condition suivante :

$$\Delta t \leq \frac{\Delta x \Delta y}{4g_{max}} \quad (\text{III.59})$$

Cependant, avec toutes les améliorations portées sur le *snake*, il reste le problème de changement de topologie.

III.10. contours actifs géométrique

Cette approche proposée par caselle et al ne correspond pas directement à une approche énergétique, elle est basée sur la théorie de l'évolution des courbes planes et des équations aux dérivées partielles (EDP) géométriques.

Cette méthode met en œuvre une courbe se déformant dans le temps et l'espace selon sa normale, à une vitesse proportionnelle à sa courbure pour atteindre les frontières d'un objet à détecter dans une image $I(x, y)$.

III.10.1. Courbe se déformant suivant sa normale

L'équation d'évolution du contour est donnée par :

$$\frac{\partial C}{\partial t} = F \vec{N} \quad (\text{III.60})$$

Tel que :

- C est la courbe.
- \vec{N} la normale intérieure à la courbe.
- F est un terme de vitesse dépendant de la courbure k.

Pour les besoins de convergence de la courbe vers les frontières d'objets, la vitesse dépend également d'une fonction g qui permettra l'arrêt de la convergence dès que les frontières de l'objet à détecter sont atteintes. g dépend du gradient de l'image et tend vers 0 à l'infini et il est

souvent définie ainsi :

$$g(I) = \frac{1}{1+|I|^p} \quad (61)$$

\hat{I} Étant l'image régularisée par un opérateur gaussien et $p = 1$ ou 2 .

On introduit aussi une constante positive v , chargé de forcer l'évolution du contour vers les frontières de l'objet. On peut alors écrire :

$$\frac{\partial C}{\partial t} = g(I)(v + k)N \quad (\text{III.62})$$

Voici la signification de chaque terme de cette équation :

- $\frac{\partial C}{\partial t} = k\vec{N}$ est l'équation de la chaleur, connue pour ses propriétés Géométriques lissantes, Selon la valeur de la courbure k (positive ou négative)
- $k\vec{N}$ est la force attractive ou répulsive comme l'illustre la figure.

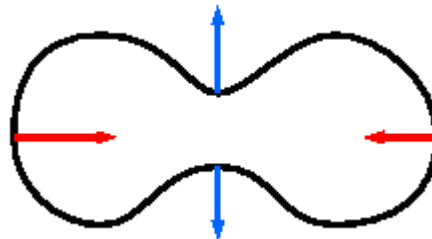


Figure III. 5 : Courbe se déformant selon sa courbure

vN : Définie comme une vitesse constante. Elle agit comme une "force ballon". Elle exerce sur la courbe une force à deux sens (vers l'intérieur ou l'extérieur) en chaque point du contour, et permet à une courbe initialement convexe de devenir non-convexe et de détecter ainsi des objets non-convexes. Cela est nécessaire car en absence de cette force une courbe convexe restera convexe.

$g(I)$: Est une "fonction d'arrêt" attachée aux données de l'image, elle permet d'arrêter la convergence de la courbe quand celle-ci arrive sur les frontières de l'objet à détecter.

III.11. Les contours actifs géodésiques

Les contours actifs géodésiques ont été introduits en 1997 par Caselles, Kimmel et Sapiro comme une extension du modèle des contours actifs «classiques» permettant de remédier à certains inconvénients comme la détection simultanée des contours intérieurs et extérieurs à une

forme. Ils sont essentiellement utilisés pour la détection de contours ou de surfaces, la détection de mouvement..., dans des domaines cruciaux tels que l'imagerie médicale ou satellitaire.

III.12. Conclusion

Dans ce chapitre, nous nous sommes intéressés aux contours actifs (snake) afin d'atteindre notre objectif (détection d'une tumeur cérébrale). Dans une première partie, nous les avons replacés dans le cadre de traitement d'image et particulièrement parmi les méthodes de segmentations. Dans une deuxième partie, pour obtenir de bons résultats, il est donc utile de régler les paramètres (α , β , γ) et de bien initialiser le contour.

Plusieurs solutions ont été proposées, comme l'introduction de force de ballon et calcul du vecteur gradient (GVF).

