



N° d'ordre :

UNIVERSITE DE M'SILA
FACULTE DES MATHÉMATIQUES ET DE L'INFORMATIQUE
Département d'Informatique

MEMOIRE de fin d'étude
Présenté pour l'obtention du diplôme de MASTER
Domaine : Mathématiques et Informatique
Filière : Informatique
Spécialité : Systèmes d'Informations Avancés
Par: ABDI NADIA

SUJET

**Segmentation Floue et Possibiliste d'images
par Fuzzy C-Means (FCM) et Possibilistic C-Means (PCM)**

Soutenu publiquement le : /06 /2013 devant le jury composé de :

.....	Université de M'sila	Président
Mme. O.Assas	M.A(A)	Université de M'sila Rapporteur
.....	Université de M'sila	Examineur
.....	Université de M'sila	Examineur

Promotion : 2012 /20 13

Dédicaces

Car à dieu voilà notre travail terminé et il est temps pour moi de partager ma joie avec tous ceux qui m'ont soutenu et encouragé.

Aux être les plus chers à mon cœur, ma mère et mon père, qui vous consacré votre vie à notre éducation et qui m'ont toujours tout donné : amour, affection, ... Je ne pourrai tout citer, ni trouver les mots pour les remercier, qui dieu me les garde inchaallah !

A mon encadreur Assas Warda, qui m'a beaucoup aidé et encouragé, je lui serai toujours reconnaissante d'avoir guidé mes !

A mes chères adorables sœurs, Assia, Saïda, Zoulikha, Fatma, Samira, Meriem, Halima, Sabrina, Itab, Chahinaze, Yakout, Sabah: mercie pour votre encouragements !

A mes frères !

A les mariés de mes chères sœurs : Laamouri, Lakhdar, Ahmed, Mouhammed !

A mon oncle Oumar !

A mes meilleurs amis : Hala, Souad, Naoual, Fatima, Zahra, Amal, Safa, Habiba, Dalal, Sabah, et pour tous mes amis ou qu'ils soient.

A l'ensemble de groupe d'UNJA surtout : Oussama, Rachid et Salim. l'ensemble des amis que j'ai connu pendant mes études et à ceux qui ont prodigués leurs vif conseil, encouragements et témoigné de leur amitié.

Nadia

Remerciement

*Je remercie avant tout premièrement Dieu pour la volonté, la santé et la
patience qu'il
m'a donné durant toutes les années d'étude et de pouvoir parachever ce
travail.*

*Mes remerciements les plus sincères sont adressés à mon encadreur
Prof. Assas Ouarda qui
m'a orienté d'une façon efficiente.*

*Je remercie les membres du jury qui ont accepté de juger ce travail.
Enfin, les derniers mots sont pour remercier toute ma famille et mes amis
pour leur aide et
Leurs encouragements constants et pour leur soutien moral.*

Liste des figures	IV
Liste des tableaux	V
Introduction générale	1

Chapitre 1 Traitement d'images

1.1. Introduction	2
1.2. Définition de l'image	2
1.3. Image numérique	3
1.4. Type des images	3
1.4.1. Les images vectorielles	3
1.4.2. Une image matricielle (ou bitmap)	4
1.5. Conception de base d'un Images bitmap	4
1.5.1. Le pixel et le niveau de gris.....	4
1.5.2. Echantillonnage et quantification	5
1.5.3. Le codage de pixel	5
1.5.4. La taille d'un images	5
1.5.5. Résolution d'une images	6
1.5.6. La luminance	6
1.5.7. Le bruit	6
1.5.8. Histogramme.....	7
1.6. Le codage des couleurs	7
1.7. Les différents formats d'images	10
1.8. La structure d'un fichier BMP	12
1.9. Quelques aspects du Traitement d'Image	13
1.9.1. Binarisation.....	13
1.9.2. Segmentation	14
1.9.3. Squelettisation	14
1.9.4. Convolution	15
1.9.5. Filtrage	16
a. Le filtre de Moyenne.....	16
b. Le filtre Médian	17
1.10. Morphologie mathématique	17
1.11. La detection des contours	17

