

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU



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

Mémoire de fin d'études

En vue de l'obtention

Du Diplôme de Master II en Electronique

Option : Réseaux et télécommunication

Thème :

**Segmentation d'images en niveaux de gris par
modélisation basée sur les niveaux de gris
ordonnés**

Proposé et dirigé par :

Mme AMEUR Zohra

Présenté par :

M^r TAMDRARI Youcef

M^{elle} BAKIRI Dihia

Année universitaire 2010/2011

Dédicace et remerciement

***A tous ceux qui ont contribué,
de près ou de loin, à
l'élaboration de
ce travail***

SOMMAIRE

Introduction générale.....	1
Chapitre I : Généralités sur le traitement d'image.	
Préambule	2
I-1 Définition de l'Image	2
I-2 L'image numérique	2
I-2-1 Les images binaires (noir ou blanc).....	3
I-2-2 Les images en niveaux de gris.....	3
I-2-3 Les images couleurs.....	3
I-3 Caractéristiques d'une image.....	4
I-3-1 Pixel	4
I-3-2 Dimension	5
I-3-3 Résolution.....	5
I-3-4 Luminance	5
I-3-5 Contraste	6
I-4 Images bitmap et images vectorielles	6
I-5 Notions de voisinage et connexité	7
I-6 Histogramme d'une image	8
I-7 Modification d'histogramme	9
I-8 Contours et texture.....	10
I-9 Bruit et filtrage.....	11
I-9-1 Définition du bruit	11
I-9-2 Filtrage linéaire.....	11
➤ Filtre passe-bas	12
➤ Filtre passe-haut.....	13
➤ Filtre passe-bande	14
I-9-3 Filtrage non linéaire	14
➤ Filtre d'ordre.....	14

➤ Filtrage par la médiane	14
➤ Filtrage par le plus proche voisin radiométrique	14
➤ Filtre de position	15
I-10 Opérations morphologiques.....	15
Discussion	16

Chapitre II : Notions de segmentation, contours et textures.

Préambule	17
II-1 Définition de la segmentation.....	17
II.2 Méthodes de segmentation	17
II-2-1 Segmentation basée sur les régions	17
II-2-2 Segmentation basé sur les contours.....	18
II-2-2-1 Opérateurs Classiques de détection de contours.....	18
➤ Le gradient.....	18
➤ Filtre optimaux	20
II-2-2-2 Propriétés d'un détecteur de contour	22
II-2-2-3 Quelques méthodes existantes en localisation de contour.....	22
➤ La morphologie.....	22
➤ La méthode de passage par zéro	23
➤ Extraction des extrema locaux du gradient	23
II-3 Analyse d'images texturées.....	25
II-3-1 Définition d'une signature texturale	25
II-3-2 Qualité d'une signature texturale.....	26
II-3-3 Extraction d'attributs de texture.....	26
II-4 Quelque technique de segmentation d'images	27
II-4-1 La matrice de cooccurrence	27
II-4-2 Matrice des longueurs de plage.....	28
II-4-3 Différence des niveaux de gris	28

II-4-4 Modélisation basée sur les niveaux de gris ordonnés	29
Discussion	30

Chapitre III : segmentation par modélisation basée sur les niveaux de gris ordonnés

Préambule	31
III- 1 Définition de niveaux de gris ordonnées (NGO)	31
III-2 Modélisation de région par régression sur les NGO	32
III-2-1 Un modèle linéaire	32
A- Caractéristiques de a	34
B- Caractéristiques de b	37
III-2-2 Détection de contours par la modélisation s'appuyant sur NGO	39
III-2-2-1 Détection des transitions de niveaux de gris en utilisant le premier paramètre dans le modèle.....	39
III-2-2-2 Détection des transitions de niveaux de gris en utilisant la multiplication du premier paramètre et la variance du deuxième	39
A- Méthode de classification.....	40
B- Méthode de rangs.....	41
III-3 Mise en œuvre de la détection de contour	42
III-3-1 Détection de contour en utilisant le paramètre a de la fonction linéaire	42
A- Algorithme C_a.....	42
B- Algorithme R_a.....	43
III-3-2 Détection de contours utilisant le produit du paramètre et la variance du deuxième paramètre	44
A- Algorithme $R_a * \text{Var}(b)$	44
B- Algorithme $C_a * \text{Var}(b)$	45
Discussion	46

Chapitre IV : Tests et résultats.

IV-Présentation des données	47
IV-1 Les images utilisées.....	47
IV-1-1 Images de l'album de BRODATZ	47
IV-1-2 Les images médicales utilisées	47
IV-1-3 matériel et logiciel utilisés.....	48
IV-2 Résultats obtenus sur les images tests.....	48
IV-2-a En utilisant une fenêtre 3x3	48
IV-2-1-b En utilisant une fenêtre 5x5.....	50
IV-2-2 Valeurs du facteur P.....	52
IV-3 Interpretation des resultats.....	52
IV-4 Discussion.....	54
Conclusion	55

INTRODUCTION

GÉNÉRALE

INTRODUCTION GENERALE :

L'analyse d'images s'inscrit dans une problématique plus générale appelée la vision par ordinateur. La vision est la science définissant les bases théoriques et algorithmiques permettant d'extraire, d'analyser automatiquement des informations sur notre environnement à partir d'une image, un ensemble d'images ou une séquence d'images, sur une machine spécialisée ou non. Cette définition laisse imaginer combien ce domaine de recherche est vaste. Il est donc impératif de réfléchir à la manière d'aborder ce problème.

La définition d'une méthode générale pour l'analyse d'images est actuellement très difficile vue la diversité des problèmes abordés. De même, il est délicat d'élaborer un système complet de vision par ordinateur. Nous allons alors nous intéresser plus particulièrement à la première phase, c'est à dire la segmentation. Le but d'un processus de segmentation est l'extraction des attributs caractérisant les entités représentées dans l'image, ou encore de décrire l'image à l'aide d'indices visuels simples.

Dans notre travail, on s'intéresse à une technique de segmentation d'images en niveaux de gris par modélisation basée sur les niveaux de gris ordonnés. C'est une modélisation des régions à l'aide d'une fonction linéaire obtenue par approximation des niveaux de gris dans une fenêtre carrée. Les niveaux de gris ordonnés ont été initialement utilisés pour le filtrage non linéaire d'image (filtre médian, filtre à moyenne tronquée). L'avantage de cette méthode est qu'elle enlève efficacement les bruits impulsionnels tout en conservant la structure c'est-à-dire le contour de l'image. Cette méthode ordonne d'abord les niveaux de gris, puis réalise un filtrage en utilisant un filtre non linéaire basé sur les rangs de niveaux de gris et détecte ensuite les contours en utilisant la différence entre le niveau de gris de rang maximum et celui de rang minimum dans une fenêtre donnée.

Pour se faire, le premier chapitre est consacré aux notions de base utilisées en traitement d'image. Les principales méthodes d'analyses et de segmentation d'images feront l'objet du second chapitre. Le troisième chapitre décrira la méthode adoptée pour la segmentation d'images et nous donnerons l'algorithme retenu pour cette approche. Le dernier chapitre illustrera les différents tests effectués, puis une interprétation sur les différents résultats expérimentaux obtenus par la méthode implémentée sera faite. Enfin, nous terminerons notre travail par une conclusion et des perspectives ouvertes par ce travail.

CHAPITRE I

GÉNÉRALITÉS SUR LE TRAITEMENT D'IMAGES

Préambule :

L'image est l'un des moyens les plus efficace pour communiquer, car chacun peut faire sa propre analyse, afin d'en dégager une impression et d'extraire des informations bien précises.

Le traitement d'image est un ensemble de techniques opérant sur cette dernière. Il a pour but l'exploitation maximale des informations contenues dans celle-ci.

On distingue quatre types de traitement formant ainsi quatre disciplines:

- Rehaussement et restauration d'image.
- Codage et compression d'image.
- Synthèse d'image.
- Analyse d'image.

I-1 Définition de l'Image :

L'image n'est qu'une représentation spatiale d'un objet, d'une scène ou d'une autre image par différentes formes comme la peinture, la photographie, etc. Elle peut être réelle ou virtuelle, c'est à dire issue d'un procédé de synthèse.

En analyse d'images, on travaille généralement sur des images digitales obtenues à partir du monde réel. On peut alors la représenter par une fonction de brillance analogique continue $I(x, y)$, définie dans un domaine borné, où x et y sont les coordonnées d'un point de l'image et $I(x, y)$ représente l'information observée qu'elle soit une fonction d'intensité lumineuse (niveaux de gris) ou bien couleur. Sous cet aspect, l'image est inexploitable par la machine, il faut alors la numériser.

I-2 L'image numérique :

Une image numérique est un signal numérique composé d'unités élémentaires (appelées pixels). Chaque unités représente un point de l'image.

Une image numérique est définie par :

- Le nombre de pixels qui la compose en largeur et en hauteur.
- L'étendue des teintes de gris ou des couleurs que peut prendre chaque pixel.

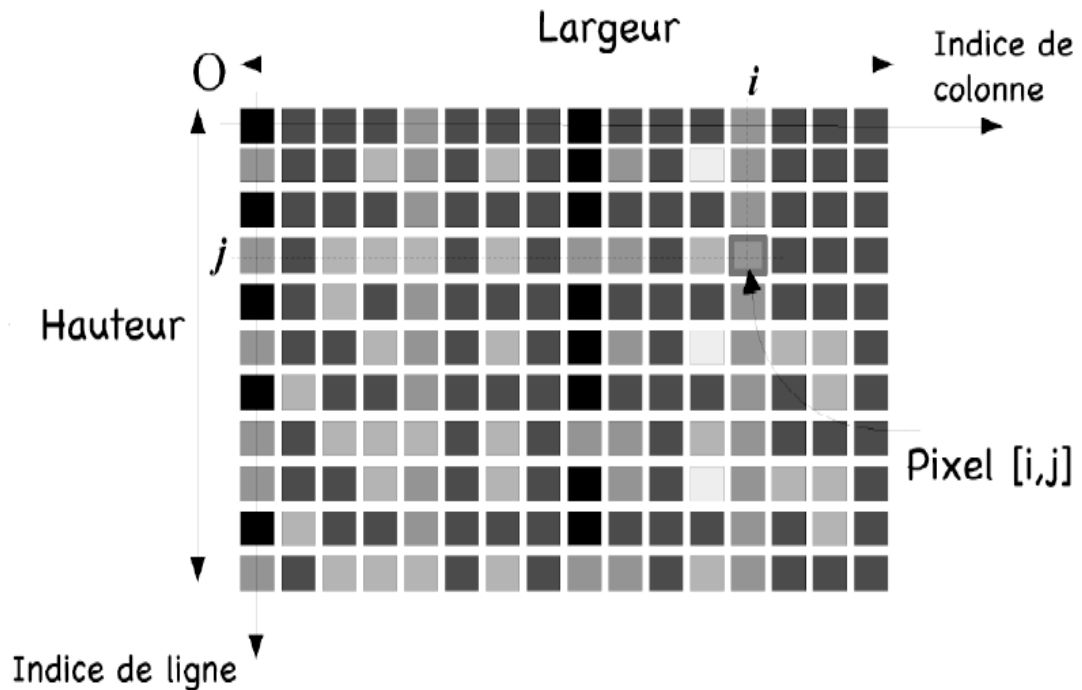


Figure.1 : pixels et niveaux de gris

I-2-1 Les images binaires (noir ou blanc) :

Une image binaire est une image dont les pixels ne peuvent prendre que deux valeurs, le noir absolu ou le blanc absolu. C'est le type d'image que l'on obtient en scannant des textes.

I-2-2 Les images en niveaux de gris :

En général, les images en niveaux de gris renferment 256 teintes de gris. Par convention la valeur zéro représente le noir absolu et la valeur 255 le blanc absolu. En effet chaque entier représentant un niveau de gris codé sur 8 bits. Ce type d'image peut aussi être codé sur 16 bits ou 2 bits : dans ce dernier cas le « niveau de gris » vaut 0 ou 1 : il s'agit alors d'une image binaire (Noir et Blanc).

I-2-3 Les images couleurs :

L'espace couleur est basé sur la synthèse additive des couleurs, c'est à dire que le mélange de trois composantes (par exemple (R, V, B) rouge vert bleu, (C, M, J) cyan Magenta jaune) donne une couleur. On garde l'information couleur, ou intensité lumineuse. Un pixel est codé par trois valeurs numériques donc sur 24 bits.

L'espace couleur le plus utilisé pour le maniement des images numériques est Rouge, Vert, Bleu (R, V, B) (RGB en anglais).

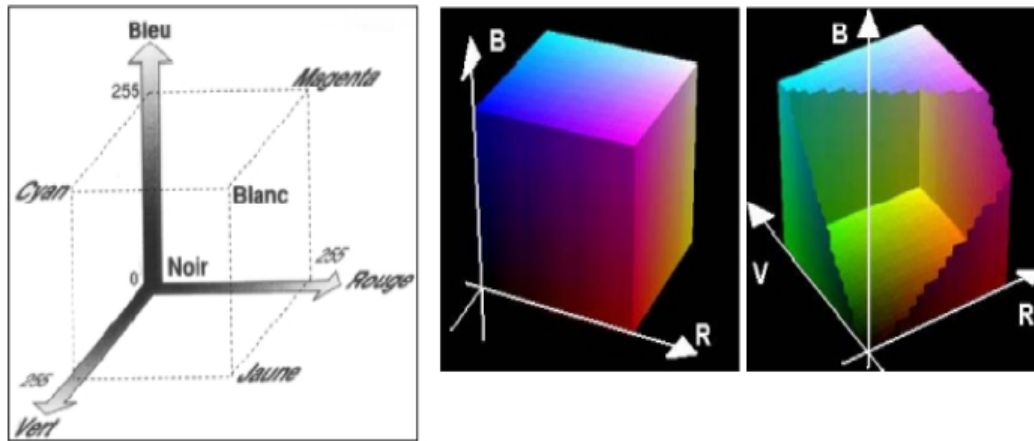


Figure.2 : L'espace couleur Rouge Vert Bleu (RGB)

I-3 Caractéristiques d'une image :

L'image est un ensemble structuré d'informations caractérisées par plusieurs paramètres que nous décrivons ci-après.