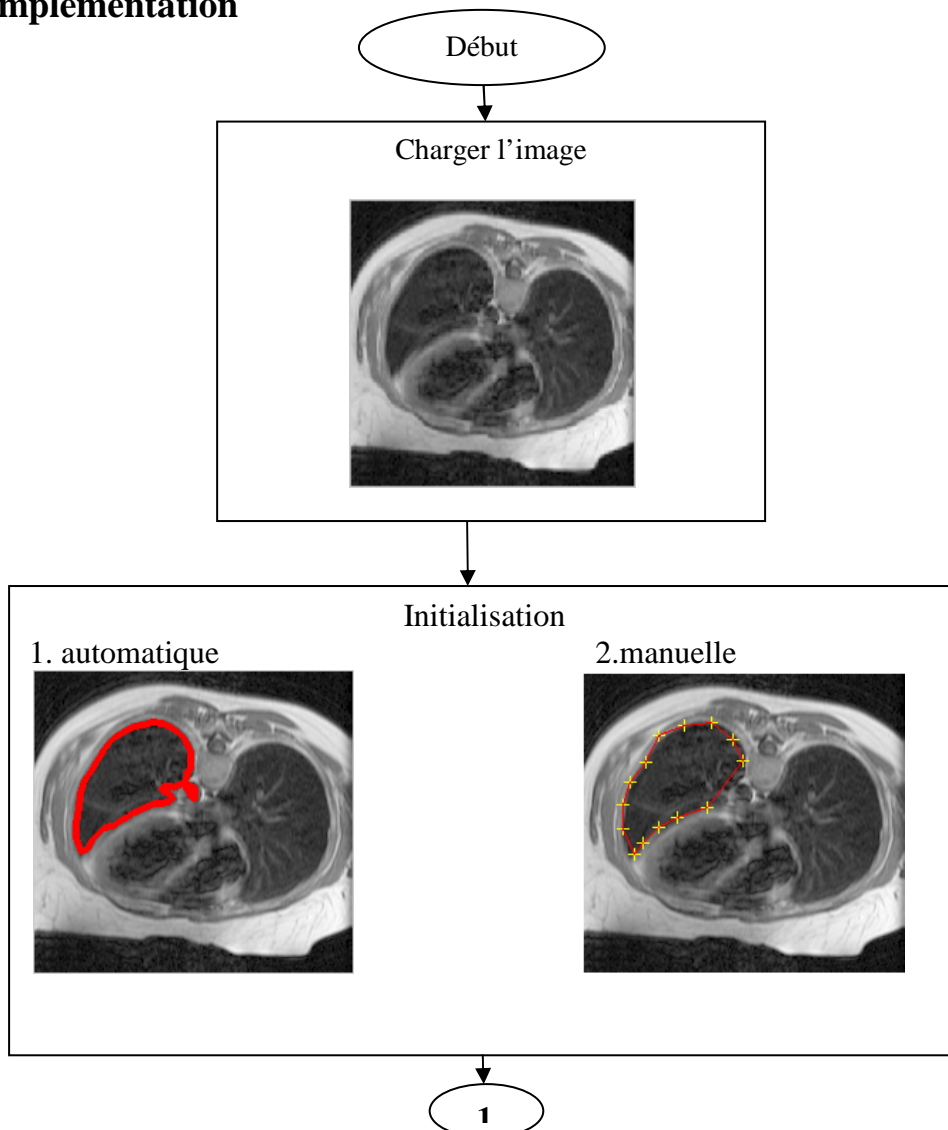
Chapitre IV
conception et realisation

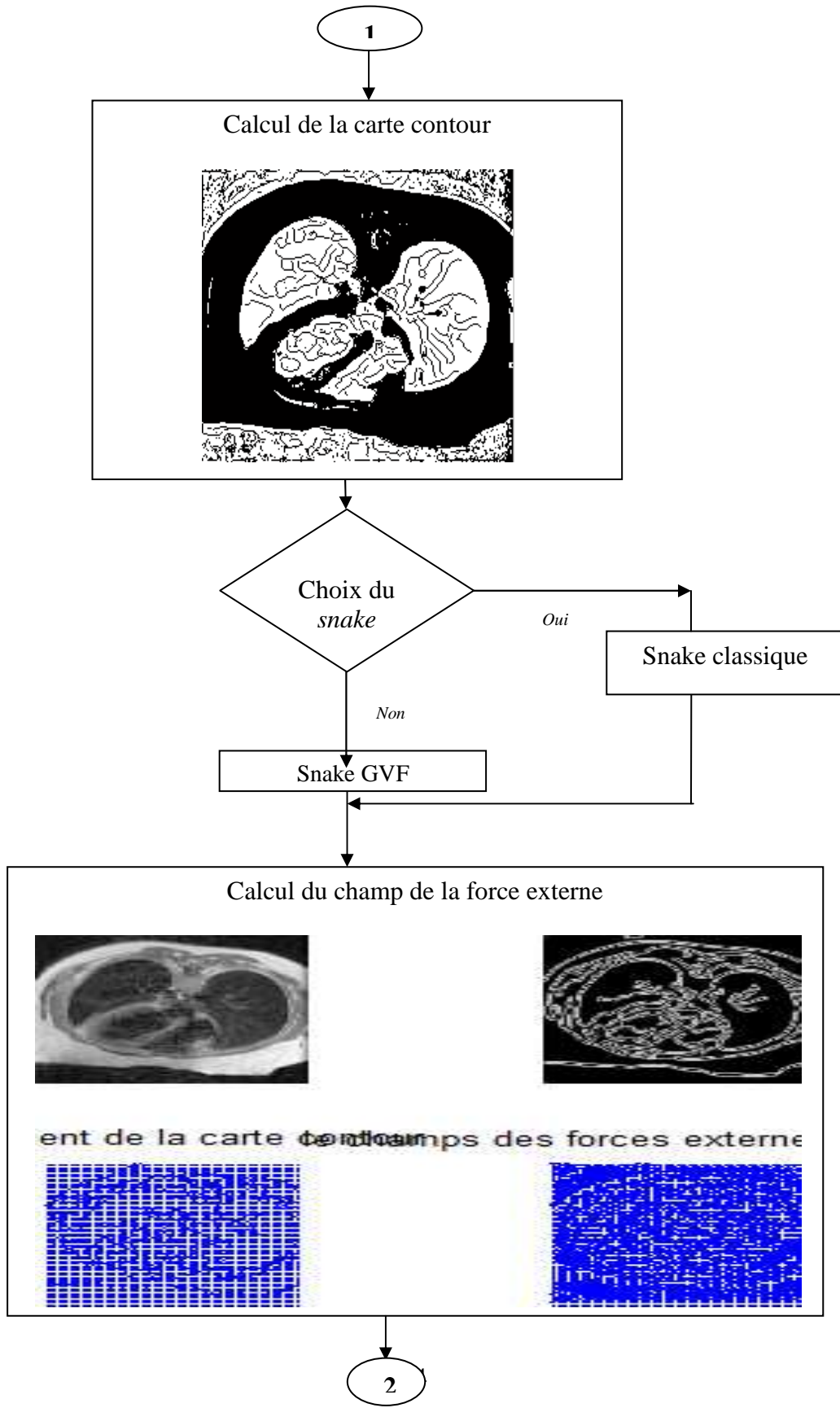
IV.1. Introduction

On a vu durant les trois premiers chapitres les notions et les techniques essentielles auxquelles on aura recours durant le reste de notre travail. Dans le chapitre précédent on a introduit les contours actifs, et on a donné les différentes formules mathématiques régissant leurs évolutions dans le temps.

Afin de montrer l'apport des contours actifs dans la segmentation d'images, et dans le but de faire une comparaison entre le snake classique et le snake GVF, on a programmé sous matlab ces deux méthodes. Dans le présent chapitre on donnera les différents algorithmes et routines utilisées dans notre approche ainsi que les méthodes utilisées pour donner une meilleure estimation des différents paramètres de l'algorithme.

IV.2. Implémentation





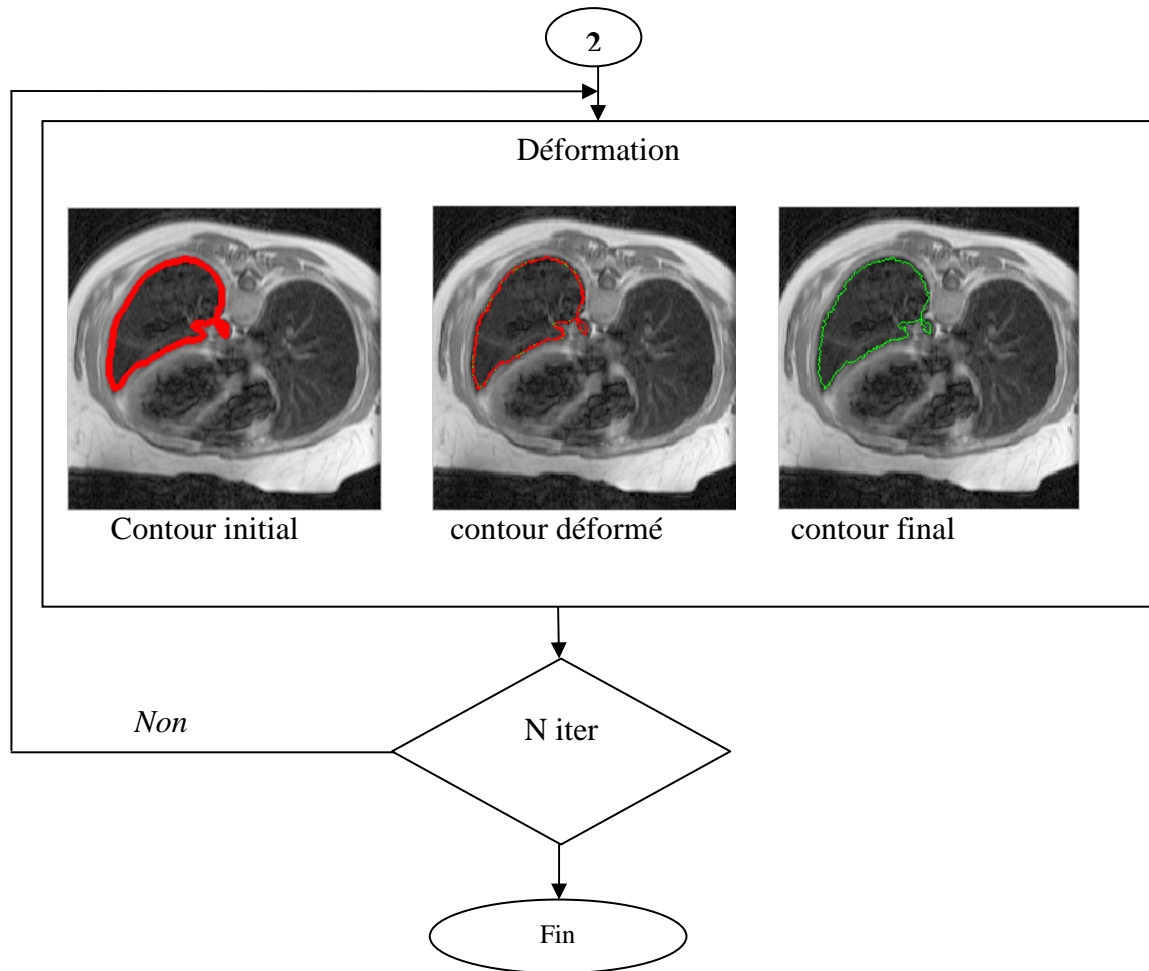


Fig. IV.1 : organigramme de l’algorithme d’implémentation.

IV.2.1. Transformation et affichage d’image en niveaux de gris

Dans cette partie de l’algorithme on affiche l’image $I(x, y)$ par la lecture de sa matrice en utilisant la commande de matlab ($I=imread('X')$). Cette image doit être convertie en niveaux de gris.

X : représente les arguments qui spécifie le chemin de fichier ou se trouve l’image.

IV.2.2. Initialisation automatique

Afin de réaliser une segmentation d’IRM cérébrale, des méthodes de détection de contours par modèle déformable sont très utilisées mais nécessitent une initialisation manuelle proche du contour réel. Pour résoudre ce problème, nous proposons dans notre approche, un

processus d'initialisation automatique, obtenue par une étude de l'histogramme et différents traitements de bas niveau.

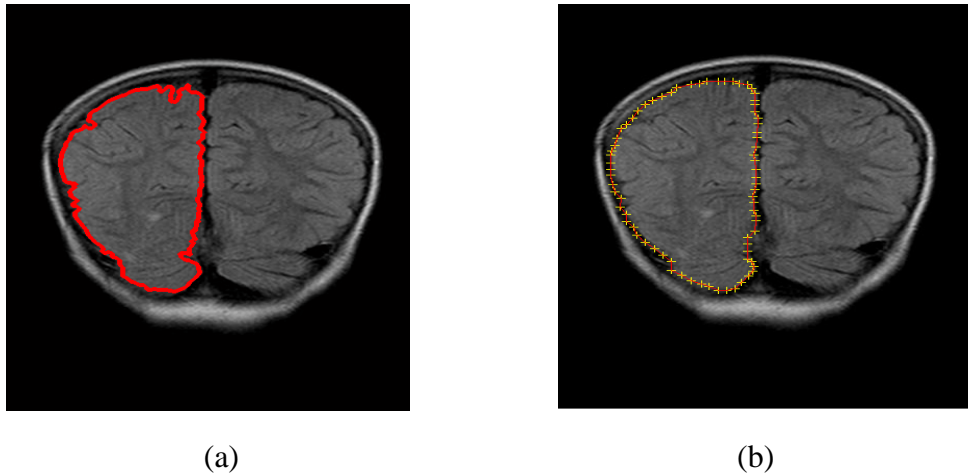


Fig. IV .1 :l'initialisation :(a)initialisation automatique
(b) initialisation manuelle.

IV.2.2.1. Etude de l'histogramme

Nous avons remarqué que l'histogramme d'une IRM est composé de 3 classes de niveaux de gris (Fig. IV.2). Ces classes (le fond, l'os et les vaisseaux sanguins) peuvent être estimées par 3 gaussiennes.

L'histogramme peut alors être modélisé par une somme de densités de ces gaussiennes :

$$G(x) = \sum_{i=1}^{\alpha} w_i \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{\left(\frac{-1}{2\sigma_i^2}(x^2 - m_i^2)\right)} \quad (\text{IV.1})$$

σ_i , m_i et w_i représentent respectivement la variance, la moyenne et le poids de chacune des trois gaussiennes.

Pour ajuster une gaussienne, nous estimons donc ses paramètres (σ_i , m_i , w_i).cette estimation est réalisée en utilisant un algorithme de demélange [c] qui utilise le maximum de vraisemblance [Annexe A].