1.12. Conclusion..... 19

Chapitre 2 Segmentation d'images

2.1. Introduction..... 21

2.2. Définition de la segmentation..... 21

2.3. Méthodes de la segmentation..... 22

2.3.1. Méthodes basées sur l'approches Région..... 23

2.3.1.1. Méthodes de type division-fusion "split-merge"..... 23

2.3.1.2. Méthodes de type croissance des régions..... 24

2.3.2. Méthodes basé sur l'approche Contours..... 25

2.3.3. Méthodes basé sur la classification..... 25

2.3.3.1 Méthode basé sur la classification supervisées..... 26

a. Méthodes bayésiennes..... 26

b. Réseaux de Neurones..... 27

c. Segmentation par les champs de Markov..... 28

2.3.3.2. Méthodes basé sur la classification non-Supervisées..... 29

a. L'algorithme K-Maens..... 29

b. L'algorithme Fuzzy C-Maens 30

c. L'algorithme Possibilités C-Maens..... 31

2.4. Exemples d'applications de la segmentation..... 33

2.5. Conclusion..... 33

Chapitre 3 Les algorithmes de la segmentation possibiliste

3.1. Introduction..... 34

3.2. La classification floue..... 34

3.2.1. La logique floue 34

3.2.2. La théorie des possibilités 35

3.2.2.1. La théorie des sous-ensembles flous..... 35

3.2.2.2. Approches floues et possibilistes en segmentation d'images..... 35

3.3. Algorithmes les C- moyennes floue FCM (Fuzzy C- Means)..... 35

3.3.1. Formulation du Problème 36

3.3.2. Les limites du FCM 38

3.4. Algorithme les C-moyennes possibiliste PCM(Possibilistic C-Means)..... 40

3.4.1. L'algorithme PCM 40

3.4.2. Nouvelle formule itérative.....	41
3.5. L'avantage et L'inconvénient algorithme possibiliste.....	43
3.6. Choix des paramètres de l'algorithme.....	44
3.6.1. Initialisation de l'algorithme.....	44
3.6.2. Choix des vecteurs forme	44
3.6.3. Choix de paramètre m	44
3.6.4. La métrique utilisée.....	44
3.7. Conclusion.....	45

Chapitre 4 Réalisation et Evaluation des Résultats

4.1. Introduction.....	46
4.2. Utilisation de l'environnement de développement Visual C#.....	46
4.3. Présentation et implémentation du système	47
4.4. Système de segmentation d'image.....	47
4.5. Critères de validité.....	48
4.5.1. Critère contraste de Rosenberger.....	48
4.5.2. Contraste de Zeboudj	49
4.5.3. Mesure de dissimilarité de Liu et Yang.....	50
4.5.4. Critère d'uniformité de mesure	50
4.6. Les résultats de segmentation d'images par FCM, PCM et FCM-PCM.....	51
4.6.1. Présentation des résultats obtenus par FCM	52
4.6.2. Présentation des résultats obtenus par PCM	53
4.6.2. Présentation des résultats obtenus par la coopération FCM-PCM.....	54
4.7. Evaluation des résultats obtenus par FCM, PCM et FCM-PCM.....	55
4.8. Discussion des résultats obtenus par FCM, PCM et FCM-PCM.....	55
4.9. Quelques interfaces de l'application	57
4.10. Conclusion	58
Conclusion générale et perspectives.....	59
Bibliographie.....	61

Fig.1.1. Représentation d'un image numérique dans un dans le plan cartésien.....	2
Fig.1.2. Images vectorielles	3
Fig.1.3. Images numérique.....	4
Fig.1.4. Images matricielles.....	4
Fig.1.5. Représentation de Echantillonnage et quantification.....	5
Fig.1.6. La résolution d'une image.....	6
Fig.1.7. L'histogramme et la palette associés à image.....	7
Fig.1.8. Exemples de squelettes pour des formes simples.....	15
Fig.1.9. Mise en œuvre du filtre Médian.....	17
Fig.1.10. De détection de contours (Sobel).....	18
Fig.2.1. Algorithme division/ fusion par les quadtree.....	21
Fig.3.1. Algorithme de FCM.....	38
Fig.3.2. L'influence des points ambigus sur les centres des classes.....	39
Fig.3.3. Algorithme de PCM.....	42
Fig.3.4. Algorithme de coopérative FCM et PCM.....	43
Fig.4.1. L'interface de Microsoft Visuel Studio 2010.....	47
Fig.4.2. Le processus de segmentation d'un image.....	47
Fig.4.3. L'interface segmentée.....	57
Fig.4.4. Le menu de Fichier.....	57
Fig.4.5. Le menu segmentation.....	58