I-3-1 Pixel :

Pour un ordinateur, une image est un ensemble de pixels (picture element). Un pixel est le plus petit élément de l'image.

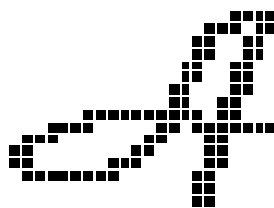


Figure.3 : Image présentant l'affichage des pixels

Un pixel possède une valeur qui peut être un scalaire et représenter un niveau de gris, ou un vecteur représentant une couleur.

Les images dites en (noire et blanc) sont composées de pixels noire ou blanc. Les images en niveaux de gris sont composées de pixels de valeurs scalaires représentant la luminosité. En général, les valeurs sont entières et codées sur 8bits.

I-3-2 Dimension :

Comme la représentation de l'image est une matrice dont les éléments sont des valeurs numériques, donc la multiplication du nombre de ligne par le nombre de colonne nous donnera la taille de l'image [4].

I-3-3 Résolution :

La résolution d'une image est définie par un nombre de pixels par unité de longueur de la structure à numériser (classiquement en ppp). Ce paramètre est défini et dépend principalement des caractéristiques du matériel utilisé lors de la numérisation. Plus le nombre de pixels par unité de longueur de la structure à numériser est élevé, plus la quantité d'information qui décrit cette structure est importante et donc une résolution plus élevée, c'est-à-dire un degré de détail de l'image assez grand.

I-3-4 Luminance :

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de celle-ci. Pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet.

La moyenne ou luminance (brillance) d'une image numérique en niveau de gris est définie comme la moyenne des pixels de l'image [4]:

$$\text{Lum}(I) = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x, y) \quad (\text{I.1})$$

Avec :

M : Nombre de colonnes,

N: Nombre de lignes,

I(x, y) : La valeur de niveau de gris dans le point (x, y).

I-3-5 Contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de celle-ci. Le contraste est défini en fonction des luminances de deux zones d'image.

Si L_1 et L_2 sont les degrés de luminosité respectivement de deux zones voisines A_1 et A_2 d'une image, le contraste C est défini par le rapport :

$$C = \frac{L_1 - L_2}{L_1 + L_2} \quad (I.2)$$

I-4 Images bitmap et images vectorielles :

On distingue généralement deux grandes catégories d'images :

- **Les images bitmap** : Il s'agit d'images pixellisées, c'est-à-dire un ensemble de points (pixels) contenus dans un tableau, chacun de ces points possédant une ou plusieurs valeurs décrivant sa couleur.
- **Les images vectorielles**: Les images vectorielles sont des représentations d'entités géométriques telles qu'un cercle, un rectangle ou un segment. Ceux-ci sont représentés par des formules mathématiques (un segment est défini par deux points, un cercle par un centre et un rayon, une courbe par plusieurs points et une équation). C'est le processeur qui sera chargé de "traduire" ces formes en informations interprétables par la carte graphique.

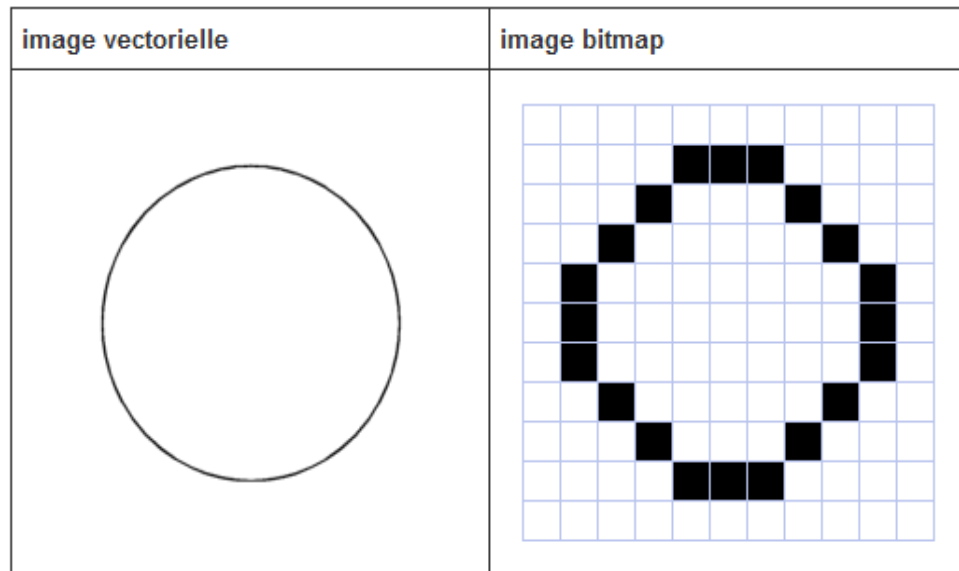


Figure.4: image vectorielle et bitmap

I-5 Notions de voisinage et connexité :

Le traitement d'image s'appuie fondamentalement sur des traitements à l'aide de voisinages. Cela signifie que les traitements effectués sur un pixel en un endroit donné, dépendent aussi de son voisinage.

Considérons un pixel P dont la position dans l'image est donnée par les coordonnées (m, n) . Un voisinage de P , noté $V(P)$, se définit comme un ensemble de pixels P' connectés à P . Ce dernier, encerclé dans la figure.5, appartient à son propre voisinage $V(P)$.

Donc, à ce niveau, il est nécessaire de définir la notion de « composantes connexes » au sein d'une image discrète, afin de savoir comment ses pixels sont connectés entre eux.

Plutôt que de s'étendre sur la notion de connexité, nous présentons ci-dessous des figures illustrant les deux cas les plus couramment utilisés :

- **un voisinage de « 4-connexité »** : le pixel (centré et entouré dans la figure 5 à droite) n'a que quatre pixels voisins, chacun d'eux étant à distance unité (d_4) de P .

- **un voisinage de « 8-connexité »** : le pixel (au centre et entouré dans la figure 5 à gauche) a huit pixels voisins, chacun d'eux étant également à une distance unité (d_8) de P .

On définit les deux distances suivantes en voisinage numérique à structure carrée par:

$$d4(P, P') = |m - m'| + |n - n'|. \quad (I.3)$$

$$d8(P, P') = \text{Sup} (|m - m'|, |n - n'|). \quad (I.4)$$

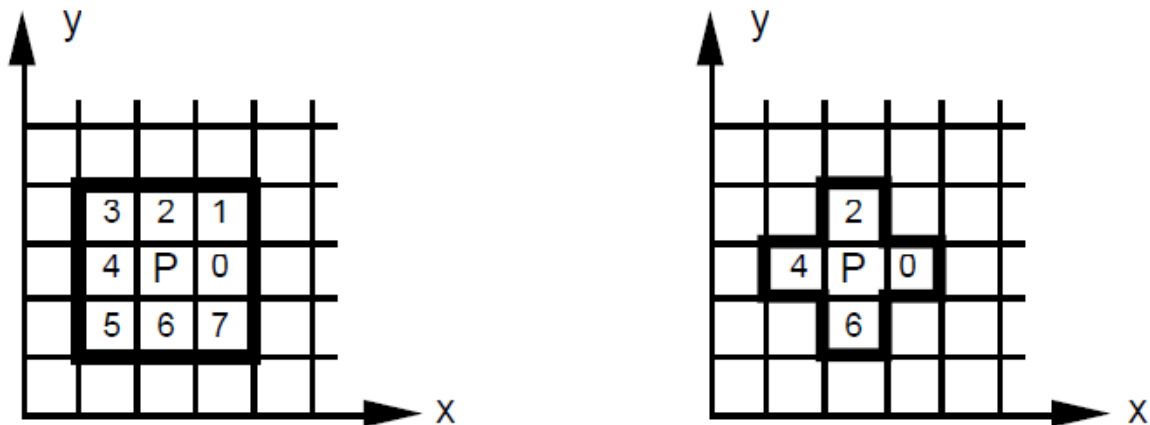


Figure.5: Différents voisinages d'un point

I-6 Histogramme d'une image :

L'histogramme des niveaux de gris est un outil qui représente un résumé simple du contenu d'une image. Il donne la fréquence d'apparition de chaque niveau de gris et permet alors de donner leur distribution. On appelle cette distribution « dynamique de l'image ».

On distingue trois types d'histogrammes :

1. **Histogramme unimodal** : formé d'un seul pic représentant soit l'objet, soit le fond.
2. **Histogramme bimodal** : formé de deux pics séparés mettant en évidence l'existence d'un objet et d'un fond.
3. **Histogramme multimodale** : formé de plusieurs modes séparés (pics séparés par plusieurs vallées) mettant en évidence l'existence de plusieurs objets.

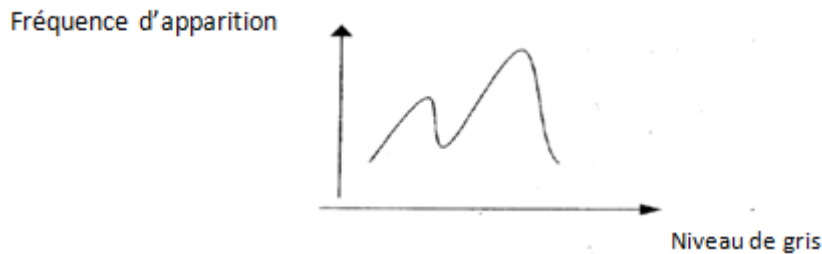


Figure.6: Exemple d'histogramme bimodal

I-7 Modification d'histogramme :

La modification d'histogramme a pour objectif l'amélioration de l'image en lui appliquant plusieurs transformations. Parmi ces transformations, on a :

A- Egalisation d'histogramme :

L'égalisation d'histogramme est une transformation des niveaux de gris dont le principe est d'équilibrer le mieux possible la distribution des pixels dans la dynamique (Idéalement, on cherche à obtenir un histogramme plat).

Pour une image $\{x\}$ en niveaux de gris codée sur L niveaux, on définit n_k le nombre d'occurrences du niveau x_k . La probabilité d'occurrence d'un pixel de niveau x_k dans l'image est:

$$p_x(x_k) = p(x = x_k) = \frac{n_k}{n}, \quad 0 \leq k < L$$

Avec n le nombre total de pixels de l'image, et p_x l'histogramme normalisé sur $[0,1]$.

La transformation T qui à chaque pixel de valeur x_k de l'image d'origine associe une nouvelle valeur s_k , $s_k = T(x_k)$ est alors définie par :

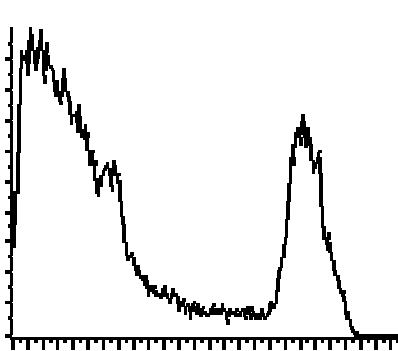
$$T(x_k) = \frac{(L-1)}{n} \sum_{j=0}^k n_j$$



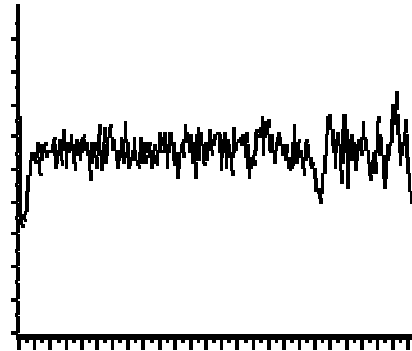
Avant égalisation d'histogramme



Après égalisation d'histogramme



Histogramme avant égalisation



Histogramme après égalisation

Figure.7 : Traitement par égalisation d'histogramme

B- Etirement d'histogramme :

C'est une opération qui sert à modifier l'histogramme de telle manière à répartir aux mieux les intensités sur l'échelle des valeurs disponibles. Ceci revient à étendre l'histogramme afin que la valeur d'intensité la plus faible soit à zéro et que la plus haute soit à la valeur maximale. que l'on calcule comme suit :

$$\text{Nouvelle valeur} = 255 * \left(\frac{\text{valeur} - \text{min}}{\text{max} - \text{min}} \right)$$

Avec min = la valeur du niveau de gris minimale dans l'image.

max = la valeur du niveau de gris maximal dans l'image

De cette façon, si la valeur de l'histogramme sont très proches les une des autres, l'étirement va permettre de fournir une meilleure répartition afin de rendre les pixels claires encore plus claires et les pixels foncés proche du noir.

C- Seuillage :

Le seuillage consiste à mettre à zéro “0” tous pixels ayant un niveau de gris inférieur à une certaine valeur appelée seuil et à un “1”, les pixels ayant une valeur supérieure à ce seuil [13]. Ainsi le résultat de seuillage est une image binaire contenant les pixels noirs et blancs, c’est la raison pour laquelle le terme de binarisation est employé. Les seuillages permettent de mettre en évidence des formes ou des objets dans une image. On distingue deux types de seuillage :

- **Seuillage global** : le principe du seuil global est de déterminer à quelle classe appartient le pixel (objet ou fond).
- **Seuillage local** : le principe est d’utiliser une étude localisée au tour du pixel pour déterminer quel seuil utiliser. Cette étude se fait à l’aide d’une fenêtre centrée en ce pixel. Elle peut avoir différentes tailles et elle est choisie souvent en fonction de la taille de l’image.

I-8 Contours et texture:

La texture est une propriété naturelle de chaque surface telle que la toile d’un tissu, les motifs formés par les récoltes d’un champ..., c’est une traduction de la structure existante dans une région. On peut la percevoir dans presque tous les types d’images. Les effets visuels de la texture peuvent être schématisés par la notion de finesse, de grossièreté, de granularité ou de directionnalité des motifs. Le problème inhérent à toute la recherche dans ce domaine est de donner une définition unique et précise de la texture. Vaguement, on peut la considérer comme la distribution spatiale plus ou moins périodique des niveaux de gris ou couleurs [8].

Les contours représentent la frontière entre les objets de l’image, qui se traduit par une brusque variation du niveau de gris. Un contour peut être défini comme une marche d’escalier s’il est net ou comme une rompe s’il est flou.

I-9 Bruit et filtrage :

I-9-1 Définition du bruit :

Le bruit est un terme issu du domaine de l’acoustique et désigne un signal parasite. Que ce soit pour le son ou pour l’image, le principe est identique : sur tout signal de base vient s’adjoindre un ensemble d’informations parasites aléatoires.