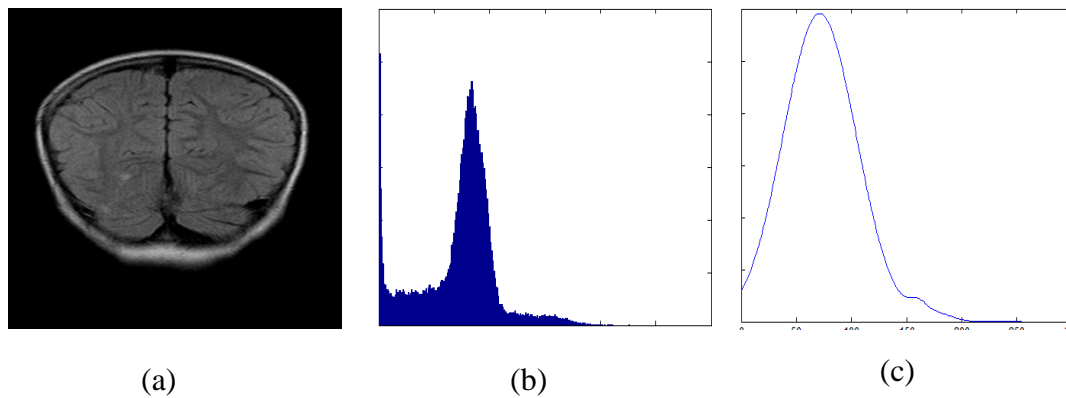


Fig. IV.2 : exemple de modélisation d’histogramme : (a) image originale, (b) histogramme, (c) histogramme estimé

IV.2.2.2. Traitements de bas niveau

Après l’étude de l’histogramme, un ensemble de traitements de bas niveau sont réalisés pour obtenir le contour initial. L’ensemble de ces processus nous permet d’obtenir un contour proche du contour réel. Nous utilisons notamment:

a) Le seuillage

A partir de l’histogramme estimé, nous pouvons définir un ou plusieurs seuils permettant la sélection de la classe de la région d’intérêt (Fig. IV.3).

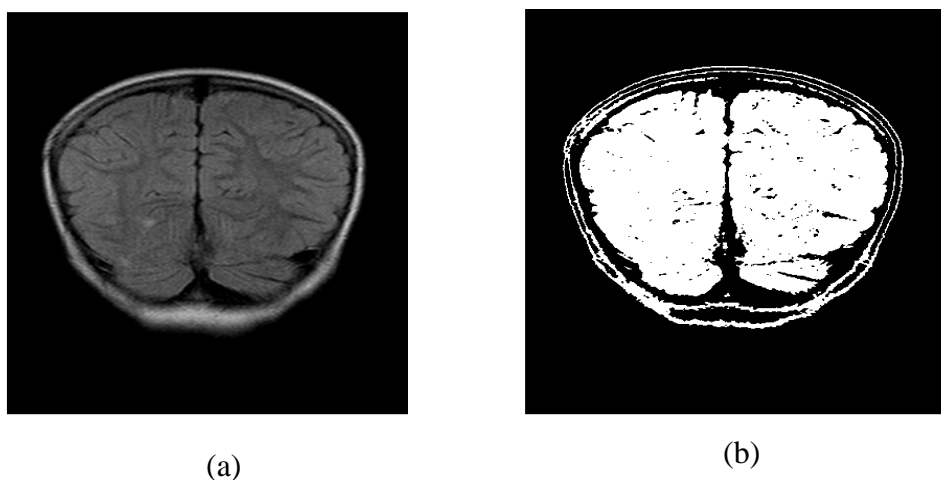


Fig. IV.3 : le résultat après seuillage : (a) image originale et (b) image binaire.

La multiplication de l'image seuillée (binaire) par l'image originale nous permet de mettre en évidence la classe de la région d'intérêt (Fig. IV. 4).

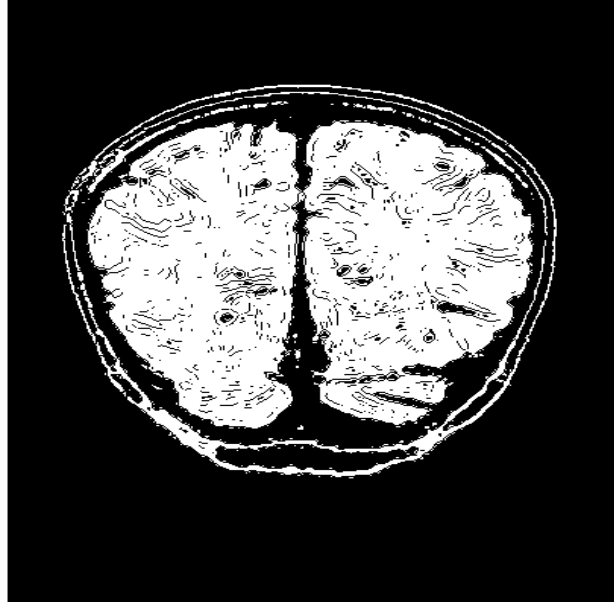


Fig. IV.4 : résultat de la multiplication

b) Détection de contour

Pour détecter les contours de l'image seuillée, nous appliquerons, tout d'abord, un filtrage *médian* qui élimine le bruit de l'image sans dégrader ou modifier ces contours. Ensuite, nous effectuerons le filtre de *canny* pour accentuer les points du contour (Fig. IV.5).

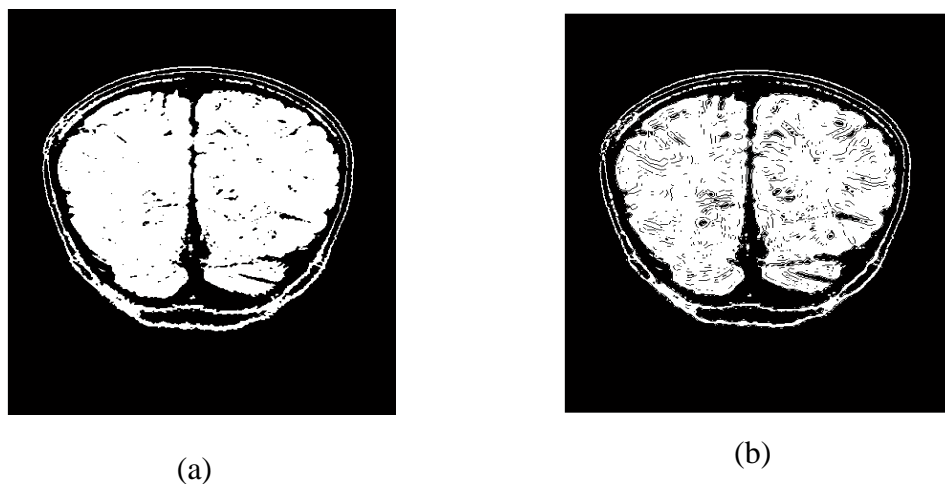


Fig. IV.5 : exemple de détection de contour :

(a) image filtrée et (b) image avec contours accentués et soustrait de l'image binarisée

c) *Le filtrage morphologique*

Après la séparation des différents organes constituant la région d'intérêt, nous réaliserons un remplissage de trous en utilisant un filtre *morphologique* (Fig. IV.5), puis nous combinerons le contour obtenu avec l'image originale (Fig. IV. 6).

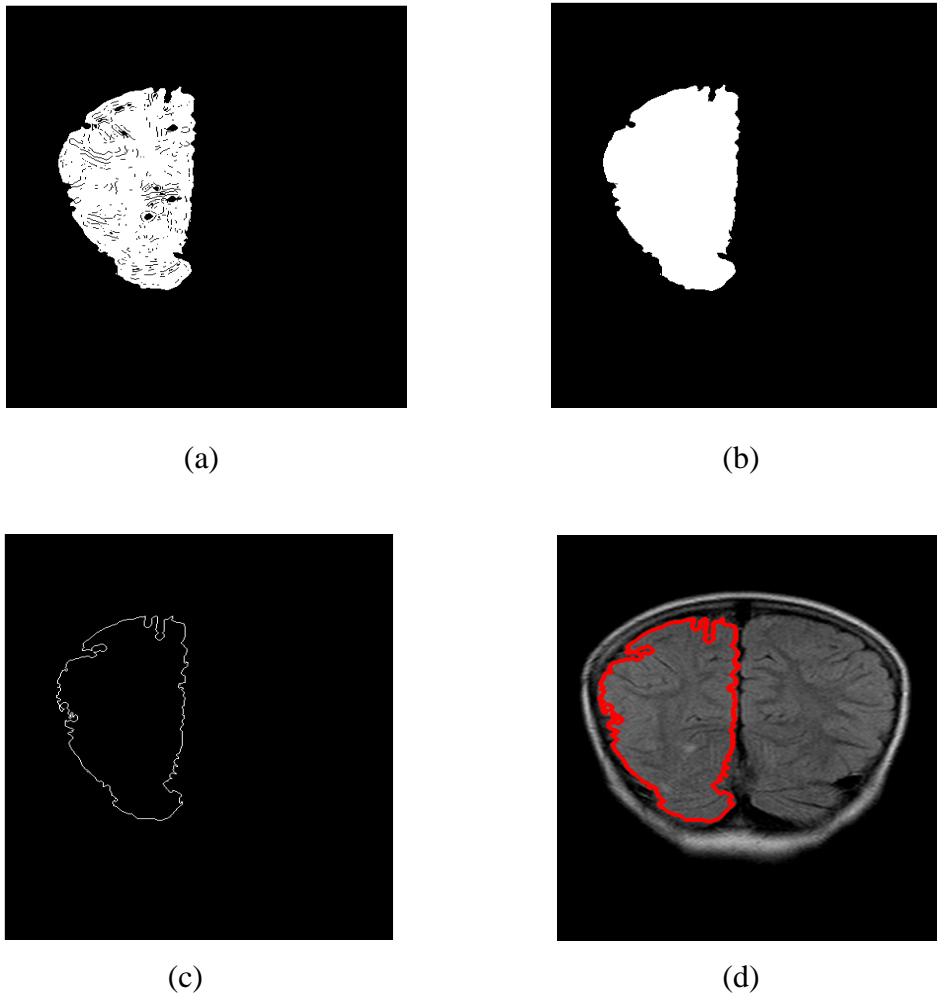


Fig. IV.6 : (a) la région sélectionnée, le remplissage des trous, (c) contour détecté, (d) superposition du contour sur l'image originale

d) *Calcul de la carte contour*

La carte contour, est réalisée en utilisant l'image des contours détecté par le filtre de canny a la place de l'image $I(x, y)$ (Fig. IV.)