Table 1.1. L'information couleur de pixel est codée sur 3 octets.....	8
Table 1.2. Les principaux modes de codage.....	9
Table 1.3. Exemples de poids d'images avec différentes profondeurs de codage.....	10
Table 1.4. Entête d'un fichier BMP.....	12
Table 1.5. Entête de l'image BMP.....	12
Table 1.6. Les différents types de masques de convolution.....	19
Table 4.1. Les résultats de segmentation d'images (a), (b), (c) et (d) par FCM.....	52
Table 4.2. Les résultats de segmentation d'images (a), (b), (c) et (d) par PCM.....	53
Table 4.3. Les résultats de segmentation d'images (a), (b), (c) et (d) par FCM et PCM....	54
Table 4.4. Evaluation tous les résultats de segmentation d'images.....	56

INTRODUCTION GENERALE

INTRODUCTION GENERALE

La segmentation constitue une étape importante du traitement d'une image. Cette opération a pour but de séparer différentes zones homogènes d'une image afin d'organiser les objets en groupes (clusters) dont les éléments ont en commun diverses propriétés (intensité, couleur, texture,..., etc).

Plusieurs algorithmes de segmentation d'images ont été développés ces dernières années chacun possédant ses propres avantages et ses limites d'utilisation. L'objectif de ce travail est étudié, implémenter et de comparer trois algorithmes de segmentation: le Fuzzy C-means, le Possibiliste C-means. Et nous indiquant ici les raisons qui nous ont amené à effectuer une algorithme de coopération qui combine entre l'aspect floue (FCM) et possibiliste (PCM). Cela a pour but de rendre l'algorithme de classification plus robuste face aux imprécisions et aux données aberrantes. Le PCM peut être utilisé dans un deuxième passage pour les points aberrants, après l'application d'un autre algorithme de regroupement, tel que les FCM, qui fournit une partition initiale pour les PCM. Ce dernier améliore cette partition obtenue suite à la première étape.

Le présent mémoire est composé de quatre chapitres sont organisées de la manière suivante :

- ✓ **Le premier chapitre** nous introduisons des notions générales sur le traitement d'images, en donnant la définition et les caractéristiques de l'image numérique, le format d'un fichier BMP et les filtres les plus utilisées pour l'amélioration de la qualité des images.
- ✓ **Le deuxième chapitre** sera dédié à la présentation de différent approche de la segmentation d'image.
- ✓ **Le chapitre trois** nous décrivons brièvement le principe de base des méthodes du segmentation floue et possibiliste à savoir l'algorithme C-moyennes floue (FCM) et l'algorithme de C-moyennes possibiliste (PCM).
- ✓ **Le quatrième chapitre** présente le travail réalisé ainsi que les méthodes d'évaluation de performance dédiées aux techniques de segmentation d'images. Après cela une partie de ce chapitre réservé à l'évaluation des performances des différents algorithmes implémentées.

Nous terminons par une conclusion et des perspectives.

CHAPITRE 1
TRAITEMENT D'IMAGES

1.1. Introduction

Aujourd'hui, l'image constitue l'un des moyens les plus importants qu'utilise l'homme pour communiquer avec autrui. C'est un moyen de communication universel dont la richesse du contenu permet aux êtres humains de tout âge et de toute culture de se comprendre. C'est aussi le moyen le plus efficace pour communiquer, chacun peut analyser l'image à sa manière, pour en dégager une impression et d'en extraire des informations précises.

On désigne par "technique de traitement d'images" toutes les techniques ayant pour but la modification des caractéristiques chromatiques des pixels des images bitmap.

Traitement d'images est souvent synonyme d'amélioration des images avec pour but d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes qu'on va utiliser dans différentes applications par exemple la reconnaissance, la segmentation.