Si le niveau du signal est suffisant, la proportion de bruit dans le signal utile (le fameux rapport signal/bruit) reste insignifiante. Par contre, si le niveau de bruit prend le pied sur l'information principale, le bruit sera pesant.

Afin d'améliorer la qualité visuelle des images, on doit effectuer un traitement appelé « filtrage numérique ». Ce dernier consiste à modifier la distribution fréquentielle du signal selon des spécifications données.

I-9-2 Filtrage linéaire:

Le filtrage est la convolution d'une image $I(x, y)$ avec une fonction $F(x, y)$ qui s'appelle réponse impulsionnelle du filtre. Dans le cas discret, les domaines de I et de F sont bornés. Le domaine de I est de $[-n/2, +n/2]$, et celui de F est de $[-k/2, +k/2]$.

Avec

n^2 : la taille de l'image.

k^2 : la taille de la fenêtre de traitement.

➤ Filtre passe-bas :

Ces filtres ont généralement pour but de réduire les parasites (bruits de mesure). Ils agissent par moyenne sur un voisinage et atténue donc les détails, par conséquent, ils provoquent un adoucissement de contour.

Par exemple, le masque suivant utilise la moyenne non pondérée de ses 8 points voisins:

$$H = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

L'image filtrée $g(x,y)$ est définie par l'expression suivante:

$$G(x,y) = \frac{1}{9} \sum_{i=-1}^{+1} \sum_{j=-1}^{+1} (f(x+i, y+j))$$



Image Lena originale

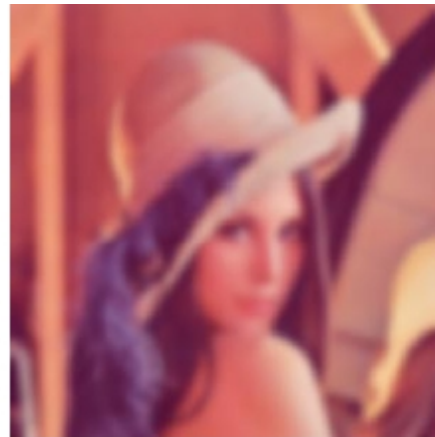


Image après filtrage passe-bas

Figure.8 : Exemple de filtrage passe-bas

➤ **Filtre passe-haut :**

Ces filtres ont pour but l'augmentation du contraste et de mettre en évidence les contours en atténuant la composante basse fréquence.

Les masques ci-dessous mettent en évidence les gradients positifs suivant l'axe des y et l'axe des x.

$$P_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$P_y = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$



Image Lena originale



Image Lena après filtrage Passe-haut

Figure.9 : Exemple de filtrage passe haut

➤ **Filtre passe-bande :**

Cette opération est une dérivée du filtre passe-bas. Elle élimine la redondance d'information entre l'image originale et l'image obtenue par filtrage passe-bas, seule la différence entre l'image source et l'image traitée est conservée.

I-9-3 Filtrage non linéaire :

On désigne par filtre non linéaire, un opérateur qui remplace la valeur de chaque pixel par une combinaison non linéaire des valeurs de ses pixels voisins.

Le but principal du filtrage non linéaire est d'apporter des solutions aux limites des filtres linéaires en utilisant plusieurs techniques comme les statistiques d'ordre (basée sur le classement des valeurs de pixels voisins) etc. On distingue plusieurs types de filtres non linéaires :

➤ **Filtre d'ordre :**

Les filtres d'ordres procèdent en remplaçant les valeurs de chaque pixel par la valeur qui occupe un certain rang lorsqu'on trie les valeurs observées dans un certain voisinage du pixel. Il existe deux types de filtre d'ordre :

- **Filtrage par la médiane :** la valeur du point central d'une fenêtre de l'image est affectée par celle de ces voisins en classant les valeurs de cette dernière par ordre croissant (en prenant la valeur médiane).



Image initiale



Image après filtrage médian

Figure.10 : Exemple de filtrage médian

- **Filtrage par les plus proches voisins radiométriques :** Dans ce type de filtrage, on attribut au point central la moyenne des K pixels voisins, dont les valeurs radiométriques sont plus proches de la sienne.

➤ **Filtre de position :**

Ce filtre utilise des voisinages 5x5, le sous voisinage le plus homogène est sélectionné et son niveau de gris moyen est affecté au point central.

I-10 Opérations morphologiques :

➤ **Dilatation :**

Elimine les pixels noirs isolés sur un fond blanc. On promène une fenêtre 3x3 pixels et pour le pixel P, on effectue le OU logique de ses huit voisins.

Si $OU(P_i) = 1$ alors $P = 1$ Sinon P_i reste inchangé.

Où 1 = blanc 0 = noir.

➤ **Erosion :**

Procédé dual de la dilatation, l'érosion permet d'éliminer les pixels blancs isolés sur un fond noir en effectuant le ET logique.

Si $ET(P_i) = 1$ alors P reste inchangé Sinon $P = 0$.

➤ **Ouverture :**

Cette opération adoucit les pics trop acérés (zone claire), elle est réalisée par une érosion suivie d'une dilatation.

➤ **Fermeture :**

L'opération de fermeture comble les vallées trop étroites (zones sombres) et permet de remplir les trous et elle est réalisée par une dilatation suivie d'une érosion.

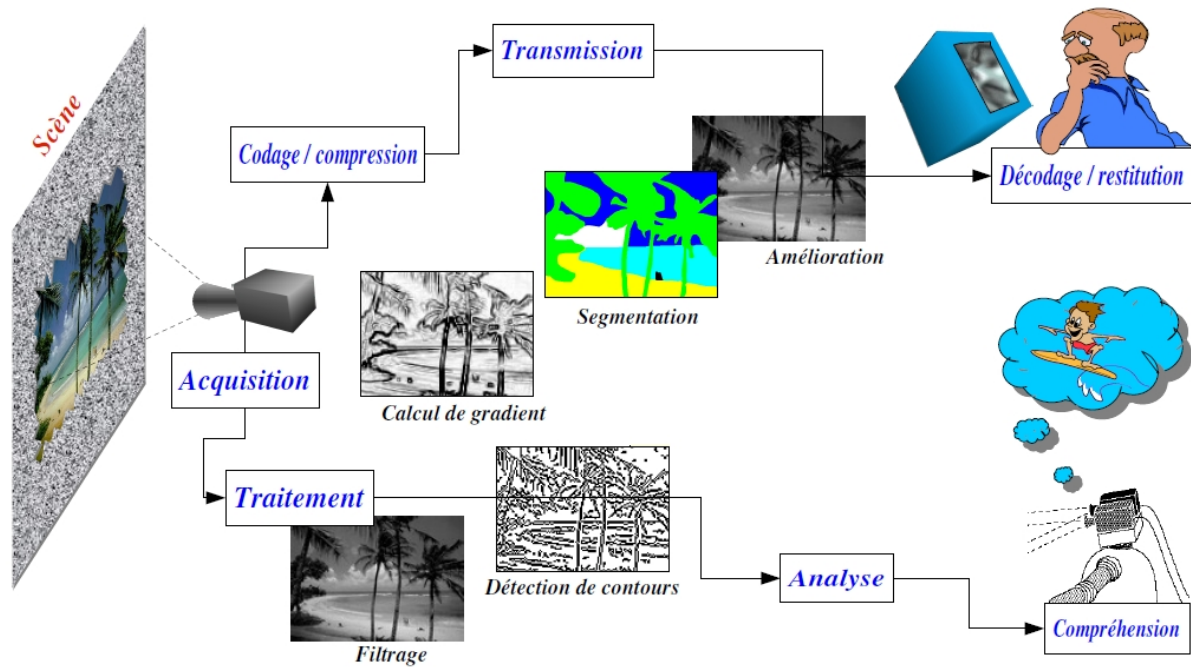


Figure.11 : Exemple d'un système basé sur le traitement d'image

Discussion :

La représentation des images fixes est un des éléments essentiels des applications multimédias, comme dans la plupart des systèmes de communication.

La manipulation des images pose des problèmes beaucoup plus complexes. En effet, l'image est un objet à deux dimensions, censée représenter un espace à trois dimensions, ce qui a deux conséquences majeures :

- Le volume des données à traiter est beaucoup plus important.
- La structure de ces données est nettement plus complexe.

Il en résulte que la manipulation, le stockage et la représentation de ces données se heurtent à certaines limitations et grâce au traitement d'image, ces contraintes sont levées ou contournées.

Le prochain chapitre abordera les différentes techniques de segmentation d'images.

CHAPITRE II

*NOTIONS DE
SEGMENTATION,
CONTOURS
ET
TEXTURES*

Préambule

Dans le traitement d'image, la segmentation est une étape fondamentale dans la mesure où elle conditionne l'interprétation de l'image. En effet, elle a pour objectif l'extraction des éléments de l'image, et permet également la description de l'information qu'elle contient en donnant des informations facilement exploitable.

II-1 Définition de la segmentation :

Elle consiste en l'affectation des pixels d'une image à des sous ensembles homogènes et disjoints, formant ainsi une partition de l'image. Les pixels qui appartiennent à une même région partagent une propriété commune, dite critère d'homogénéité de région.

II.2 Méthodes de segmentation :

Dans l'objectif d'analyser une image, la segmentation est une étape primordiale. On regroupe de façon usuelle les méthodes de segmentation en deux groupes basés respectivement sur les régions et les contours.

II-2-1 Segmentation basée sur les régions :

Cette catégorie correspond aux algorithmes d'accroissement ou de découpage de région.

- L'accroissement de région est une méthode qui consiste à démarrer d'un ensemble de petites régions uniformes dans l'image (de la taille d'un ou de quelques pixels), et on regroupe les régions adjacentes de même couleur jusqu'à ce qu'aucun regroupement ne soit plus possible.
- Le découpage de région est une méthode qui consiste à choisir initialement l'image entière que l'on va subdiviser récursivement en plus petites régions tant que ces régions ne seront pas suffisamment homogènes.

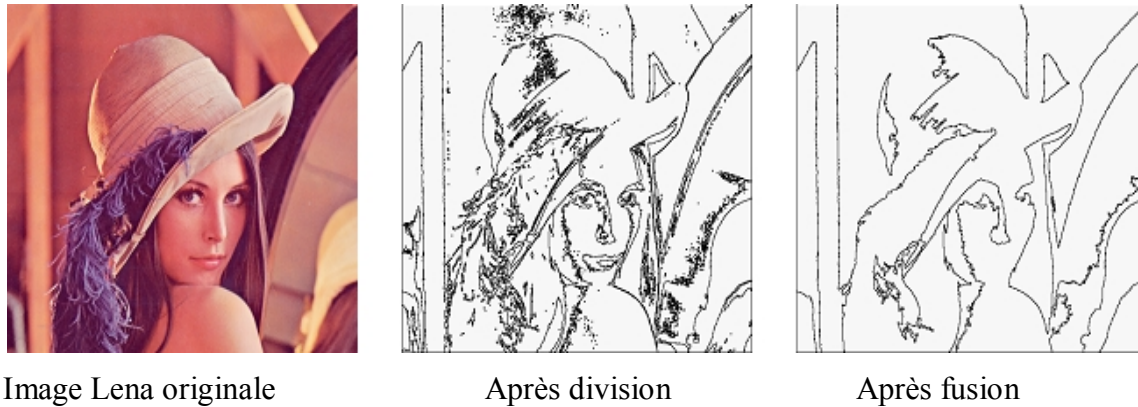


Figure.12 Exemple de segmentation d'image par fusion et division

II-2-2 Segmentation basé sur les contours :

Cette méthode s'intéresse à la détection des contours des objets dans l'image, elle permet une segmentation par une génération de frontière entre les objets. On retrouve des points de contours qui définissent généralement une discontinuité dans l'homogénéité des régions. La plupart de ces algorithmes sont locaux, c'est à dire fonctionnent au niveau du pixel. Des filtres détecteurs de contours sont appliqués à l'image, le plus souvent de type gradient tel que le filtre de Sobel et celui de Canny-Deriche [17].

Une fois la norme et la direction du gradient calculées en chaque pixel de l'image, ces méthodes extraient des contours d'un seul pixel d'épaisseur en sélectionnant les maxima locaux des normes des gradients.

II-2-2-1 Opérateurs Classiques de détection de contours :

A- Le gradient :

L'opérateur gradient permet de détecter les variations brusques des niveaux de gris. Le passage par un maximum du module du gradient détermine alors un point de contour [16]. Soit $I(x, y)$ la fonction de répartition de la luminance dans l'image. Le gradient en coordonnées cartésiennes est exprimé par :

$$, = \frac{(,)}{1} + \frac{(,)}{1} \quad (\text{II-1})$$

- **Opérateur de Prewitt :**

Prewitt a choisi le masque de taille 3x3 suivant:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Ce filtre peut être séparé en deux filtres monodimensionnels :

Le premier $\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$ est un moyenneur suivant l'axe des X dont la dérivée est $\begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$,
le second $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ est un moyenneur suivant l'axe des Y dont la dérivée est $\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$.

A partir des définitions des filtres séparables, nous pouvons déterminer les masques de convolution du filtre de **Prewitt Px** et **Py** pour chaque direction [2] :

$$P_x = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ -1 & 1 & 0 & -1 \end{bmatrix}$$

$$P_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ -1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

- **Opérateur de Sobel :**

Sobel a choisi un masque de type:

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Ce filtre peut être séparé en deux filtres monodimensionnels.

Le premier $\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$ est un moyenneur suivant l'axe des X dont la dérivée est $\begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$.

Le second $\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & -1 \end{bmatrix}$ est un moyenneur suivant l'axe des Y dont la dérivée est $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 1 & -1 \end{bmatrix}$.

A partir des définitions des filtres séparables, nous pouvons déterminer les masques de convolution du filtre de Sobel S_x et S_y pour chaque direction [2] :

$$S_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & -1 \\ 2 & 0 & -2 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

$$S_y = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 & 1 \\ -1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Remarque:

Nous avons présenté quelques uns des masques les plus utilisés. Cependant, ils ne sont qu'un cas particulier. Pour savoir quelle taille choisir pour un opérateur, nous donnons ici quelques règles simples :

1. Plus le masque est grand, moins le gradient est sensible au bruit.
2. Plus le masque est grand, plus le temps de calcul est élevé.
3. Plus le masque est grand, moins bonnes est la localisation des contours.