Sa formule est la suivante :

$$f(x, y) = |\nabla G_\sigma * I(x, y)| \quad (\text{IV.2})$$

IV.2.3. Choix du snake

Après avoir sélectionné le contour initial, on choisit le type de snake à appliquer (snake classique ou snake GVF), grâce à une commande de matlab qui fait un appel à ce choix de menu.

IV.2.4. Calcul du champ GVF

Le champ GVF minimise la fonctionnelle d'énergie expliquée dans le chapitre précédent (équation), il est obtenu par le calcul variationnel (équation).

Après discrétisation, nous avons :

$$\begin{cases} u(x, y, t + 1) = (1 - b\Delta t)u(x, y, t) + \Delta t\mu\nabla^2 u(x, y, t) + \Delta tc^1(x, y) \\ v(x, y, t + 1) = (1 - b\Delta t)v(x, y, t) + \Delta t\mu\nabla^2 v(x, y, t) + \Delta tc^2(x, y) \end{cases} \quad (\text{IV.2})$$

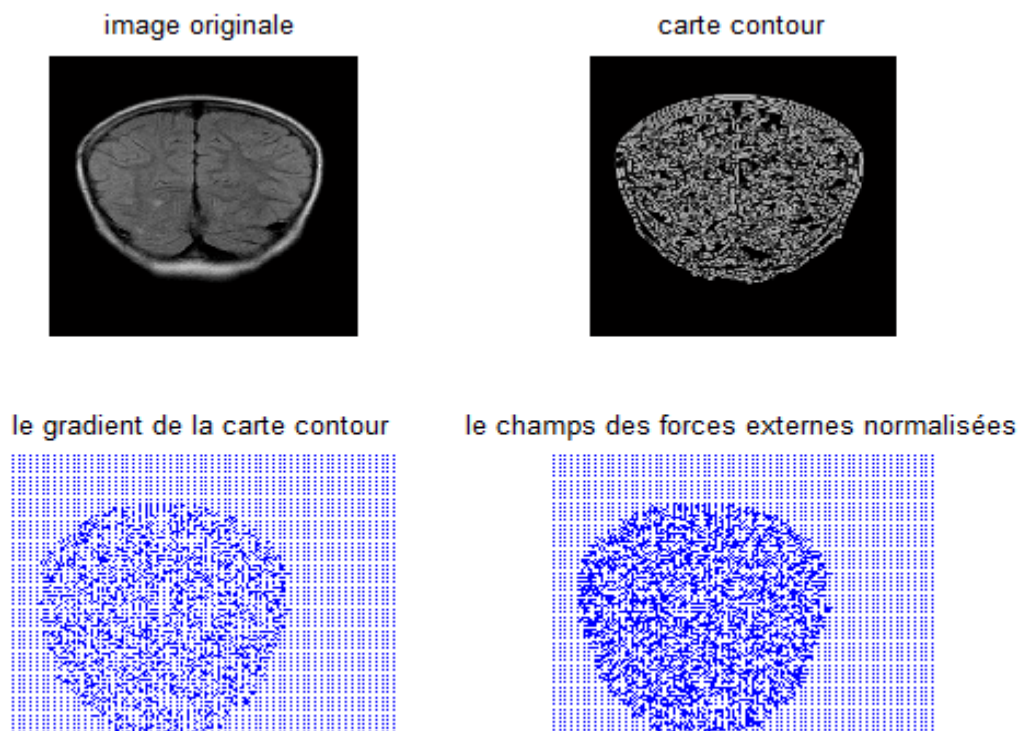


Fig. IV.7 : calcul du champ GVF

IV.2.5. Interpolation linéaire

On cherche désormais à paramétrer la courbe, Pour cela, nous avons choisi de l'interpoler linéairement, en prenant deux points successives qui forment un segment de droite.

Durant la déformation, les deux points $p(i)$ et $p(i + 1)$ peuvent se rapprocher ou s'éloigner. L'espace entre ces deux points varie d'une itération à une autre, ou on peut ajouter des points là où cet espace devient trop important, et seront enlevés quand il est trop faible, c'est ce qu'on appelle reparamétrisation. Pour ce faire on définit deux valeurs de distance (d_{\min} et d_{\max}) et on calcule la distance entre deux points successifs par la formule suivante :

$$d = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (\text{IV.3})$$

Dans notre cas (d_{\min}, d_{\max}) = (0.5, 2).

IV.2.6. Déformation du snake

La déformation du contour initial dépend des paramètres $\alpha, \beta, \gamma, \lambda$ et également des composantes du champ GVF.

La déformation (Fig. IV.9) est définie par un module qui calcule les éléments de la matrice de rigidité A et de son inverse (équation).

Les composantes du contour voisin est calculé par :

$$\begin{cases} X^{i+1} = (A + \gamma I_d)^{-1} (\gamma X^i - \lambda f_x(X^i, Y^i)) \\ Y^{i+1} = (A + \gamma I_d)^{-1} (\gamma Y^i - \lambda f_y(X^i, Y^i)) \end{cases} \quad (\text{IV.4})$$

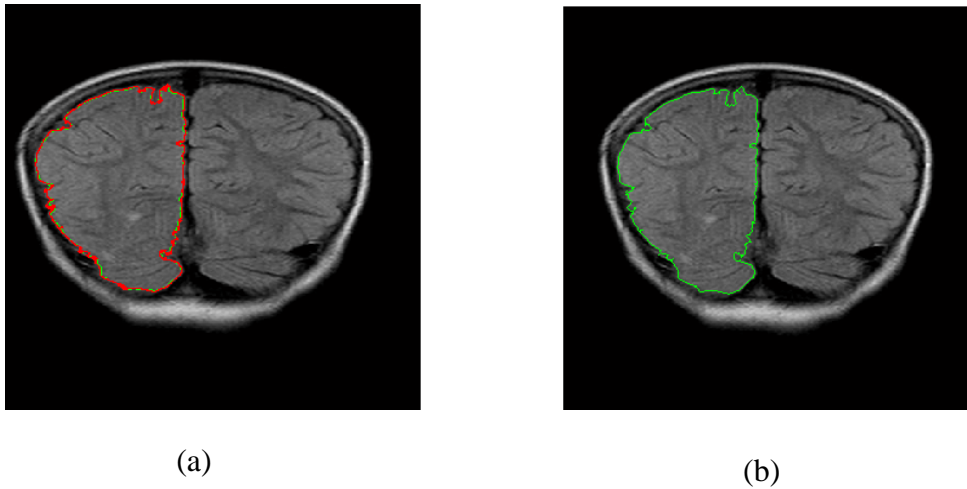


Fig. IV.8 : exemple de déformation d'un *snake* initialisé automatiquement:
(a) contour déformé et (b) contour final.

IV.2.7. Critère d'arrêt

Dans le cas théorique, la déformation de snake s'arrête lorsque le minimum d'énergie est atteint. Mais dans le cas pratique, ceci est difficile à réaliser à cause de la présence des minimums locaux ou les contours parasites, qui empêchent le snake d'évoluer jusqu'à le contour souhaité. C'est pour cette raison que nous avons choisi comme critère d'arrêt : le nombre d'itérations, mais une mauvaise initialisation de ce dernier, peut amener le snake à s'arrêter avant ou après le contour de la région d'intérêt.

IV.3. conclusion

Dans ce chapitre, nous nous sommes intéressés à une approche assez récente en traitement d'image (la méthode GVF). Nous avons utilisé cette dernière pour l'extraction de contour et pour la minimisation de l'énergie de la courbe dans les régions concaves, contrairement au snake classique qui ne peut pas s'attirer vers ces régions. C'est cette méthode qui nous permet de satisfaire notre exigence qui consiste à localiser une tumeur cérébrale.

Nous avons étudié dans ce chapitre l'initialisation automatique qui se fait près du contour à détecter, ensuite on a expliqué l'implémentation de *snake* GVF.

Afin de valider cette méthode nous allons l'appliquer à différentes images IRM dans le chapitre suivant.

Chapitre V
Testes et resultats

V.1.Introduction

Dans ce chapitre nous allons présenter l'évaluation de notre application afin de juger la qualité de l'approche adoptée et la justesse des résultats.

L'objectif de notre travail est de trouver les contours de la tumeur dans une image IRM en utilisant les différentes méthodes implémentées (snake classique et GVF). Cependant, afin de valider notre algorithme et surtout l'efficacité de celui-ci, nous allons, tout d'abord, présenter les résultats obtenues par une initialisation manuelle, ensuite celles obtenues par une initialisation automatique.

V.2. La tumeur

La tumeur cérébrale désigne une prolifération tissulaire, elle a pour principale origine les cellules gliales (les cellules structurelles du cerveau).

La tumeur peut donc à priori apparaître à n'importe quel endroit du cerveau. Elle peut être primaires, situées dans la partie arrière du cerveau, chez les enfants, et dans la partie antérieure des deux tiers des hémisphères_cérébraux, chez les adultes.

Pour détecter une tumeur sur une image IRM on utilise ce qu'on appelle produit de contraste (gadolinium). Ce produit est introduit par une injection intraveineuse, il apparaît comme un signal hyper intense, rendant ainsi la tumeur visible.

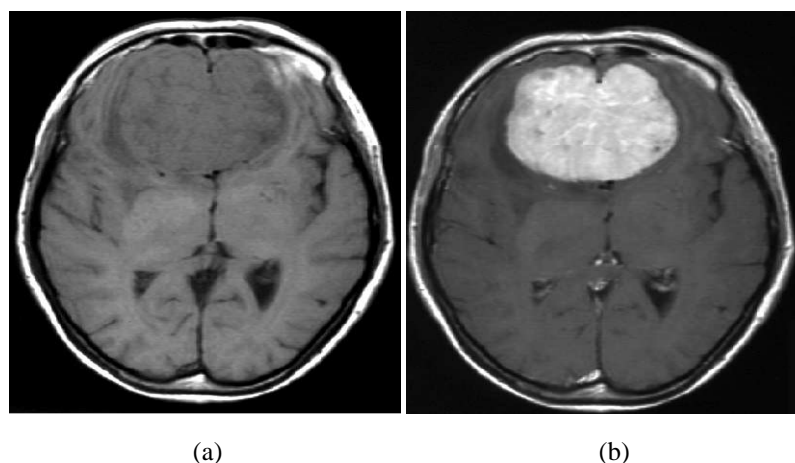


Fig. V .1: Intérêt d'une acquisition avec injection de gadolinium
(a) : image avant injection de gadolinium
(b) : image après injection de gadolinium

Au niveau du cerveau, les parois du système vasculaire cérébral empêchent le produit du contraste de pénétrer à l'intérieur des cellules cérébrales, par contre la tumeur fabrique des vaisseaux dont la paroi est anormale, ce qui permet au produit de contraste de pénétrer à l'intérieur des cellules tumoral.

V. 3. Application de l'approche sur une image IRM

L'objectif est de détecter le contour d'une tumeur dans une image d'IRM cérébrale. A cet effet, nous avons choisis deux images, la première est récupérée du site «[http // www .STI biotechnologie-ST2S/imagerie médicale et endoscopie/programme de terminal ST2S.fr](http://www.STI.biotechnologie-ST2S/imagerie_médicale_et_endoscopie/programme_de_terminal_ST2S.fr)», elle a pour dimension 635x635, format JPG, enregistrée par le docteur Fabrice BING, Centre Hospitalier Universitaire de Grenoble, figure (V.2). La deuxième est récupérée du site «[http //www.IRM.fr](http://www.IRM.fr) », elle a pour dimension 635x635, format JPG, figure(V.3).

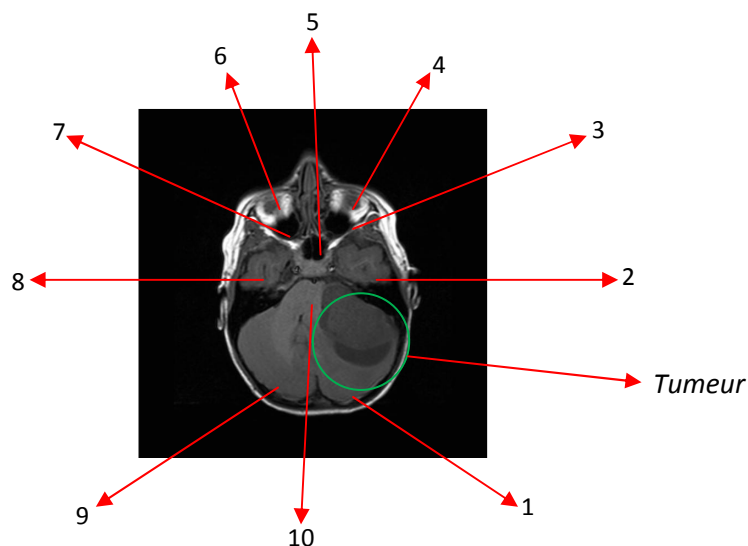


Fig. V.2: image IRM du cerveau de la coupe axiale pondération T₁, avec : 1- lobe occipital droit, 2 - lobe temporal droit, 3 - sinus maxillaire droit, 4 - globe oculaire droit, 5 - sinus sphénoïdal, 6 - globe oculaire gauche, 7 - sinus maxillaire gauche, 8 - lobe temporal gauche, 9 - lobe occipital gauche et 10 - pont.

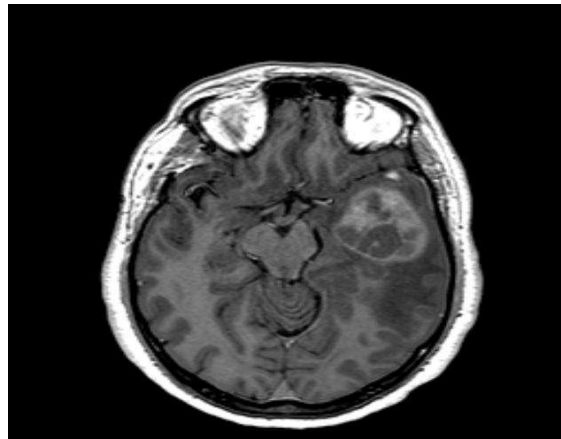


Fig. V. 3 : imageIRM_ponderation_T1_apres_injection_de_gadolinium_Tumeur_temporale_metastase_1

V.3.1 Initialisation manuelle

Les figures V.2 et V.3, montrent le résultat de détection des contours de la tumeur dans l'image IRM, par initialisation manuelle d'un *snake* classique et *snake* GVF, respectivement.

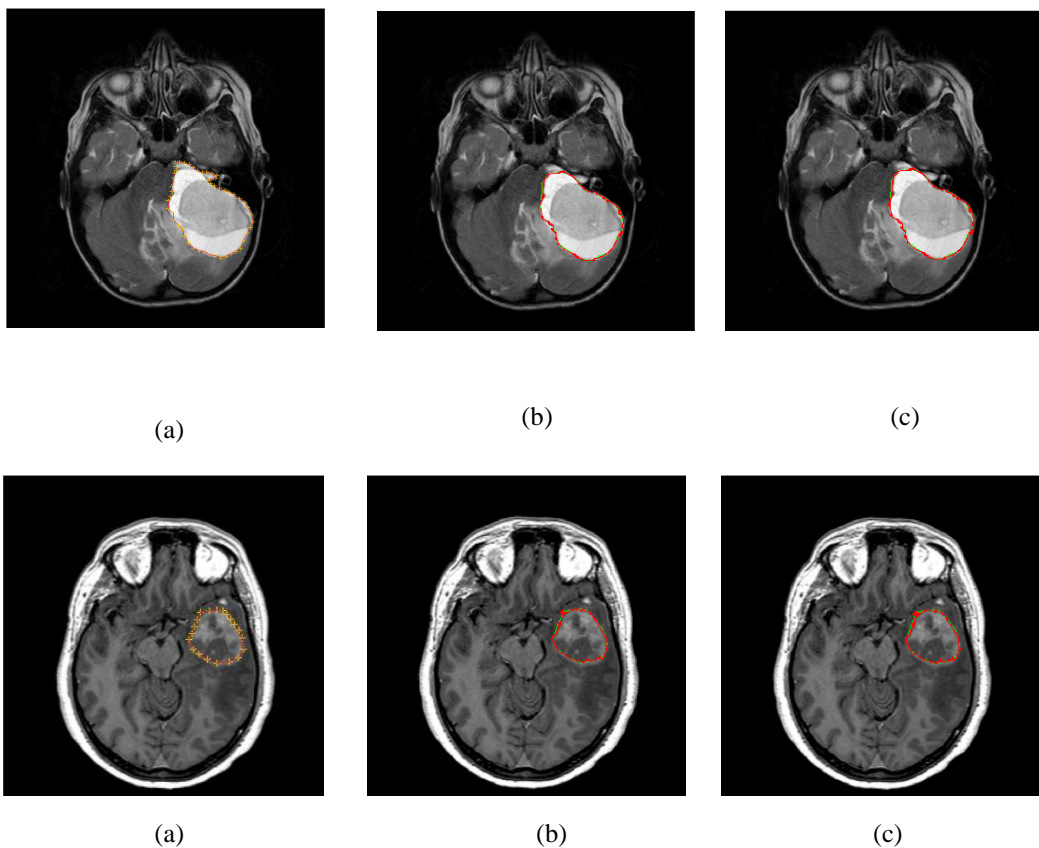


Fig. V. 4 : application du *snake* classique :
(a) image IRM avec initialisation manuelle,
(b) déformations à 10 itérations, (c) résultat après 40 itérations

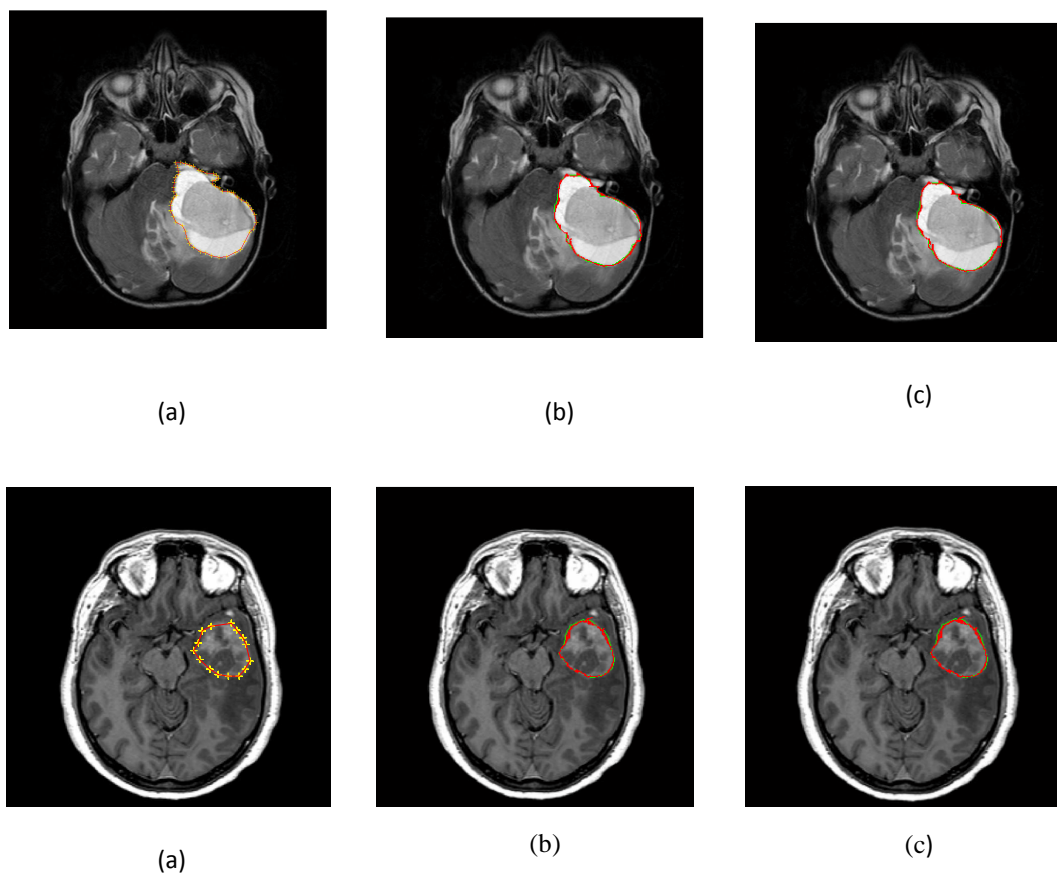


Fig. V. 5 : application du *snake* GVF : (a) image IRM avec initialisation manuelle, (b) déformations à 10 itérations, (c) résultat après 40 itérations