Ces indications sont contradictoires et il faut donc réaliser un compromis. Généralement, il existe des contours de natures très différentes qui nécessitent l'utilisation de plusieurs tailles de masques.

B- Filtre optimaux :

Ces détecteurs sont dits optimaux car s'appuyant sur le modèle de contour de type marche, ils cherchent à obtenir les meilleurs résultats possibles : partant du principe que l'étape de lissage a entraîné une perte d'information, il faut donc essayer de minimiser cette perte. L'idée est de déterminer des critères de performances et essayer de définir le détecteur optimisant ces critères. Cette approche et cette modélisation sont dues à J. Canny.

➤ **Canny:**

Pour J. Canny, les critères à optimiser sont l'exhaustivité de la détection et l'exactitude de la localisation. Il a utilisé la dérivée première d'une gaussienne qui est une bonne approximation de son filtre [15]. L'implantation 2D est réalisée à l'aide d'un masque de convolution.

➤ **Shen-Castan :**

Pour J. Shen et S. Castan minimiser ces pertes revient à minimiser l'énergie du bruit et maximiser l'énergie du signal [15]. De plus, comme on va procéder à une étape de différentiation, il faut minimiser le bruit dans la dérivée première du filtre. La différence de leur modèle par rapport à celui de J. canny, c'est qu'ils effectuent une combinaison différente de ses critères. Ils obtiennent le filtre :

$$S(x) = c \cdot \exp\left(-\frac{|x|}{d}\right) \quad \text{avec } c = \frac{1}{2d} \quad (\text{II-2})$$

Où d est l'échelle du filtre et c est une constante de normalisation calculée pour avoir un maximum de la réponse (soit une réponse égale à 1) en $x = 0$.

Et pour le filtre dérivateur :

$$S'(x) = \begin{cases} -\frac{1}{d} \exp\left(-\frac{|x|}{d}\right) & x \geq 0 \\ \frac{1}{d} \exp\left(-\frac{|x|}{d}\right) & x \leq 0 \end{cases} \quad (\text{II-3})$$

Avec $d = 1$

➤ **Filtre de Deriche :**

Au filtre de Canny, on préfère souvent le détecteur de Deriche, qui répond exactement aux mêmes critères de qualité que celui de Canny, mais qui possède une réponse impulsionnelle infinie (filtre RII). Il a pu donc être synthétisé de façon récursive particulièrement efficace. Le filtre de Deriche a une expression générale de la forme :

$$S(x) = \frac{1}{2} \exp\left(-\frac{|x|}{d}\right) \quad (\text{II-4})$$

Avec

$$= \frac{[1 - \exp(-\alpha)]}{\exp(-\alpha)}$$

II-2-2-2 Propriétés d'un détecteur de contour :

On peut définir trois propriétés fondamentales permettant de déterminer les caractéristiques d'un détecteur [7]:

➤ Une bonne détection :

L'opérateur doit avoir une forte probabilité de détecter le point P quelque soit le bruit dans l'image.

➤ Une bonne localisation :

L'opérateur doit permettre de localiser avec précision la position du point de contour. Pour cela, il faut minimiser les variations Δx et ΔY représentant l'erreur de position.

➤ L'unicité de la réponse :

L'opérateur doit détecter un et un seul point de contour. Il faut donc éliminer la possibilité d'avoir une réponse multiple pour un point de contour.

II-2-2-3 Quelques méthodes existantes en localisation de contour :

Pour pallier aux deux inconvénients existant en localisation des contours qui sont [5]:

- 1- les contours trop épais entre les régions lorsque les transitions de niveaux de gris entre régions sont très fortes.
- 2- Les contours perdus dans des zones où les transitions des niveaux de gris entre régions sont faibles.

De nombreuses méthodes ont été proposées :

A- La morphologie :

On procède d'abord au seuillage des amplitudes de transitions de niveaux de gris avec un seuil assez bas. Cela permet de ne pas perdre de contours dans l'image où les transitions de niveaux de gris sont faibles, mais cela produit des contours épais dans l'image où les transitions sont fortes. Les contours épais sont ensuite amincis à l'aide d'érosion de contours en utilisant des masques spécifiques. L'érosion est faite itérativement pour affiner les contours

épais, jusqu'à ce que les fronts d'érosion se rejoignent en une ligne appelé squelette. Ce squelette est pris comme position du contour. L'inconvénient de cette méthode est que les masques doivent être correctement choisis.

B- La méthode de passage par zéro :

Par définition, le contour est situé entre deux régions à forte variation de niveaux de gris. L'amplitude des transitions des niveaux de gris est non seulement globalement maximale, mais est aussi localement maximale. On peut donc utiliser deux seuils (un seuil global, un seuil local) pour effectuer une détermination de contour par « seuillage ».

Marr et Hildreth[16] ont réalisé un seuillage local en choisissant le contour qui se trouve à l'endroit où la dérivée seconde des niveaux de gris passe par zéro. Le passage par zéro de la fonction de la dérivée seconde correspond à un extrema de la dérivée première (notons que celle-ci représente l'amplitude des transitions de niveaux de gris entre régions). Donc on choisit la position où l'amplitude de transition est globalement maximale. Une position x_0 , est déclarée comme contour si $R(x)$ passe par zéro en x_0 , et si l'amplitude de ce changement ($| \frac{dR}{dx} - 1 - \frac{dR}{dx} |$) est supérieur à un seuil prédéfini, les dérivées sont calculées sur une image filtrée par un filtre Gaussien.

De la même façon, Haralick [15] utilise deux seuils pour choisir la position du contour. Une position est choisie comme contour lorsque la dérivée première des niveaux de gris dans cette position est supérieure à un seuil prédéfini et si la dérivée seconde est inférieure à un autre seuil prédéfini.

C- Extraction des extrema locaux du gradient :

Au lieu d'utiliser le passage par zéro pour choisir la transition localement maximale, Canny [16] a proposé d'extraire ce maximum local par suppression de la transition non localement maximale.

Soit $g(i, j)$, $0 \leq i \leq N-1$, $0 \leq j \leq M-1$, l'image de la norme du gradient, $g_v(i, j)$ et $g_h(i, j)$, $0 \leq i \leq N-1$, $0 \leq j \leq M-1$, les images des composantes verticales et horizontales du gradient.

On supprime les positions où le gradient est non maximal dans la sélection du gradient. Le principe est montré dans figure ci-dessous :

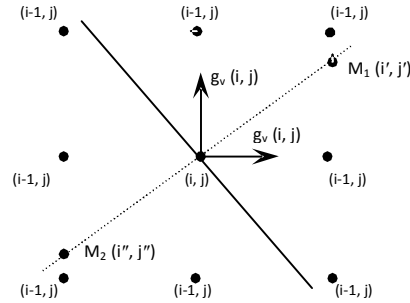


Figure.13 : Principe de suppression des transitions non maximales

Soit $g(i, j)$ la norme du gradient sur le pixel (i, j) , on cherche deux points voisins $M_1(i', j')$ et $M_2(i'', j'')$ dans la direction du gradient avec un espacement d'un pixel. Ces deux points sont calculés de la manière suivante :

$$\begin{cases} i' = i + \text{Ent} \left(\frac{(\cdot)}{(\cdot)} \right) \\ j' = j + \text{Ent} \left(\frac{(\cdot)}{(\cdot)} \right) \end{cases} \quad (\text{II-5}) \text{ et } \begin{cases} i'' = i - \text{Ent} \left(\frac{(\cdot)}{(\cdot)} \right) \\ j'' = j - \text{Ent} \left(\frac{(\cdot)}{(\cdot)} \right) \end{cases} \quad (\text{II-6})$$

où

$$G = \max \{ |g_v(i, j)|, |g_h(i, j)| \}$$

et

$$\text{Ent}(x) = \begin{cases} 1, & \text{si } x \geq 0.5 \\ 0, & \text{si } |x| < 0.5 \\ -1, & \text{si } x \leq -0.5 \end{cases}$$

Un point (i, j) est considéré comme transition maximale locale si $g(i, j) > g(i', j')$ et $g(i, j) \geq g(i'', j'')$. si cette condition n'est pas satisfaite, le pixel (i, j) ne sera pas pris en compte comme position de contour.

Avec cette méthode, un point est déclaré comme contour si le gradient est supérieur à un seuil global et s'il est un maximum local.

Par rapport à la méthode de passage par zéro, elle est plus robuste au bruit, mais elle nécessite la connaissance de la direction du contour.

II-3 Analyse d'images texturées :

Ces dernières années, on a constaté un plus grand intérêt porté vers l'analyse de l'image afin d'élaborer un système plus intelligent comparable à celui de l'être humain. Malgré ça, les capacités de ce dernier à discerner les textures restent très supérieures à celle de la machine.

II-3-1 Définition d'une signature texturale :

De façon générale, on distingue deux classes de texture : la macro texture et la micro-texture.

- Les macro-textures : présente un aspect régulier sous formes de motifs répétitifs spatialement placés selon une règle précise, d'où l'approche structurale déterministe.

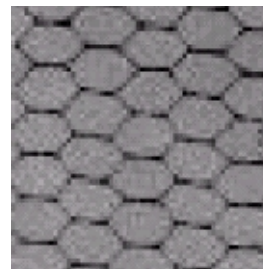
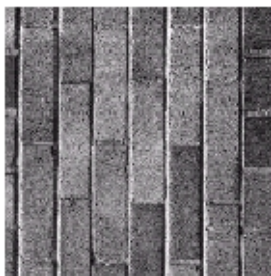


Figure.14 : Exemple de macro texture

- La micro-texture : caractérise un aspect anarchique sous forme de primitives microscopiques distribuées de manière aléatoire, d'où l'approche probabiliste.

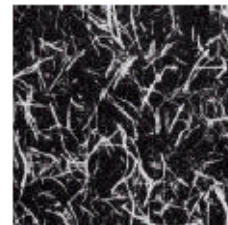
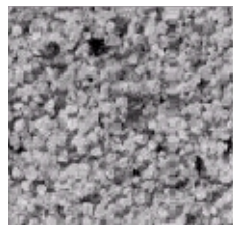


Figure. 15: Exemple de micro-texture

Une signature de texture désigne une représentation formelle caractérisant la texture. Cette dernière est souvent une quantification de la texture par une valeur ou par un vecteur de valeur.

On utilise les termes attributs ou paramètres pour designer l'ensemble de ces caractéristiques.

II-3-2 Qualité d'une signature texturale :

Une signature de texture doit exprimer au mieux toutes les caractéristiques de textures qu'elle représente. La qualité d'une signature texturale est évaluée à travers la capacité à discerner des textures sensiblement différentes et à reconnaître les textures identiques.

II-3-3 Extraction d'attributs de texture :

Après la phase de prétraitement, afin d'extraire les attributs d'une texture ou les paramètres la caractérisant, l'extraction de ces attributs constitue une étape fondamentale afin de segmenter une image en régions ayant chacune des propriétés statiques, et visuelles différentes [4]. Vu la difficulté d'une définition précise de la texture, plusieurs approches ont été élaborées ces dernières années pour identifier cette opération. Ces approches peuvent être de nature:

➤ Statistiques :

Elles se réalisent soit en exploitant les propriétés statistiques de la texture en réécrivant l'image de texture dans un plan transformé (la densité spectrale, méthode des extrema locaux, méthodes de transformation de Fourier) ; soit en utilisant directement ces propriétés statistiques (matrice de cooccurrence, matrice de voisinage, modèle de Markov,...).

➤ Spatio-fréquentielles :

Elles préservent les informations globales et locales, elles sont adaptées aux signaux quasi périodiques sachant que les textures sont des signaux quasi périodiques ayant une énergie fréquentielle localisée.

Parmi les méthodes spatio-frequentilles, on peut citer :

- La transformée de Fourier.
- La transformation en ondelettes.
- ...

➤ **Structurelles ou déterministes :**

Ce sont des méthodes adaptées aux textures macroscopiques basées sur l'information structurelle et contextuelle d'une forme. Ces méthodes utilisent principalement des techniques d'auto-corrélation afin de retrouver l'emplacement des primitives des textures initialement extraites.

Ces méthodes se déroulent en deux étapes : Extraction des la primitive puis la recherche de règle de placement.

II-4 Quelques techniques de segmentation d'images:

Dans les méthodes de premier ordre, qui correspondent à une description de l'histogramme des niveaux de gris, il n'y a pas d'informations sur la localisation du pixel. Il est donc nécessaire d'utiliser des méthodes d'ordre supérieur pour une analyse plus précise. L'ordre des méthodes est donné par le nombre de pixels mis en jeu dans le calcul des paramètres.

II-4-1 La matrice de cooccurrence :

La matrice de cooccurrence M_c détermine la fréquence d'apparition d'un motif, formé de deux pixels reliés entre eux selon deux paramètres : la distance d et l'angle θ .

Elle est de taille $N_g \times N_g$, où N_g correspond au niveau de gris maximum de l'image.

On la note $M_c(d, \theta)$. [3]

Cette distance représente le pas de déplacement de deux pixels d'une valeur égale à 1, 2, 3, 4, etc., mais généralement prise égale à 1. L'angle θ est la direction, par rapport à l'horizontale, suivant laquelle ces deux pixels sont comparés. Ses valeurs sont : $0^\circ, 45^\circ, 90^\circ$ et 135° . La taille de M_c peut être réduite, afin de ne pas avoir une matrice de taille trop élevée, en divisant les niveaux de gris de l'image soit sur 8, 16 ou 32.

Exemple :

Soit une image donnée (Figure.16). On extrait une partie représentée sous forme de matrice où chaque pixel est remplacé par une valeur caractérisant son niveau de gris. Enfin, la matrice de cooccurrence est calculée pour une distance d et une direction $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$. On prend, par exemple, le couple (1,4) qui est répété deux fois au niveau de la matrice de niveaux de gris. Alors, on va placer le nombre 2, dans la case correspondante à l'intersection 1-4 (case entourée). On voit bien que la matrice de

cooccurrence trouvée est une matrice carrée de taille 4×4 , où 4 correspond au niveau de gris maximum de (b).