V.3.2 Initialisation automatique

Les figures V.4 et V.5 montrent le résultat de détection des contours de la tumeur dans l'image IRM, par initialisation automatique d'un *snake* classique et *snake* GVF, respectivement.

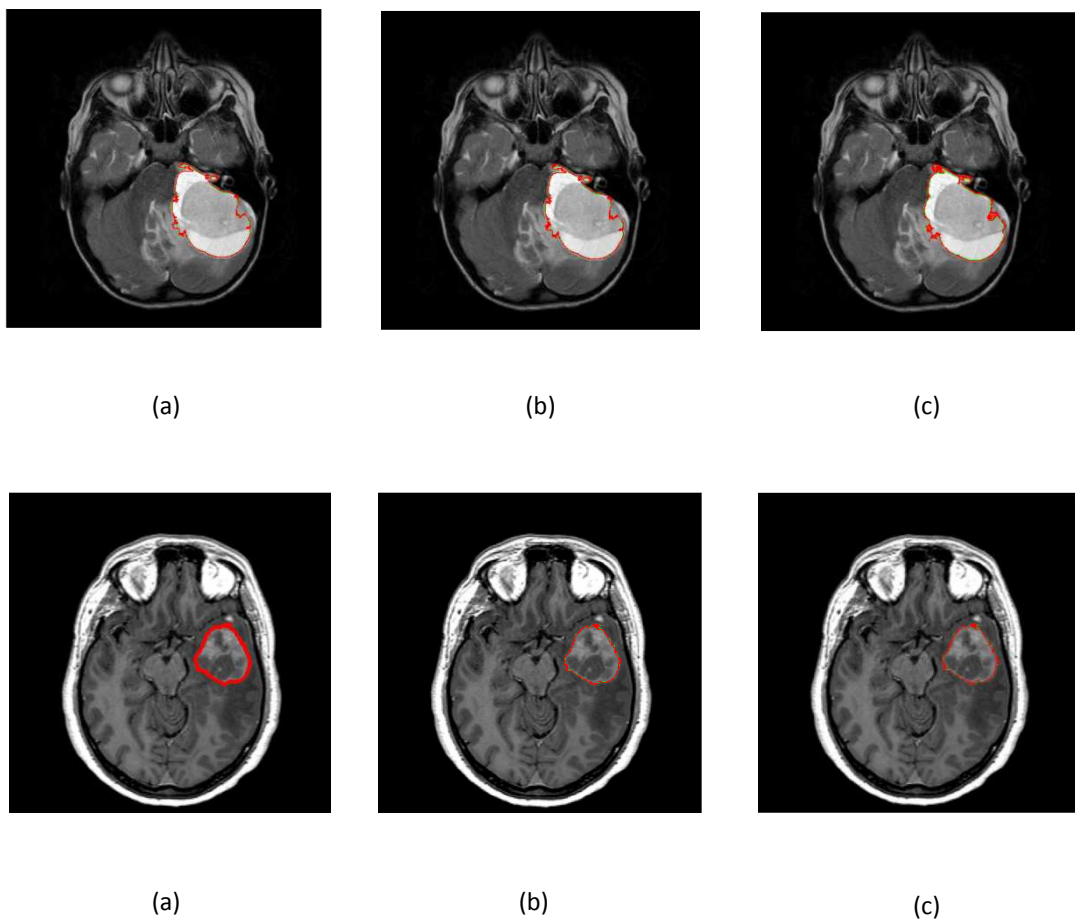


Fig. V.6: application du *snake* classique : (a) image originale avec initialisation automatique, (b) déformation à 10 itérations, (c) résultat après 40 itérations.

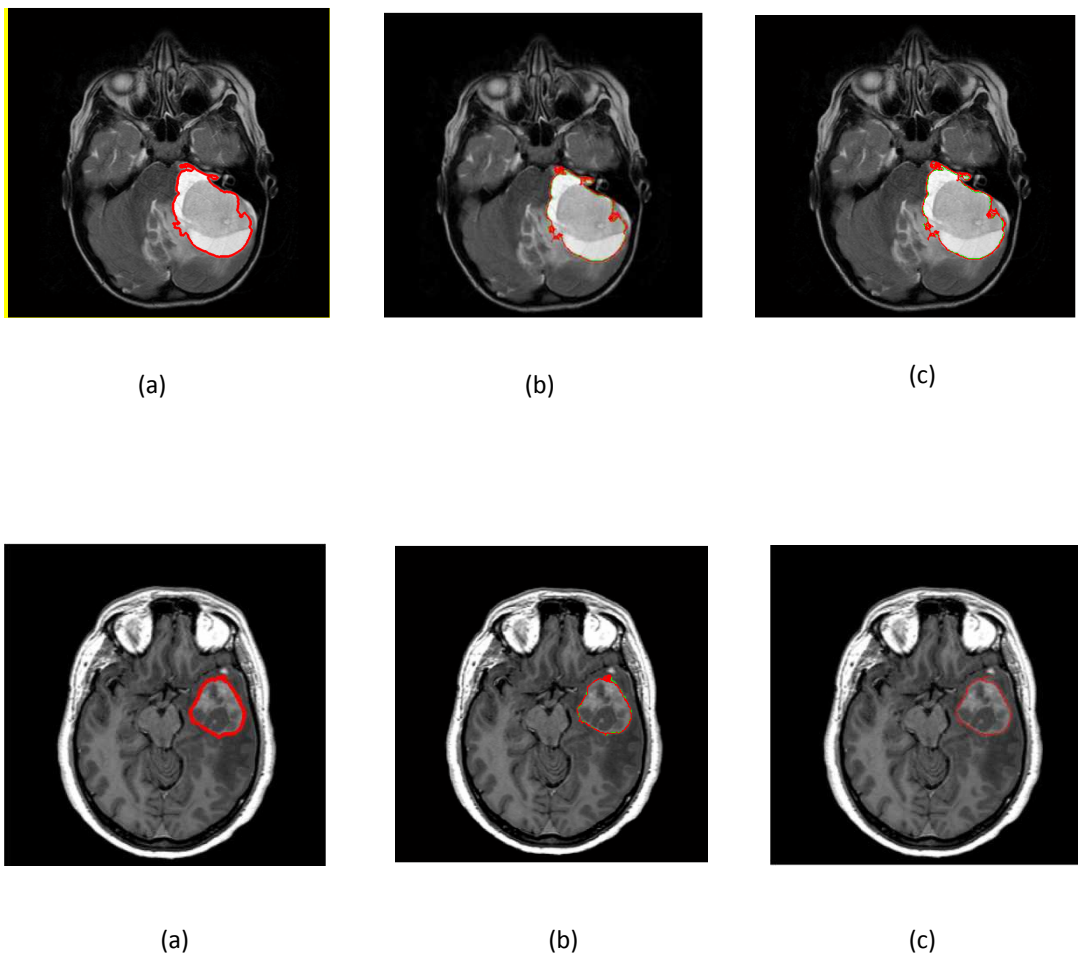


Fig. V.7: application du *snake* GVF : (a) image IRM avec initialisation automatique, (b) déformations à 10 itérations, (c) résultat après 10

V.3. 3. Interprétation des résultats

On remarque que dans l'initialisation manuelle les cavités de la tumeur, sont mal détectées, car si nous observons l'évolution de la courbe dans le temps, nous constatons qu'il commence par se contracter sous l'effet de l'énergie interne qui empêche les irrégularités, pour se fixer, enfin, sur les frontières grâce à l'énergie externe, qui n'est pas assez intéressantes dans ce modèle.

Cependant, pour le *snake* GVF, la cause revient, probablement, à la multitude des contours dans cette région, ce qui crée des forts gradients piégeant.

Pour l'initialisation automatique, les deux snakes détectent bien la tumeur.

Les tableaux V.1 et V.2 donnent les différents temps d'exécution du programme sur les deux images IRM.

<i>snake</i> / initialisation	Initialisation manuelle	Initialisation automatique
<i>snake</i> classique	248.015	119.433
<i>snake</i> GVF	314.357	151.601

Tableau IV.1 : les différentes estimations des temps (en seconde) d'exécutions du programme sur l'image IRM (fig. V.2).

<i>snake</i> / initialisation	Initialisation manuelle	Initialisation automatique
<i>snake</i> classique	94.2023	44.103
<i>snake</i> GVF	156.567	88.5575

Tableau IV.2 : les différentes estimations des temps (en seconde) d'exécutions du programme sur l'image IRM (fig. V.3).