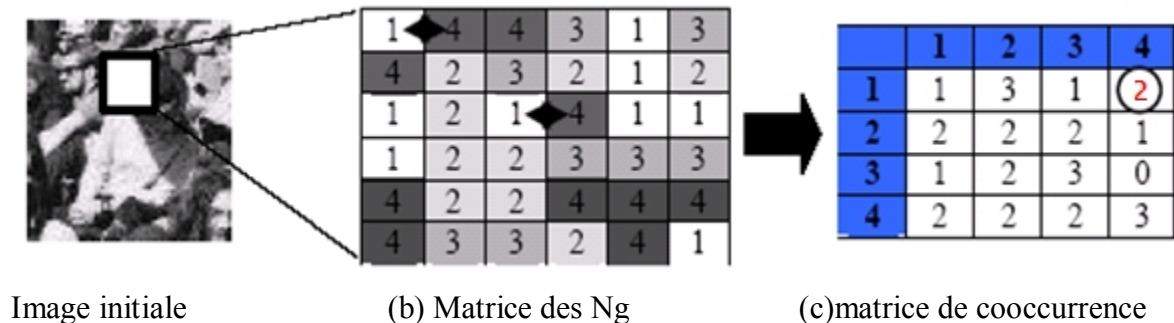


Figure.16 : Méthode de calcul de la matrice de cooccurrence

II-4-2 Matrices de longueur de plage (dite aussi matrice des iso-segments):

Une plage de niveau de gris est un ensemble de pixels consécutifs, dans une direction donnée ayant le même niveau de gris. La longueur de la plage est le nombre de pixels dans cet ensemble. A chaque direction, on peut associer une matrice de longueur de plage. Cette méthode détermine l'ensemble des plages dans l'image et effectue des statistiques sur les longueurs.

Il s'agit donc de rechercher le nombre maximum de pixels adjacents ayant le même niveau de gris dans une direction. On définit alors des primitives qui ont pour caractéristiques un niveau de gris, une longueur et une direction.

II-4-3 Les différences des niveaux de gris :

C'est une approche statistique qui consiste à effectuer la différence entre les niveaux de gris entre paire de pixels d'une région donnée qui seront, ensuite, cumulées dans un tableau monodimensionnel. On les utilisera pour la construction de l'histogramme des niveaux de gris.

II-4-4 Modélisation basée sur les niveaux de gris ordonnés :

Cette méthode réalise d'abord un filtrage en utilisant un filtre non linéaire, basé sur les rangs des niveaux de gris, elle détecte en suite les contours en utilisant la différence entre le niveau de gris de rang maximum et celui de rang minimum dans une fenêtre. Elle est décrite comme suit :

Soit (i, j) le pixel en cours, on choisit 8 fenêtres non chevauchantes de taille k pixels autour de (i, j) .

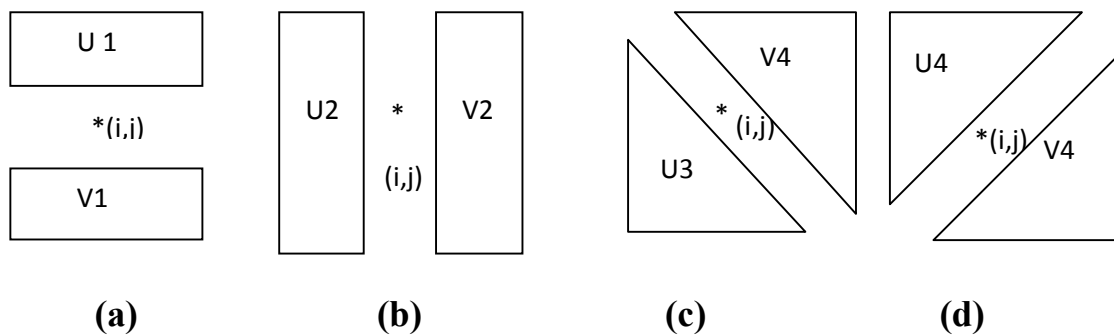


Figure.17 : Fenêtre de détection de contours

Pour chaque fenêtre, on ordonne les niveaux de gris et l'amplitude du contour pour le pixel est calculée par la formule suivante :

$$d_{ij} = \max_k |U_k - V_k| \quad (II-7)$$

Où d est une distance entre les niveaux de gris ordonnés en correspondance dans les fenêtres U_k et V_k , $K= 1, \dots, 4$.

Cette distance peut être la distance Euclidienne, celle de « City Block » ou celle de Minkovski, etc.

L'avantage de cette méthode est qu'elle ne nécessite pas de modélisation de la texture. Elle peut détecter correctement les contours dans de nombreuses images réelles, mais reste problématique pour les images très texturées.

Discussion :

Dans ce chapitre, nous avons abordé les notions de segmentations, contours et textures de manière générale.

Dans ce travail, nous proposons une modélisation des régions à l'aide d'une fonction monodimensionnelle de degré 1. Cette fonction est obtenue par approximation des niveaux de gris ordonnés dans une fenêtre carrée. Cette modélisation sera présentée en détail dans le chapitre qui va suivre.

CHAPITRE III

SEGMENTATION
PAR MODÉLISATION
BASÉE SUR LES
NIVEAUX DE GRIS
ORDONNÉS

Préambule :

Dans une image réelle, afin d'avoir des régions uniformes et homogènes, il faut rechercher des propriétés pour les caractériser. Cela est obtenu en modélisant ces régions. Le modèle peut être construit avec des fonctions polynomiales, un processus stochastique ou des paramètres statistiques.

Dans ce travail, nous proposons une modélisation à l'aide d'une fonction monodimensionnelle de degré 1. Cette fonction est obtenue par approche des niveaux de gris ordonnés dans une fenêtre carrée. Ce modèle considère les contours comme une distribution spécifique des niveaux de gris dans une région.

III- 1 Définition de niveaux de gris ordonnés (NGO) :

Soit une fenêtre carrée de taille K pixels, centrée sur le pixel (i, j) dans une image

$f(i, j)$ ($0 \leq i \leq M - 1$, $0 \leq j \leq N - 1$) où $f(i, j)$ représente le niveau de gris du pixel (i, j) . On repère d'abord les pixels de cette fenêtre par les entiers de 1 à K comme montre dans la figure ci-dessous.

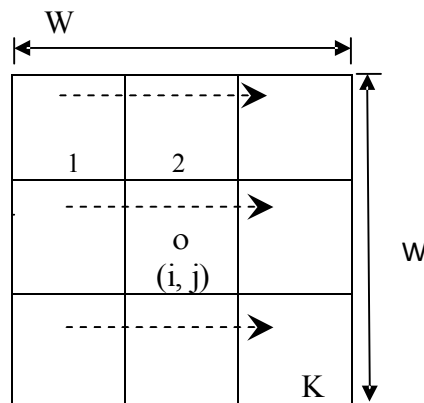


Figure.18 Numérotation des pixels dans la fenêtre.

On note ensuite l'ensemble des niveaux de gris des pixels de cette fenêtre par :

$$Y = (y_1, y_2, \dots, y_K) \quad (\text{III-1})$$

Où y_i , $i=1, \dots, K$ est alors le niveau de gris du pixel numéro i .

Puis, on trie les éléments de Y dans un ordre croissant :

$$Y(1) \leq Y(2) \leq \dots \leq Y(K) \quad (\text{III-2})$$

Où $y_{(i)}$, $i=1, \dots, K$ est le niveau de gris de rang i . pour faciliter la notation, on représente les rangs et les niveaux de gris ordonnés par les deux vecteurs ci-dessous :

*. Le vecteur de rangs : $R = (1, 2, \dots, K)^T$.

*. Le vecteur de niveaux de gris ordonnés : $Y = (y_{(1)}, y_{(2)}, \dots, y_{(K)})^T$.

III-2 Modélisation de région par régression sur les NGO :

III-2-1 Un modèle linéaire :

Dans notre étude, on considère une région comme une distribution des niveaux de gris des pixels. Le tri des niveaux de gris permet de choisir une fonction polynomiale de degré 1 pour approcher ces distributions. La figure.19 nous donne un exemple de la distribution des niveaux de gris ordonnés et leur approximation par une fonction linéaire.

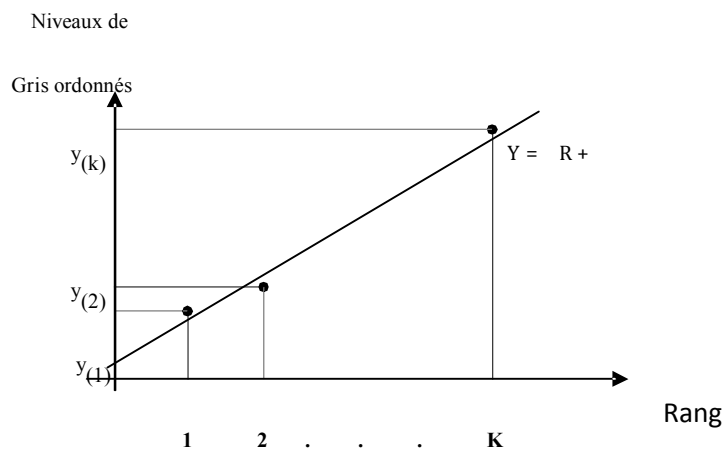


Figure.19 Les niveaux de gris ordonnés et leur approximation.

Le choix de la forme de la fonction est assez important pour l'approximation. On espère que cette fonction sera simple et robuste au bruit, et assez précise pour discriminer au mieux les distributions de niveaux de gris dans des régions différentes.

Pour des raisons de simplicité, nous avons choisi une fonction simple comme modèle. La fonction proposée est définie comme suit :

$$= \mathbf{a} \cdot \mathbf{R} + \mathbf{b} \quad (\text{III-3})$$

Où $\mathbf{I} = (1, \dots, l)^T$, $\mathbf{R} = (1, \dots, K)$ est le vecteur associé aux K rangs, K la taille de la fenêtre, $\hat{\mathbf{y}} = (\hat{y}_{(1)}, \dots, \hat{y}_{(K)})^T$ le vecteur estimé des niveaux de gris ordonnés et $\mathbf{a}$ et $\mathbf{b}$ sont des constantes.

Comme le montre la figure.19, $\mathbf{a}$ représente la pente de la droite, $\mathbf{b}$ l'ordonnée à l'origine. En utilisant ce modèle, une région sera représentée par deux paramètres : $\mathbf{a}$ représente le changement de niveaux de gris dans cette région, $\mathbf{b}$ représente le résultat d'un filtre non linéaire sur l'ensemble des niveaux de gris de cette région.

La méthode décrite ci-après, permet de calculer les paramètres $\mathbf{a}$ et $\mathbf{b}$.

Soit $\mathbf{y} = (y_{(1)}, y_{(1)}, \dots, y_{(l)})^T$ le vecteur de niveaux de gris ordonnés, $\hat{\mathbf{y}} = (\hat{y}_{(1)}, \dots, \hat{y}_{(K)})^T$ le vecteur estimé des niveaux de gris ordonnés. On cherche une estimation qui minimise l'erreur quadratique :

$$\min_{\mathbf{a}, \mathbf{b}} \sum_{i=1}^l (y_{(i)} - \mathbf{a} \cdot \mathbf{R}_i - \mathbf{b})^2 = \{\min \{ \sum_{i=1}^l (y_{(i)} - \mathbf{a} \cdot \mathbf{R}_i - \mathbf{b})^2 \} \quad (\text{III-4})$$

Soit :

$$\min \{ (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y})^T (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y}) \}$$

En annulant les dérivées premières :

$$\begin{cases} -[(\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y})^T (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y})] = 0 \\ -[(\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y})^T (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y})] = 0 \end{cases}$$

Après simplification, on obtient :

$$2 \mathbf{R}^T (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y}) = 0$$

$$2 \mathbf{I}^T (\mathbf{a} \cdot \mathbf{R} + \mathbf{b} \cdot \mathbf{I} - \mathbf{y}) = 0$$

La solution est :

$$\left\{ \begin{array}{l} \mathbf{a} = \frac{\sum_{i=1}^I \bar{I}}{\sum_{i=1}^I \bar{I}} \\ \mathbf{b} = \frac{\sum_{i=1}^I \bar{I}}{\sum_{i=1}^I \bar{I}} - \frac{\sum_{i=1}^I \bar{I}}{\sum_{i=1}^I \bar{I}} \\ \bar{I}^T \bar{I} = K \end{array} \right. \quad \text{où} \quad \left\{ \begin{array}{l} \mathbf{R}^T \mathbf{R} = \sum_{i=1}^I \mathbf{i}^2 = \text{---} (\text{---}) \\ \mathbf{R}^T \bar{\mathbf{I}} = \bar{\mathbf{I}}^T \mathbf{R} = \sum_{i=1}^I \mathbf{i}^2 = \text{---} (\text{---}) \\ \mathbf{R}^T \mathbf{Y} = \sum_{i=1}^I \mathbf{i} * y_{(i)} \\ \bar{\mathbf{I}}^T \mathbf{Y} = \sum_{i=1}^I y_{(i)} \end{array} \right.$$

Pour simplifier, on réécrit $\mathbf{a}$ et $\mathbf{b}$ comme suit :

$$\mathbf{a} = \frac{\sum_{i=1}^I \text{---} ()}{(\text{---}) /} \quad (\text{III-5})$$

$$\mathbf{b} = \frac{\sum_{i=1}^I ()}{\text{---}} - \text{---} * \mathbf{a} \quad (\text{III-6})$$

$$\mathbf{b} = \text{---} \sum_{i=1}^I (2K+1-3i) * y_{(i)} \quad (\text{III-7})$$

On notera, en particulier, qu'on aura toujours $a \geq 0$.

A- Caractéristiques de $\mathbf{a}$:

Dans la formule (III-5), le dénominateur est une constante pour une fenêtre fixée. On peut donc réécrire la formule comme suit :

$$\mathbf{a} = \text{---} = \frac{\sum_{i=1}^I \text{---} ()}{(\text{---}) /} \quad (\text{III-8})$$

Où

$$\text{---} = \sum_{i=1}^I \mathbf{i} - \text{---} y_{(i)} = \sum_{i=1}^I \mathbf{c} y_{(i)} \quad (\text{III-9})$$

Les différentes valeurs de C_i (pour K impair) sont indiquées par la figure ci-dessous

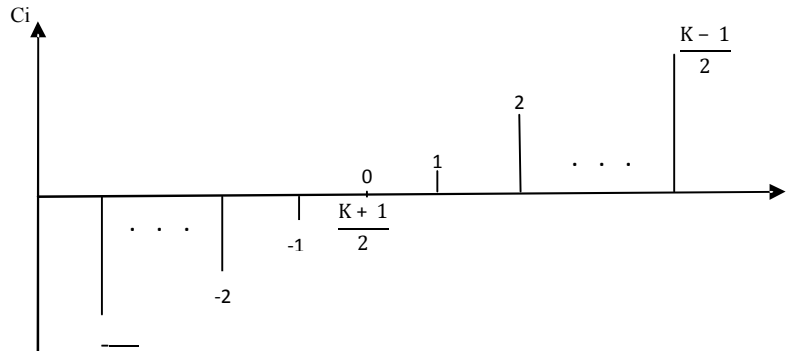


Figure.20 : Distribution des coefficients C_i

Notons que la distribution des C_i est symétrique par rapport au centre $\frac{K+1}{2}$. Plus i s'éloigne de cette valeur, plus la valeur de C_i est importante. On peut réécrire Δ sous la forme :

$$\Delta = \frac{1}{K} * (Y_{(K)} - Y_{(1)}) + \dots + 2 * (Y_{(\frac{K+1}{2} + 2)} - Y_{(\frac{K+1}{2} - 2)}) + 1 * (Y_{(\frac{K+1}{2} + 1)} - Y_{(\frac{K+1}{2} - 1)}) \quad (\text{III-10})$$

La valeur de Δ est obtenue en soustrayant les niveaux de gris les plus faibles des niveaux de gris les plus forts. Cette différence est pondérée par un poids qui diminue lorsque l'on se rapproche des niveaux de gris centraux. Ainsi Δ modélise le taux de changement de niveaux de gris dans la région.

Intuitivement, Δ est proche de la notion de « variation » sur la région, la variation peut être représentée comme suit :

$$D = - \sum_{i=1}^K |y_i - m| \quad (\text{III-11})$$

Où $m = \frac{1}{K} \sum_{i=1}^K y_i$.

La formule (III-11) montre que D est la somme des différences entre y_i , $i=1, \dots, K$, et la moyenne des niveaux de gris dans cette fenêtre. Les pondérations pour chaque différence sont $1/K$. Par contre, Δ mesure la somme des différences entre y_i , $i=1, \dots, K$, et la

médiane $Y_{(-)}$, de plus, les coefficients pondérateur de chaque différence ne sont pas les mêmes. Par ailleurs, on peut réécrire :

$$\begin{aligned} \sigma &= \sum_{i=1}^K \left(Y_{(-+i)} - Y_{(-i)} \right) \\ &= \sum_{i=1}^K \left(\left| Y_{(-+i)} - Y_{(-)} \right| + \left| Y_{(-i)} - Y_{(-)} \right| \right) \end{aligned} \quad (\text{III-12})$$

Au sens statistique, la valeur médiane conserve mieux les niveaux de gris originaux (que la moyenne par exemple). Donc si la fenêtre en cours se trouve dans la transition entre deux régions, σ sera supérieur à la variation. En effet, dans ce cas, la médiane $Y_{(-)}$ représente plutôt le vrai niveau de gris d'une de ces deux régions et σ représente donc la différence des niveaux de gris entre deux régions. En revanche, la variation ne correspond pas directement aux différences de niveaux de gris des pixels entre ces deux. σ Peut donc être utilisé pour décrire la variation de niveaux de gris dans une région, ou pour détecter les contours.

La distribution des niveaux de gris ordonnés $Y_{(i)}$, $i = 1, \dots, K$ dans le cas d'existence d'un contour entre deux régions est schématisée par la figure.21. Dans le cas d'une région uniforme plus un bruit normal, on aura le schéma de la figure.22.

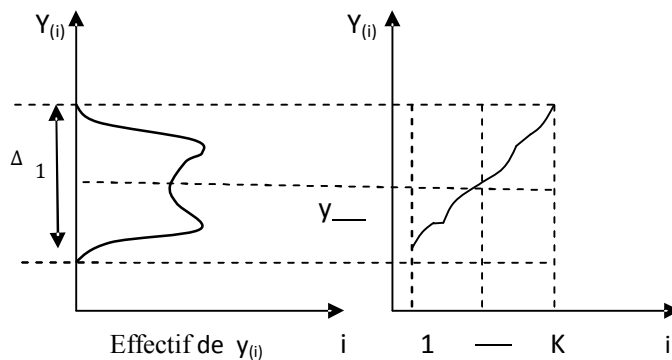


Figure.21 : NGO au voisinage
d'un contour

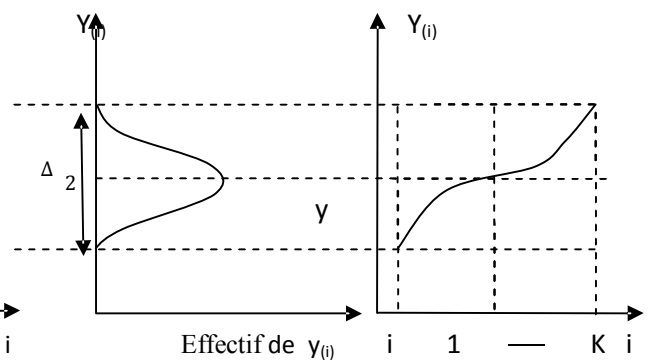


Figure.22 : NGO dans une
région homogène

Dans la figure.20, la somme des valeurs de $Y_{(K-i+1)} - Y_{(i)}$, $i=1,2,\dots,K$ sera plus grande que dans la figure.22 sous l'hypothèse : $\Delta Y_1 \geq \Delta Y_2$. Le coefficient σ sera donc plus important dans le cas d'un contour entre deux régions que dans celui d'une région homogène.

Etant obtenu en ordonnant les niveaux de gris, σ est donc invariant par rapport à la direction des contours. La figure.23 montre 8 contours de régions pour des directions variées, cependant, le coefficient σ est le même.

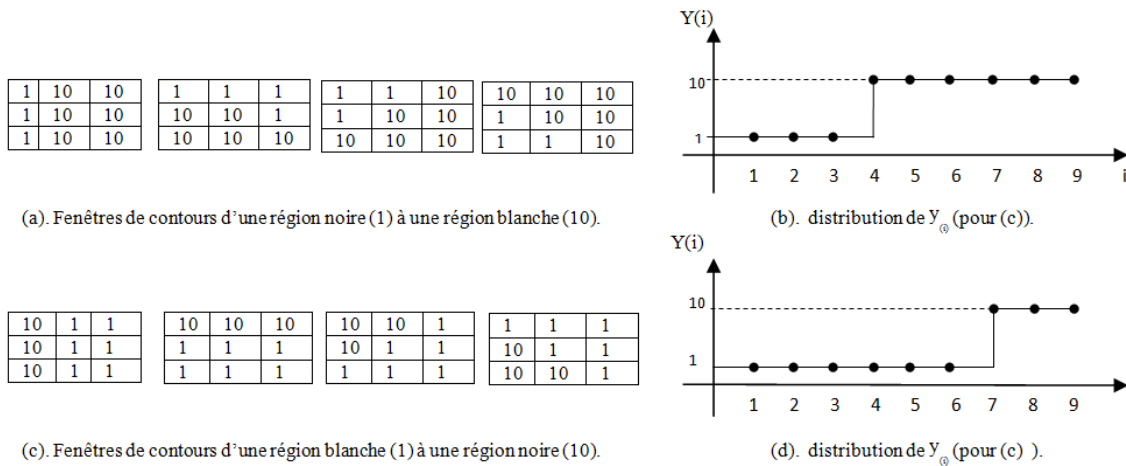


Figure.23 : Invariance de σ par rapport a la direction du contour

La valeur σ correspond aux contours dans les différentes directions.

Note : la taille de la fenêtre dans l'exemple précédant est $K=9$.

Pour toutes les fenêtres :

$$\sigma = 4*(10-1)+3*(10-1)+3*(10-1)+1*(10-1) = 81$$

B- Caractéristiques de b :

Dans la formule (III-6), on voit que b est composé de 2 termes : le premier est la moyenne des niveaux de gris dans la fenêtre considérée, le second, $[-(K+1)/2*a]$, comporte les informations sur la variation des niveaux de gris. Dans une région uniforme, a est petit, donc b se rapproche de la moyenne des niveaux de gris. Par contre dans une région qui possède des contours, a est grand et positif, donc b est petit. Cet effet montre que b est systématiquement inférieur à la moyenne des niveaux de gris de la fenêtre.

On peut réécrire la formule (III-7) sous la forme :

$$= \frac{1}{2K-1} \sum_{i=1}^{2K-1} (2K-1-i+1) d_i = \frac{1}{2K-1} \sum_{i=1}^{2K-1} (2K-i) d_i \quad (III-13)$$

d_i est positif pour $i \leq 2K-1$ sinon i est négatif. La figure.24 montre l'évolution des coefficients d_i en fonction de i .

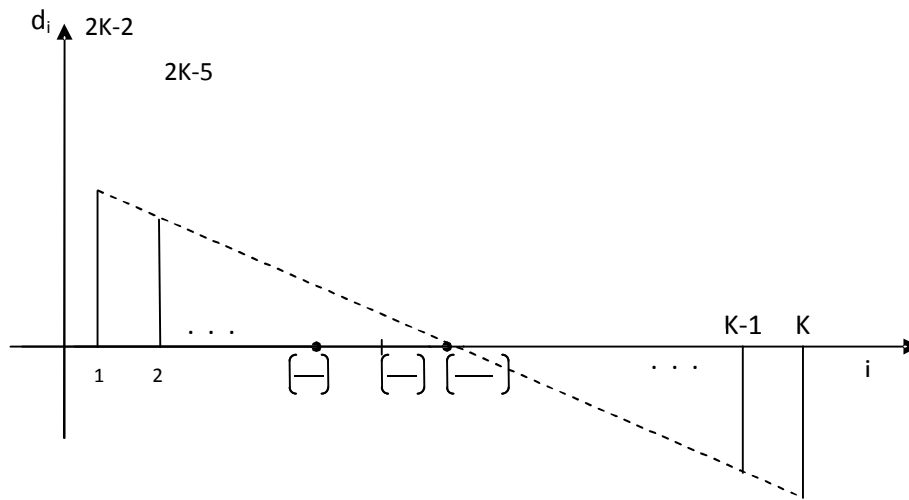


Figure.24 : Les coefficients d_i pour calculer $\mathbf{b}$

Entre $i=1$ et $i=2K-1$, d_i est positif, il permet donc de sauvegarder le niveau de gris moyen des valeurs centrales de la fenêtre. Pour $i < 1$ et $i > 2K-1$, d_i a des signes différents pour les deux parties. Les accumulations de ces deux parties représentent les différences entre les niveaux de gris faibles et les niveaux de gris importants de la fenêtre. Cet effet montre que s'il y a des changements de niveaux de gris dans une fenêtre, $\mathbf{b}$ sera petit. Car il y a des compensations entre les deux parties.

En conclusion, $\mathbf{b}$ peut être utilisé pour caractériser une région, ou bien pour détecter les contours. Selon les discussions ci-dessus, si l'on veut obtenir les régions plus uniformes, on peut choisir $\mathbf{b}$ en utilisant le premier terme dans la formule (III-6), c'est-à-dire proche de la moyenne des niveaux de gris dans une fenêtre.

III-2-2 Détection de contours par la modélisation s'appuyant sur NGO :

La détection de contours se compose des deux étapes suivantes :

- Détection des transitions de niveaux de gris.
- Localisation précise des positions contours.

La détection de transition peut se faire par plusieurs manières telles que le calcul des gradients, ou l'utilisation des masques. La localisation des contours peut se faire par seuillage sur les transitions.

La méthode proposée ici pour la détection des contours, commence par détecter les transitions des niveaux de gris en utilisant les paramètres **a** et **b** de la formule (III-3), puis localise les contours par un seuillage à l'aide des formes locales géométriques du contour.

III-2-2-1 Détection des transitions de niveaux de gris en utilisant le premier paramètre dans le modèle :

Il existe plusieurs méthodes pour détecter ces transitions. Nous nous intéressons aux méthodes d'approximations fonctionnelles de la surface de l'objet. Les avantages de ce type de méthode sont la robustesse au bruit et la détection précise des transitions.

Comme il apparait dans la figure.23, le paramètre **a** donne correctement les transitions de niveaux de gris entre régions. L'invariance de **a** par rapport à la direction du contour permet de l'utiliser comme un opérateur isotrope pour détecter les transitions de niveaux de gris entre régions.

Par rapport aux autres méthodes utilisant soit une fonction bidimensionnelle ou monodimensionnelle de degré supérieur à 1, notre méthode est notablement plus simple.

III-2-2-2 Détection des transitions de niveaux de gris en utilisant la multiplication du premier paramètre par la variance du deuxième :

Pour utiliser les informations contenues dans le paramètre **b**, nous avons proposé un autre opérateur de détection des transitions importantes de niveaux de gris, cet opérateur est $a * \text{Var}(\mathbf{b})$. C'est-à-dire qu'on calcule d'abord (**a** , **b**) pour chaque pixel, puis on calcule la variance de **b** pour chaque pixel avec une fenêtre. Puisque la variance de **b** est un opérateur isotrope et qu'elle est grande entre les régions différentes en niveaux de gris, $a * \text{Var}(\mathbf{b})$

renforce alors les transitions des niveaux de gris entre régions et supprime les bruits dans une image.

Deux méthodes ont été proposées pour la localisation de contours. Ces deux méthodes reposent sur un double seuillage : global et local. Le seuillage global se fait en choisissant un seuil empirique ; le seuillage local se fait en choisissant automatiquement un seuil local.

La première méthode proposée calcule le seuil à partir d'une méthode de classification, et la deuxième méthode choisit le seuil global à l'aide des formes géométriques locales des contours.