IV. Conclusion

Nous avons vu dans ce chapitre les résultats des tests effectués sur différentes images IRM. Pour obtenir des résultats appréciables, il est utile de bien régler les coefficients de pondérations qui assurent la bonne convergence du snake et de bien initialiser le contour. Pour cela nous avons introduit une méthode d'initialisation automatique, à fin de réduire l'intervention de l'opérateur qui rend l'initialisation manuelle sensible et réduit ses performances.

Nous avons aussi montré l'avantage de snake GVF par rapport au snake classique, qui ne converge pas correctement vers les concavités car la force verticale qui permet de passer le snake vers la celles-ci est nulle, à cause du champ de forces externes qui a été défini précédemment.

Conclusion générale

Conclusion générale

Dans notre travail, en premier lieu nous avons cité les différents outils de traitement d'image, puis nous nous sommes intéressés à l'une des méthodes de segmentation, qui est la méthode des contours actifs (particulièrement le snake classique et le snake GVF). Cette dernière consiste à initialiser une courbe et la faire évoluer de façon à minimiser l'énergie. Ce processus est relatif aux paramètres d'élasticité et de rigidité du contour.

Ensuite Nous avons présenté une technique d'initialisation automatique, qui peut être utilisée avec tout modèle de contour actif paramétrique. Cette méthode fournit un contour initial ayant la forme du contour réel, contrairement à l'initialisation manuelle, qui exige de bien choisir le contour initial.

Les tests, que nous avons effectués sur différentes images (IRM) ont montré l'avantage de l'algorithme de 'l'approche variationnelle' par l'initialisation automatique en terme de convergence de contour initié vers le contour désiré et en terme de vitesse de propagation.

À travers nos tests , nous avons constaté que les résultats obtenus lors de la segmentation par le snake GVF sont plus satisfaisants que ceux obtenus par le snake classique. Mais malgré l'efficacité des contours actifs dans la segmentation d'image, la science est en cour de développement et de recherche pour remédier aux problèmes de paramétrisation (le choix de α , β et γ).

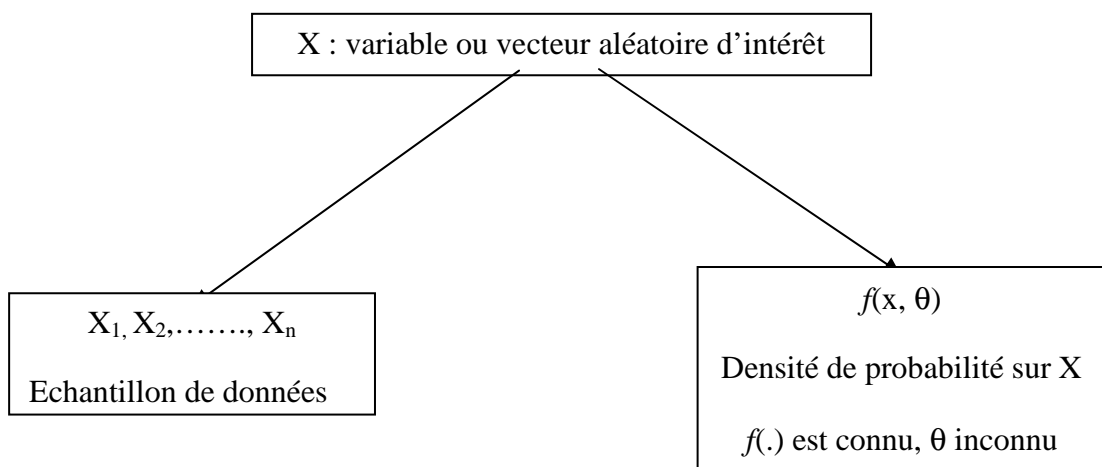
Enfin, nous espérons avoir apporté un travail qui pourra servir à d'autres finalités.

Annexes

Annexe A

Estimateur du maximum de la vraisemblance

A.1 But de l'estimation en statistique



L'estimation a pour but de trouver les valeurs possibles de θ telles que la densité $f(x, \theta)$ s'ajuste le mieux aux données disponibles

Pour effectuer l'estimation, on trouve plusieurs estimateurs : estimateur des moments, estimateur des moindres carrés, estimateur du maximum de vraisemblance.

A.2 Estimateur du maximum de vraisemblance

A.2.1 Fonction de vraisemblance

Soit X une variable aléatoire de densité de probabilité $f(x, \theta)$, où θ est un vecteur de K paramètres : $\theta = (\theta_1, \theta_2, \dots, \theta_k)$.

Soit : X_1, X_2, \dots, X_n un échantillon de données indépendantes.

La fonction de vraisemblance associée est définie par :

$$L(\theta) = \prod_{i=1}^n f(x_i, \theta) \quad (\text{A.1})$$

L'estimateur de maximum de vraisemblance de θ est la valeur de θ qui maximise la fonction de vraisemblance $L(\theta)$.

En pratique on maximise le logarithme de la fonction de vraisemblance :

$$l(\theta) = \ln(L(\theta)) = \sum_{i=1}^n \ln(f(x_i, \theta)) \quad (\text{A.2})$$

A.2.2 Ajustement d'une distribution normale (approche analytique)

La densité d'une distribution normale est définie par :

$$f(x, \theta) = f(x, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2} \quad (\text{A.3})$$

Avec : $\theta(\mu, \sigma^2)$.

La fonction de vraisemblance associée est :

$$L(\theta) = \prod_{i=1}^n f(x_i, \theta) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x_i-\mu)^2/2\sigma^2} \quad (\text{A.4})$$

Le logarithme de vraisemblance est donc :

$$l(\theta) = \sum_{i=1}^n \ln(f(x_i, \theta)) = -\frac{n}{2} \ln(\sigma^2) - \frac{n}{2} \ln(2\pi) - \sum_{i=1}^n \frac{(x_i-\mu)^2}{2\sigma^2} \quad (\text{A.5})$$

La maximisation revient à chercher les valeurs des paramètres qui annulent la dérivée de $l(\theta)$ donc :

$$\frac{\partial}{\partial \mu} l(\theta) = \sum_{i=1}^n 2 \frac{(x_i-\mu)}{2\sigma^2} = 0 \Rightarrow \hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i = \bar{X} \quad (\text{A.6})$$

$$\frac{\partial}{\partial \sigma^2} l(\theta) = -\frac{n}{2\sigma^2} + \sum_{i=1}^n \frac{(x_i-\mu)^2}{2\sigma^4} = 0 \Rightarrow \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2 \quad (\text{A.7})$$

L'estimateur de maximum de vraisemblance obtenue est :

$$\hat{\mu} = \bar{X} \quad ; \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{X})^2 \quad (\text{A.8})$$

Si une approche numérique de dérivation est difficile à résoudre, on utilise les méthodes numériques d'optimisation de fonctions, tel est le cas pour un mélange de gaussiennes.

A.2.3 Ajustement d'un mélange de gaussiennes (approche numérique)

Un mélange de gaussiennes est défini par la somme pondérée de plusieurs gaussiennes :

$$G(x, \theta_j) = \sum_{j=1}^{\alpha} w_j \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{\left(\frac{-1}{2\sigma_j^2}(x^2 - m_j^2)\right)} \quad (\text{A.9})$$

Où w_j représente le poids de chaque gaussiennes et $\theta_j(m_j, \sigma_j^2)$ le vecteur des paramètres de la gaussiennes j .

Alors la fonction de vraisemblance associée est :

$$L(\theta) = \prod_{i=1}^n G(x_i, \theta_j) = \prod_{i=1}^n \sum_{j=1}^{\alpha} w_j \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{\left(\frac{-1}{2\sigma_j^2}(x_i^2 - m_j^2)\right)} \quad (\text{A.10})$$

Le logarithme de la vraisemblance est donc :

$$l(\theta) = \sum_{i=0}^n \ln(G(x_i, \theta_j)) = \sum_{i=1}^n \ln\left(\sum_{j=1}^{\alpha} w_j \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{\left(\frac{-1}{2\sigma_j^2}(x_i^2 - m_j^2)\right)}\right) \quad (\text{A.11})$$

D'où :

$$l(\theta) = \sum_{i=1}^n \prod_j^{\alpha} \left(\ln w_j - \frac{1}{2} \ln(2\pi) - \frac{1}{2} \ln(\sigma_j^2) - \frac{1}{2\sigma_j^2} (x_i^2 - m_j^2) \right) \quad (\text{A.12})$$

La maximisation, donc, revient à résoudre l'équation suivante :

$$\frac{\partial}{\partial \theta} l(\theta) = 0 \quad (\text{A.13})$$

Une telle équation est difficile à résoudre analytiquement, alors on fait recourir aux méthodes numériques d'optimisation des fonctions

Dans le cas de mélange de gaussiennes, les paramètres sont estimés en utilisant l'algorithme EM (espérance-maximisation).

Cet algorithme estime les paramètres itérativement et chaque itération se divise en deux étapes :

➤ **Etape E** (Espérance)

Cette étape consiste à calculer l'espérance de la vraisemblance complétée conditionnellement aux données observées au vecteur paramètres courant. Alors, cette étape nécessite simplement le calcul de la probabilité *a posteriori* :

$$p^k(j, x_i) = \frac{\omega_j^k p(x_i, \theta_j^k)}{\sum_{j=1}^3 \omega_j^k p(x_i, \theta_j^k)} \quad (\text{A.14})$$

$$\text{Avec : } \left\{ \begin{array}{l} \mathcal{N}(m_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{\left(\frac{-1}{2\sigma_i^2}(x^2 - m_i^2)\right)} \\ P(x, \theta_i) = \mathcal{N}(m_i, \sigma_i^2) \end{array} \right. \quad (\text{A.15})$$

➤ **Etape M** (Maximisation)

Dans cette étape on calcule le vecteur des paramètres qui maximise l'espérance. Cette maximisation fournit les paramètres :

$$\omega_i^{k+1} = \frac{1}{n} p^k(i, x_j) \quad (\text{A.16})$$

$$m_i^{k+1} = \frac{\sum_{j=1}^n x_j p^k(i, x_j)}{\sum_{j=1}^n p^k(i, x_j)} \quad (\text{A.17})$$

$$(\sigma_i^{k+1})^2 = \frac{\sum_{j=1}^n (x_j - m_i^{k+1})^2 p^k(i, x_j)}{\sum_{j=1}^n p^k(i, x_j)} \quad (\text{A.18})$$

On définit le vecteur des paramètres $\varphi^{k+1} = \left(\omega_i^{k+1}, m_i^{k+1}, (\sigma_i^{k+1})^2\right)$.