A- Méthode de classification :

Pour calculer le seuil local, on choisit d'abord une fenêtre sur l'image des transitions de niveaux de gris (l'image de $\mathbf{a}_{ij}$ ou $\mathbf{a}_{ij} * \text{Var}(\mathbf{b}_{ij})$). On classe en suite les transitions dans cette fenêtre en deux classes. Une classe contient les transitions locales fortes, l'autre les transitions locales faibles. Le pixel central d'une fenêtre est déclaré comme point de transition locale maximale si la transition se trouve dans la classe des transitions locales fortes.

La classification des transitions en deux classes se fait à l'aide de la méthode de Fisher. Elle nous fournit le seuil local optimal. Elle utilise comme critère d'optimalité la minimisation de la somme des inerties des classes [14], calculé avec la formule III-14:

$$W(p) = \sum_{i=1}^N \sum_{j=1}^N (a_{ij} - \mu)^2 \quad (\text{III-14})$$

Ou

$$G(c_n) = \frac{\sum_{i=1}^N f(x_i) \cdot (x_i - \mu_c)^2}{\sum_{i=1}^N f(x_i)}$$

- N_c désigne le nombre de classes.
- $f(x)$, l'histogramme monodimensionnel étudié.

Cette méthode est facile à mettre en œuvre, et elle peut produire des contours fins. Son inconvénient est qu'on ne peut pas garantir des contours de largeur égale à un pixel. Pour cette raison, on propose la méthode suivante.

B- méthode de rangs :

Pour simplifier le problème, on considère localement un contour dans une fenêtre de 3x3 pixels. Les formes géométriques locales d'un contour peuvent être schématisées comme sur la figure.25. Il y a 4 formes géométriques de base pour une fenêtre de 3x3 pixels. Par rotation de 45° de ces 4 formes, on peut obtenir les 24 autres formes géométriques possibles pour un contour dans une fenêtre.

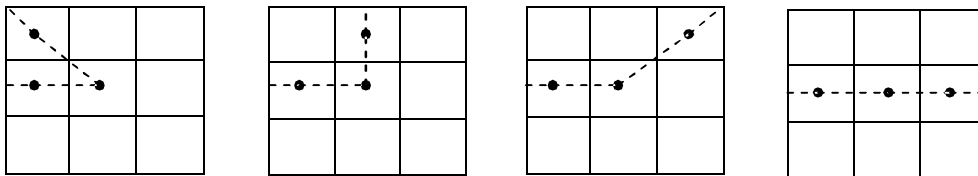


Figure.25 : Les formes géométriques locales des contours dans une fenêtre de 3x3 pixels

A partir de cette modélisation locale d'un contour, un pixel peut être assigné à un contour selon son rang de transition de niveaux de gris dans cette fenêtre.

Notons que dans chaque forme locale, il n'y a que trois pixels de transitions locales maximales dans la fenêtre. C'est-à-dire, que pour une fenêtre donnée, on peut connaître le rang des transitions de niveaux de gris de chaque pixel dans cette fenêtre. Les points qui se trouvent aux trois premiers plus grands rangs de transitions de niveaux de gris sont les maxima locaux. Le seuillage local peut donc être calculé comme suit :

Pour une fenêtre de 3x3 pixels, on calcule les rangs des transitions de niveaux de gris de chaque pixel dans cette fenêtre. Si le rang de la transition de niveaux de gris du pixel central se trouve dans les trois premiers plus grands rangs, ce point est alors décalé comme un maximum local de transition des niveaux de gris. Dans cette implémentation, on ne considère pas l'information sur la forme géométrique locale des contours dans la fenêtre.

III-3 Mise en œuvre de la détection de contour :

III-3-1 Détection de contour en utilisant le paramètre a de la fonction linéaire :

Cet algorithme consiste à détecter les transitions de niveaux de gris en utilisant le paramètre a du modèle linéaire, et à faire le seuillage à l'aide d'une méthode de classification ou d'une méthode de rangs de transitions.

A- Algorithme C_a. (Paramètre a + Classification de a)

Début

Etape 0 : initialisation : Taille de l'image (NbLigne, NbColonne).

Taille K de la fenêtre.

Paramètre P pour le calcul du seuil global S_g .

Etape 1 : calcul de a pour chaque pixel.

Pour i variant de 1 à Nbligne Faire

Pour j variant de 1 à NbColonne Faire

- Calcul des niveaux de gris ordonnés $y_{(1)} \leq y_{(2)} \leq \dots \leq y_{(K)}$.
- Calcul des a_{ij} à l'aide de la formule (III-5).

FinFaire

FinFaire

Etape 2 : Localisation du contour.

-Calcul du seuil global par $S_g = P * (\text{moyenne des } a_{ij})$

-Pour i variant de 1 à NbLigne Faire

Pour j variant de 1 à NbColonne Faire

- Calcul du seuil local S_c pour une fenêtre de taille 3×3 pixels par l'algorithme de Fisher.
- Si $a_{ij} \geq S_c$ et $a_{ij} \geq S_g$ alors $r_{ij} = 0$ ((i,j) =pixel contour)

Sinon $r_{ij} = 255$ ((i,j) =pixel non contour).

Finfaire

FinFaire

Etape 3 : Enregistrement des résultats r_{ij} dans une image.

Fin

B- Algorithme R_a. (Paramètre **a** + Rangs des transitions de **a**)

Début

Etape 0 : initialisation : Taille de l'image (NbLigne, NbColonne).

Taille **K** de la fenêtre.

Paramètre **P** pour le calcul du seuil global S_g .

Etape 1 : calcul de **a** pour chaque pixel.

Pour **i** variant de 1 à NbLigne Faire

Pour **j** variant de 1 à NbColonne Faire

- Tri des niveaux de gris ordonnés $y_{(1)} \leq y_{(2)} \leq \dots \leq y_{(K)}$.
- Calcul des a_{ij} à l'aide de la formule (III-5).

FinFaire

FinFaire

Etape 2 : Localisation du contour.

-Calcul du seuil global par $S_g = P * (\text{moyenne des } a_{ij})$

-Pour **i** variant de 1 à NbLigne Faire

Pour **j** variant de 1 à NbColonne Faire

- Calcul du rang de la transition RT_{ij} du pixel (**i**, **j**) dans une fenêtre de taille 3x3 pixels centrée sur (**i**, **j**).
- Si $a_{ij} \geq S_c$ et $a_{ij} \geq S_g$ alors $r_{ij} = 0$ ((**i**,**j**)=pixel contour)

Sinon $r_{ij} = 255$ ((**i**,**j**)=pixel non contour).

Finfaire

FinFaire

Etape 3 : Enregistrement des résultats r_{ij} dans une image.

Fin

III-3-2 Détection de contours utilisant le produit du paramètre et la variance du deuxième paramètre :

Cet algorithme consiste à détecter les transitions de niveaux de gris en utilisant le paramètre $a * \text{Var}(b)$ dans le modèle linéaire et à faire un seuillage à l'aide d'une méthode de classification ou à l'aide d'une méthode à rangs de transition. la variance de b est calculée avec une fenêtre de la même taille que pour le calcul de a et b dans la modélisation.

A- Algorithme $R_{a * \text{Var}(b)}$ (Paramètre $a * \text{var}(b)$ + Rangs des transitions)

Début

Etape 0 : initialisation : Taille de l'image (NbLigne, NbColonne).

Taille K de la fenêtre.

Paramètre P pour le calcul du seuil global S_g .

Etape 1 : calcul de a pour chaque pixel.

Pour i variant de 1 à Nbligne Faire

Pour j variant de 1 à NbColonne Faire

- Tri des niveaux de gris ordonnés $y_{(1)} \leq y_{(2)} \leq \dots \leq y_{(K)}$.
- Calcul des a_{ij} à l'aide de la formule (III-5).
- Calcul des b_{ij} à l'aide de la formule (III-6).
- Calcul de la variance de b_{ij} .

FinFaire

FinFaire

Etape 2 : Localisation du contour.

-Calcul du seuil global par $S_g = P * (\text{moyenne des } a_{ij})$

-Pour i variant de 1 à NbLigne Faire

Pour j variant de 1 à NbColonne Faire

- Calcul du rang de la transition RT_{ij} du pixel (i, j) dans une fenetre de taille 3x3 pixels centrée sur (i, j).
- Si $a_{ij} \geq S_c$ et $a_{ij} \geq S_g$ alors $r_{ij} = 0$ ((i,j)=pixel contour)

Sinon $r_{ij} = 255$ ((i,j)=pixel non contour).

Finfaire

FinFaire

Etape 3: Enregistrement des résultats r_{ij} dans une image.

Fin

B- Algorithme C_a*var(b) (Paramètre **a*var(b)**+ Classification)

Etape 0: initialisation : Taille de l'image (NbLigne, NbColonne).

Taille K de la fenêtre.

Paramètre **P** pour le calcul du seuil global S_g .

Etape 1: calcul de a pour chaque pixel.

Pour i variant de 1 à Nbligne Faire

Pour j variant de 1 à NbColonne Faire

- Tri des niveaux de gris ordonnés $y_{(1)} \leq y_{(2)} \leq \dots \leq y_{(K)}$.
- Calcul des a_{ij} à l'aide de la formule (III-5).
- Calcul des b_{ij} à l'aide de la formule (III-6).
- Calcul de la variance de b_{ij} .

FinFaire

FinFaire

Etape 2: Localisation du contour.

-Calcul du seuil global par $S_g = P * (\text{moyenne des } a_{ij})$

-Pour i variant de 1 à NbLigne Faire

Pour j variant de 1 à NbColonne Faire

- Calcul du rang de la transition RT_{ij} du pixel (i, j) dans une fenetre de taille 3×3 pixels centrée sur (i, j) .
- Si $a_{ij} \geq S_c$ et $a_{ij} \geq S_g$ alors $r_{ij} = 0$ ((i,j) =pixel contour)

Sinon $r_{ij} = 255$ ((i,j) =pixel non contour).

Finfaire

FinFaire

Etape 3 : Enregistrement des résultats r_{ij} dans une image.

Fin

Discussion :

Nous avons présenté dans ce chapitre les différentes étapes de la segmentation basée des niveaux de gris ordonnés en utilisant quatre algorithmes différents.

Nous avons opté pour cette méthode car elle est facile à mettre en œuvre et ne nécessite pas de modélisation de la texture. Son inconvénient, est que la quantité d'information à manipuler ainsi que le temps de calcul des matrices intermédiaires sont assez importants.

Les résultats de la segmentation seront exposés et discutés dans le chapitre qui suivra.

CHAPITRE IV

TESTS

ET

RÉSULTATS

IV-Présentation des données :

Afin d'évaluer notre méthode, nous l'avons appliquée sur deux types d'images à savoir:

- Des images tests tirées de l'album de Brodatz.
- Des images médicales.

Les images utilisées sont des images en niveaux de gris.

IV-1 Les images utilisées:

IV-1-1 Images de l'album de BRODATZ :

La première application de notre méthode de segmentation fondée sur les niveaux de gris ordonnés, est sur des images tirées de cet album(BRODATZ). Nous avons choisi une image (l'image MUR DE BRIQUE). Nous avons également retenu l'image LENNA qui est énormément utilisée comme référence de résultat en traitement d'images.

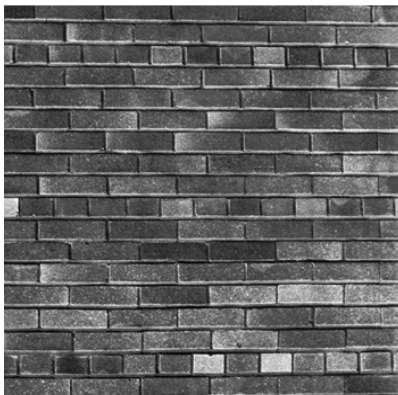


Image MUR (A)



Image LENNA (B)

Figure.26 : Images tests originales

IV-1-2 Les images médicales utilisées :

Pour l'application de notre méthode, on a utilisé des images récupérées au centre d'imagerie médicale MAHMMOUDI. Nous donnerons le résultat de segmentation d'une image qui représente la coupe axiale cérébrale d'un enfant atteint d'une tumeur sur la fosse cérébrale postérieure.

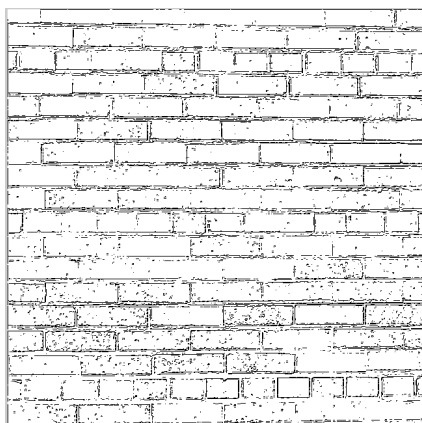
IV-1-3 Matériel et logiciel utilisés:

Pour l'élaboration de cette méthode, nous avons utilisé une machine à CPU Pentium dual-core sur laquelle nous avons installé le logiciel Matlab qui est très efficace dans les calculs et la visualisation graphique. Son utilisation est souvent motivée par ses différents avantages qui sont :

- La structure de base de Matlab est la matrice, il a été conçu pour manipuler des données matricielles.
- C'est un langage de haut niveau, il nous permet de programmer de manière très rapide.
- C'est un logiciel de calcul scientifique et de visualisation graphique dédié, plus particulièrement, aux applications numériques.
- Il permet un gain considérable de temps de développement des programmes.
- Il offre une guide très complet («help commande» en générale) («help images» pour l'imagerie)

IV-2 Résultats obtenus sur les images tests :

IV-2-1 En utilisant une fenêtre 3x3 :

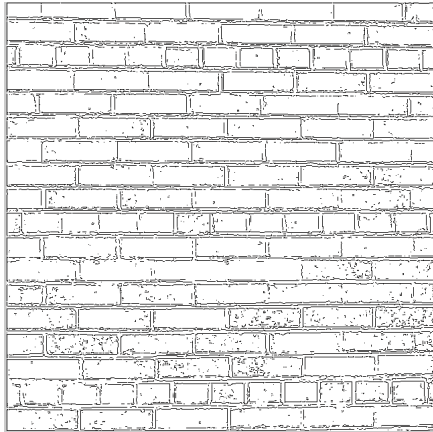


(A)



(B)