L'algorithme s'arrête lorsque :

$$\|\varphi^{k+1} - \varphi^k\| \leq \varepsilon \quad (\text{A.19})$$

Annexe B

Les différences finies

Introduction

Selon les différences finies les éléments de la courbe sont réduits en des points auxquels sont attachés les éléments mécaniques (masse, raideur...) de la courbe considérée en ces points, cette méthode consiste à remplacer les dérivées apparaissant dans le problème continu par des différences divisées. On note f l'image considérée, et $f(x, y)$ sa valeur au pixel (x, y) . Le pas d'espace h est fixé égal à 1.

- **Différences finies centrées**

$$\delta_x f_{x,y} = \frac{f_{x+1,y} - f_{x-1,y}}{2}$$

$$\delta_y f_{x,y} = \frac{f_{x,y+1} - f_{x,y-1}}{2}$$

- **Différences finies décentrées à droite (avant) :**

$$\delta_x^+ f_{x,y} = f_{x+1,y} - f_{x,y}$$

$$\delta_y^+ f_{x,y} = f_{x,y+1} - f_{x,y}$$

- **Différences finies décentrées à gauche (arrière)**

$$\delta_x^- f_{x,y} = f_{x,y} - f_{x+1,y}$$

$$\delta_y^- f_{x,y} = f_{x,y} - f_{x,y+1}$$

I. Description de la méthode des différences finies

Les méthodes aux différences finies découlent de la série de Taylor suivante :

$$f(x+h) = f(x) + \frac{df(x)}{dx}h + \frac{df^2(x)}{dx^2} \frac{(h)^2}{2!} + \frac{df^3(x)}{dx^3} \frac{(h)^3}{3!} + \dots \quad (\text{B.1})$$

Si nous éliminons les termes d'ordre supérieur ou égal à 2, nous obtenons

$$f(x+h) \approx f(x) + \frac{df(x)}{dx}h$$

En isolant le terme dérivé nous obtenons une Différence avant d'ordre 1

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h} \quad (\text{B.2})$$

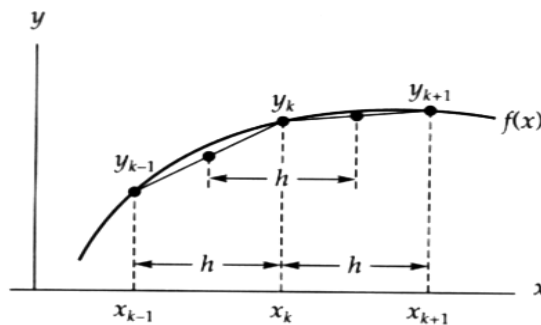


Fig. .B.1. Illustration des méthodes aux différences (dérivée première)

L'approximation d'ordre 2 de la dérivée première de $f(x)$ est obtenue en incluant un second terme à la série (B.1):

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h} - \frac{df^2(x)}{dx^2} \frac{h}{2!}$$

Nous devons estimer d'abord la dérivée seconde en utilisant une méthode aux différences finies de la forme:

$$\frac{df^2(x)}{dx^2} \approx \frac{f'(x+h) - f'(x)}{h}$$

Si nous substituons le résultat de l'estimation de la dérivée première par différence finie d'ordre 1 nous obtenons Approximation de Premier ordre de la Dérivée seconde

$$\frac{df^2(x)}{dx^2} \approx \frac{f(x+2h) - 2f(x+h) + f(x)}{(h)^2} \quad (\text{B.3})$$

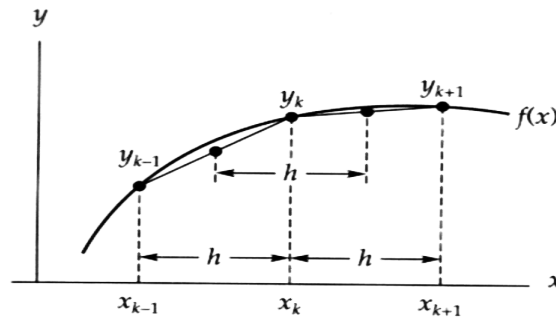


Fig. B.2. Illustration des méthodes aux différences (dérivée seconde)

En utilisant une méthode aux différences finies arrière nous obtenons les approximations d'ordre 1 pour la dérivée première et seconde

$$\frac{df(x)}{dx} \approx \frac{f(x) - f(x-h)}{h} \quad (\text{B.4})$$

$$\frac{df^2(x)}{dx^2} \approx \frac{f(x) - 2f(x-h) + f(x-2h)}{(h)^2} \quad (\text{B.5})$$

L'approximation d'ordre 1 pour les dérivée trois et quatre sont obtenue en incluant un troisième et quatrième terme respectivement à la série de (I)

$$\frac{df^3(x)}{dx^3} \approx \frac{f'(x+2h) - 2f'(x+h) + f'(x)}{(h)^2} = \frac{f(x+3h) - 3f(x+2h) + 3f(x+h) - f(x)}{h^3} \quad (\text{B.6})$$

$$\frac{df^4(x)}{dx^4} = \frac{f'(x+3h) - 3f'(x+2h) + 3f'(x+h) - f'(x)}{h^4} = \frac{f(x+4h) - 4f(x+3h) + 6f(x+2h) - 4f(x+h) + f(x)}{h^4} \quad (\text{B.7})$$

En utilisant une méthode aux différences finies centrées, nous obtenons ces approximations

$$f''(x) = \frac{-2f(x) + f(x+h) + f(x-h)}{h^2}$$

$$f'''(x) = \frac{f(x+2h) - 4f(x+h) + 6f(x) - 4f(x-h) + f(x-2h)}{h^4}$$

Bibliographie

Bibliographie

[1]: **GONZALES, R.C, WINTZ** « Digital Image Processing » ADDISON-WESLEY, 1977

[2]: **AIT MAMMAR HACENE ET MAKOUR ABDERREZAK** « segmentation morphologique d'image par la ligne de partage des eaux », mémoire d'ingénieur, département d'électronique, 2008.

[3]: **AZERRADJ DAI ET KESSOUM DJAMEL** « Application des Contours Actifs Géodésiques sur des séquences d'images », mémoire d'ingénieur, Institut National de formation en Informatique (I.N.I), Oued-Smar, Alger, 2007.

[4]: **KEDDOUR Badr ET HAMTOUCHE ALI** « Contours actifs paramétriques pour la segmentation d'images et vidéos » mémoire d'ingénieur, Institut National de formation en Informatique (I.N.I), Oued-Smar, Alger, 2007.

[5]: **AIT HAMOU YOUNGOURTHENE ET HAMROUN FERHAT** «segmentation d'image par contours actifs géodésiques », mémoire d'ingénieur, Institut National de formation en Informatique (I.N.I), Oued-Smar, Alger, 2008.

[6] **M: AKLI KAMAL ET M: HAMID SAMIR** « initialisation automatique des snakes pour une segmentation d'une image IRM », mémoire d'ingénieur, département d'automatique, UMMTO, 2007.

[7]: **CHILALI. O,** « Classification automatique de données utilisant les modèles déformables », mémoire de magister, Département d'Automatique, UMMTO, 2006.

[8]: **JEAN JACQUES ROUSSELLE** « les contours actifs, une méthode de segmentation, application à l'imagerie médicale », mémoire de doctorat, université François Rabelais de Tour, France, 2003. Téléchargé du site, « [http // www-Isis.enst.fr/rilk/gdr/kiosque/thèse HDR](http://www-Isis.enst.fr/rilk/gdr/kiosque/thèse_HDR) », dernière visite_05/09/2009

[9] : **DOCUMENT INTERNET**

[10]: **RACHID BENMOKHTAR** « segmentation par contours actifs de type GVF, application à la segmentation des artères carotides en IRM », mémoire de projet de mastère, université de Technologie de Compiègne, 2005

[11]: **XU C** « Deformable Models With Application To Human Cerebral Cortex Reconstruction From Magnetic Resonance Images », Thèse de doctorat, Université Johns Hopkins, Baltimore, Maryland, USA, janvier 1999.

[12]: **J-P COCQUEREZ ET S.PHILIPP** « Analyse d'image, filtrage et segmentation », édition Masson Paris, 1995.

[13]: **LOUNACI D ET LAZIB R** « traitement des images Meteosat de format penMTP' », Mémoire d'ingénieur, Département Electronique, UMMTO, 2004.

[14]: **BUREL G** « Introduction au traitement d'images », Simulation sous Matlab.

[15]: **DOCUMENT INTERNET** « [www.prostatecancer.ca /french/prostate_owners_manual/Living/glossary/](http://www.prostatecancer.ca/french/prostate_owners_manual/Living/glossary/) », dernière visite. 20/08/2009

[16]: **DOCUMENT INTERNET** « www.info-cancer.fr », dernière visite 10/09/2009.

[17]: **DOCUMENT INTERNET** « www.atlas-IRM/IRM_axial_du_cerveau.com », dernière visite 20/08/2009.