Figure.27 : Résultat de segmentation avec l'algorithme C-a

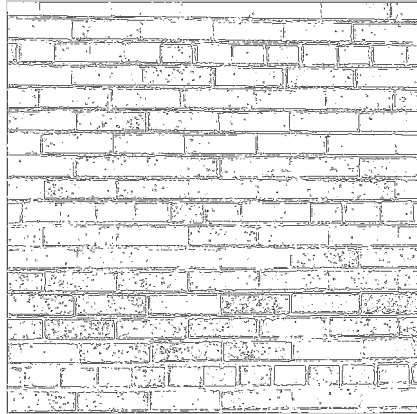


(A)



(B)

Figure.28 : Résultat de segmentation avec l'algorithme $C_a \cdot \text{var}(b)$

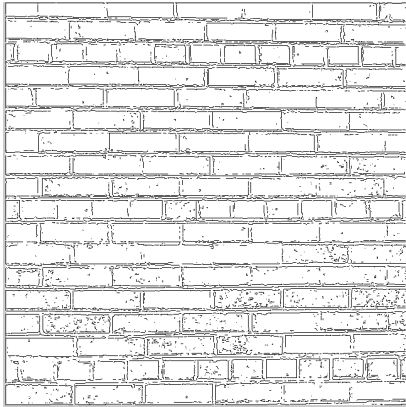


(A)

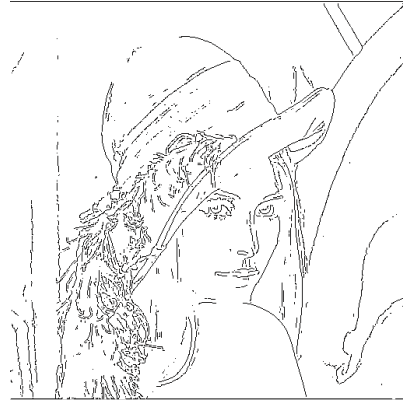


(B)

Figure.29 : Résultat de segmentation avec l'algorithme R_a



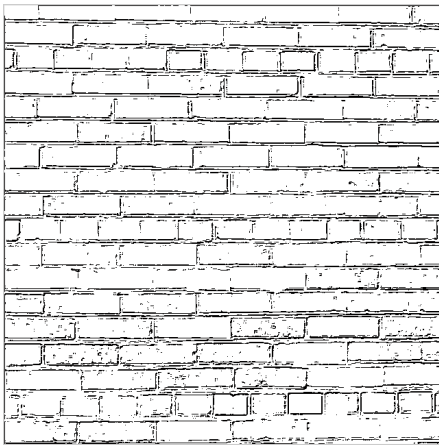
(A)



(B)

Figure.30 : Résultat de segmentation avec l'algorithme $R-a*var(b)$

IV-2-1-b En utilisant une fenêtre 5x5

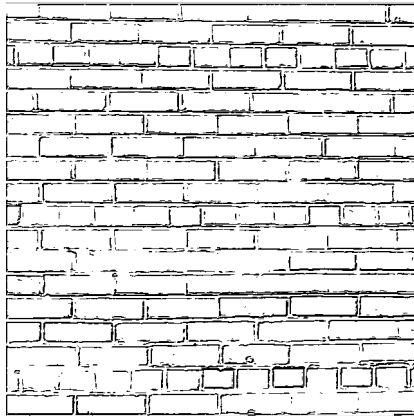


(A)



(B)

Figure.31 : Résultat de segmentation avec l'algorithme C_a

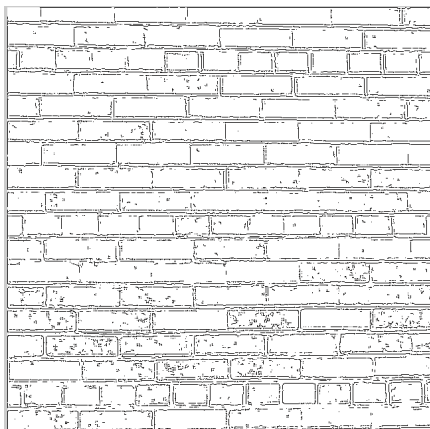


(A)



(B)

Figure.31 : Résultat de segmentation avec l'algorithme $C_a \cdot \text{var}(b)$

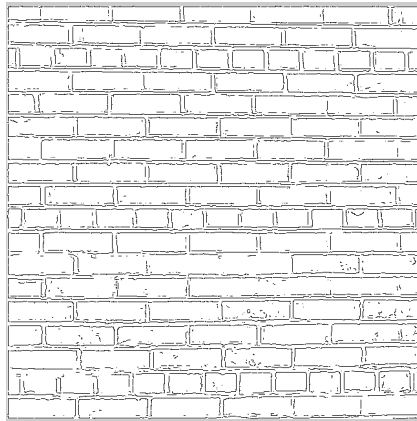


(A)



(B)

Figure.32 : Résultat de segmentation avec l'algorithme R_a



(A)



(B)

Figure.33 : Résultat de segmentation avec l'algorithme $R_a \cdot \text{var}(b)$

IV-2-2 Valeurs du facteur P :

Image \ algorithme	C_a	R-a	C-a*var(b)	R-a*var(b)
Image MUR	1	4	1.5	1
Image LENNA	1	4.5	0.6	0.5

Tableau.1 : Valeur de P utilisée pour le choix du seuil global
Des images tests

IV-3 Interprétation des résultats :

Les résultats obtenus en utilisant les algorithmes C_a et R_a avec une fenêtre 3x3 (voir figure.27 et figure.29) donnent plus de détails par rapport aux deux autres algorithmes (algorithmes $C_a \cdot \text{Var}(b)$ et $R_a \cdot \text{Var}(b)$), voir figure.28 et figure.30. Nous remarquons à titre d'exemple que les contours du chapeau de LENNA sont mieux reproduits. Cependant, une confusion entre bruit et point de contours se remarque et cela est dû au fait que le paramètre variance, pris en considération dans le deuxième cas d'algorithme, est caractérisé par sa robustesse au bruit. De plus, nous remarquons que la taille de la fenêtre joue un rôle très important dans les résultats obtenus et est fonction de l'application visée. Comme nous avons utilisé l'algorithme de Fisher qui est basé principalement sur deux types de seuils à savoir un

seuil global et un seuil local. Le seuil global dans nos algorithmes(classification et rangs), se fait de manière heuristique et s'ajuste en fonction de l'image utilisée. Un compromis doit se faire entre le choix de ce seuil, la taille de la fenêtre et le résultat obtenu.

L'algorithme $R_a \cdot \text{var}(b)$ donne de meilleurs résultats par rapport aux trois autres. Les contours obtenus avec cet algorithme sont fins et il y a moins de bruits.

Donc, on peut dire que la variance de b aide dans l'élimination du bruit, et le renforcement des transitions des niveaux de gris.

Pour la localisation de contour, la méthode des rangs de transition donne de meilleurs résultats. En effet, la plus parts des contours obtenus avec cette méthode sont de largeur d'un pixel et la localisation est précise.

En utilisant une fenêtre de taille 5x5 pour le calcul de a et b , nous remarquons que les résultats obtenus affichent moins de bruits, et donc donne une image plus claire.

D'après les résultats obtenus à partir des tests précédant, nous concluons que l'algorithme $R_a \cdot \text{var}(b)$ en utilisant une fenêtre de taille 5x5, est le plus efficace vu qu'il donne des contours fin et garantit une bonne localisation, nous l'avons donc retenu pour les tests sur l'image médicale.

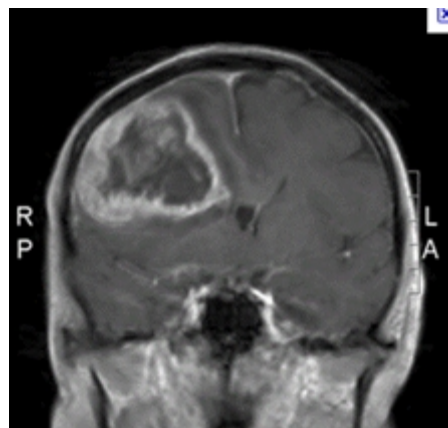
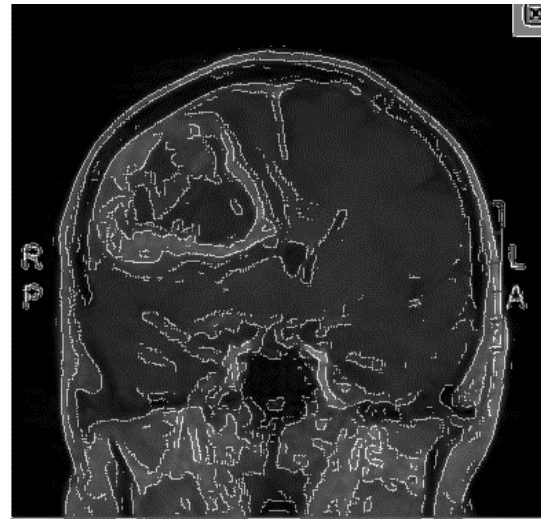


Figure.34 : (A) Image originale



(B) Image des contours



(C) Résultats de superposition

Figure.35 : Résultats de la segmentation sur une image médicale

La figure.35 (B) montre les contours obtenus avec notre méthode ($R_a * \text{var}(b)$) en utilisant une fenêtre de taille 5x5 et un facteur $P = 0.1$, et la figure (C) montre les contours obtenus insérés dans l'image originale.

Discussions :

La méthode élaborée est efficace et nous donne de bons résultats. On voit bien que les contours sont correctement détectés sur les transitions de niveaux de gris entre les régions. Et on arrive bien à distinguer la région atteinte des régions saines.

Les contraintes de cette méthode sont :

- le choix de la taille de la fenêtre.
- le choix du seuil global.
- le temps d'exécution des algorithmes qui s'avère un peu lent.

La taille de la fenêtre a une grande importance : si elle est élevée, la segmentation compte moins de détails mais il y'a moins de bruits et inversement.

Le choix du seuil global est un problème assez délicat. La plupart des méthodes choisissent ce seuil de façon heuristique. En effet, ce seuil dépend toujours du contenu de l'image. Par exemple, pour les images contenant des objets lissés, on peut choisir un seuil plus faible. Par contre, pour une image contenant des objets bruités, on doit choisir un seuil plus grand. Dans notre étude, on calcule d'abord la moyenne des transitions des niveaux de gris (a_{ij} ou $a_{ij} * \text{Var}(b_{ij})$) de tous les pixels de l'image, puis on choisit le seuil en appliquant à la moyenne des transitions de niveaux de gris un facteur **P**.

Le temps de calcul s'avère lent (4 mn) et est fonction des fonctions insérées.

CONCLUSION

Conclusion :

Dans ce mémoire, nous nous sommes intéressés à la segmentation d'image par modélisation basée sur les niveaux de gris ordonnés. Cette segmentation est appliquée aussi bien à des images tests tirées de l'album de Brodatz qu'aux images IRM, notamment en présence d'anomalie, en espérant aider les médecins à mieux comprendre le contenu des IRM et diminuer les risques d'erreurs de diagnostic.

Ce travail a consisté d'une part, à faire une recherche bibliographique dans le domaine du traitement d'image en général et d'autre part à mettre en œuvre une méthode de segmentation que l'on a présentée dans le chapitre III.

Les résultats expérimentaux obtenus avec la méthode élaborée sont très satisfaisants vu que la méthode des rangs de transition garantit des contours fins (de la largeur d'un pixel), et une localisation exacte du point contour. Par contre, la méthode de classification ne garantit pas des contours assez fins. C'est la raison pour laquelle nous avons opté pour la 1^{ère} méthode que nous avons appliquée aux images médicales.

Nous estimons que ce travail pourra servir d'aide pour d'autres applications dans le domaine de traitement d'images. Cependant, d'autres améliorations peuvent être opérées sur ces algorithmes et notamment automatiser le choix du seuil global afin de simplifier l'application.

BIBLIOGRAPHIE

- [1] Laurent Busin, Segmentation d'images couleur par analyse récursive d'histogrammes couleur monodimensionnels, Rapport de stage, USTL, 2003.
- [2] Sidi Mohamed Farssi, Segmentation d'images par détection de contours, de régions ou par morphologie mathématique, Etude Comparative et Evaluation des performances, thèse de doctorat, UCAD Dakar, 1997.
- [3] Layazid.K, Ould Amar.J : Segmentation d'images médicales par la méthode des histogrammes de la somme et la différence des niveaux de gris, mémoire d'ingénieur d'état UMMTO, institut d'électronique, 2009.
- [4] Sikal.H, Amara.K : segmentation d'image médicale par application de la matrice de cooccurrence, mémoire d'ingénieur d'état UMMTO, institut d'électronique, 2010.
- [5] Guy Almouzni, Traitement numérique des images, ESTI, 2010.
- [6] Cours de traitement d'image, Catherine Achard.
- [7] Frederic Bouchara & al, Segmentation d'image : Application aux documents anciens, Mémoire master, LSIS, 2007.
- [8] Vision par Ordinateur : Outil fondamentaux, Ed Hermès, Radu Horaud et Olivier Manga.
- [8] Feature Extracting & Image processing, second edition, Marc Nixon & Alberto Aguado, Ed Newnes, 2002.
- [9] Traitement d'Image et Vision Artificielle, ENSTA, Jean-Christophe Baillie, 2003.
- [10] Cours de traitement d'image, Yaovi Gagou, Université de Picardis,
- [11] Classification automatique pour la segmentation de signaux unidimensionnels, Article, Samé - P Aknin - G Govaert, CRR - IRNTS.

- [12] Segmentation d'images : Application aux Documents Anciens, thèse master, Thibault Lelore, 2007.
- [13] M Bergounioux : Histogramme.pdf, Méthodes mathématiques pour le traitement d'image, Master2-2008-2009.
- [14] W.D.Fisher: On grouping for maximum homogeneity, JASA 53, 1958,789-798.
- [15] Hralick.R.H: Statical and structural approach to texture, preceding of IEEE vol 67,N°5, p786-804, 1979.
- [16] Canny.J.F: Fiding edges and lines image, In MIT AI TR,1983.
- [17] Deriche.R: Optimal edge detection using recursive filters, In proc, international conference on computer vision (ICCV'87), London, United Kingdom,8-12,1987.
- [18] Bellet.F: Une approche incrémentale a base de processus coopératifs et adaptatifs pour la segmentation des images en niveaux de gris, thèse de doctorat, INP de Grenoble, 1998.