Dans ce chapitre, on présente quelques principaux concepts de traitements d'images qui sont en relation avec notre sujet d'étude.

1.2. Définition de l'image

L'image est une représentation d'une personne ou d'un objet par la peinture, la sculpture, le dessin, la photographie, le film, etc. C'est aussi un ensemble structuré d'informations qui, après affichage sur l'écran, ont une signification pour l'œil humain.

Une image est une fonction de deux variables, par exemple $I(x,y)$ représentera une fonction d'amplitude (l'intensité) de deux variables réelles de position (x,y) dans le plan cartésien.

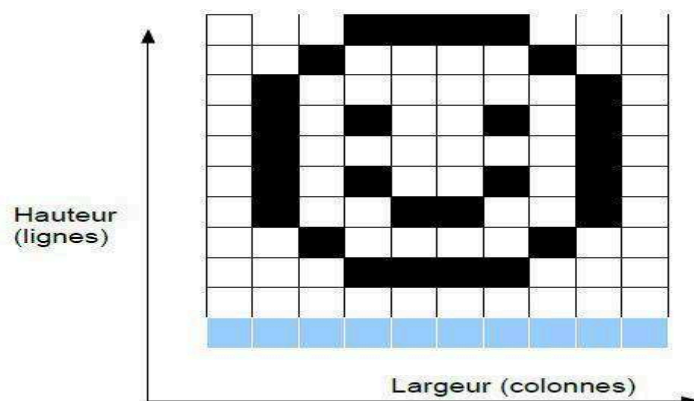


Figure1.1. Représentation d'un image numérique dans le plan cartésien.

1.3. Image numérique [29]

Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été acquise, traitée et sauvegardée sous une forme codée représentable par des nombres (valeurs numériques).

La numérisation est le processus qui permet de passer de l'état d'image physique (image optique par exemple) qui est caractérisée par l'aspect continu du signal qu'elle représente (une infinité de valeur dans l'intensité lumineuse par exemple), à l'état d'image numérique qui est caractérisée par l'aspect discret (l'intensité lumineuse ne peut prendre que des valeurs quantifiées en un nombre fini de points distincts). C'est cette forme numérique qui permet une exploitation ultérieure par des outils logiciels sur ordinateur.

1.4. Type des images

Il existe deux types d'images en informatique. Les images vectorielles et les images Bitmap.

1.4.1 Les images vectorielles

Les images vectorielles sont composées de formes géométriques qui vont pouvoir être décrites d'un point de vue mathématique. Par exemple une droite sera définie par 2 points, un cercle par un centre et un rayon. Le processeur est chargé de "traduire" ces formes en informations interprétables par la carte graphique (images Word, Publisher, CorelDraw-format WMF, CGM, etc.) (voir **Figure1.2**)



Figure1.2. Images vectorielles

1.4.2. Une image matricielle (ou bitmap)

Une image matricielle (ou bitmap) est une image constituée d'un ensemble de points: les pixels. Chaque point porte des informations de position et de couleur.

Les photos numériques et les images scannées sont de ce type (voir Figure1.2).

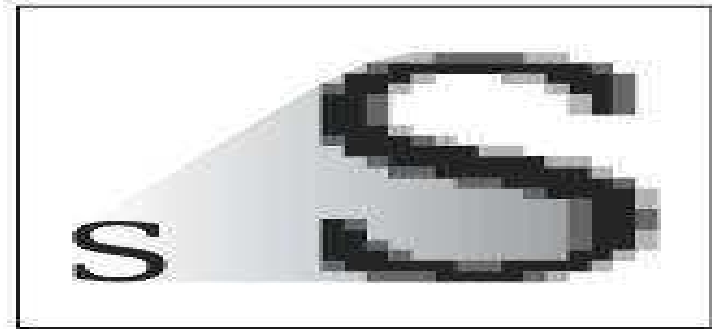


Figure1.3. Images matricielle.

1.5. Conception de base d'un Images bitmap

1.5.1. Le pixel et le niveau de gris

Le pixel représente le plus petit élément constitutif d'une image matricielle. Le mot pixel provient d'une abréviation de l'expression britannique PICTUREElement. La valeur numérique d'un pixel représente une intensité lumineuse.

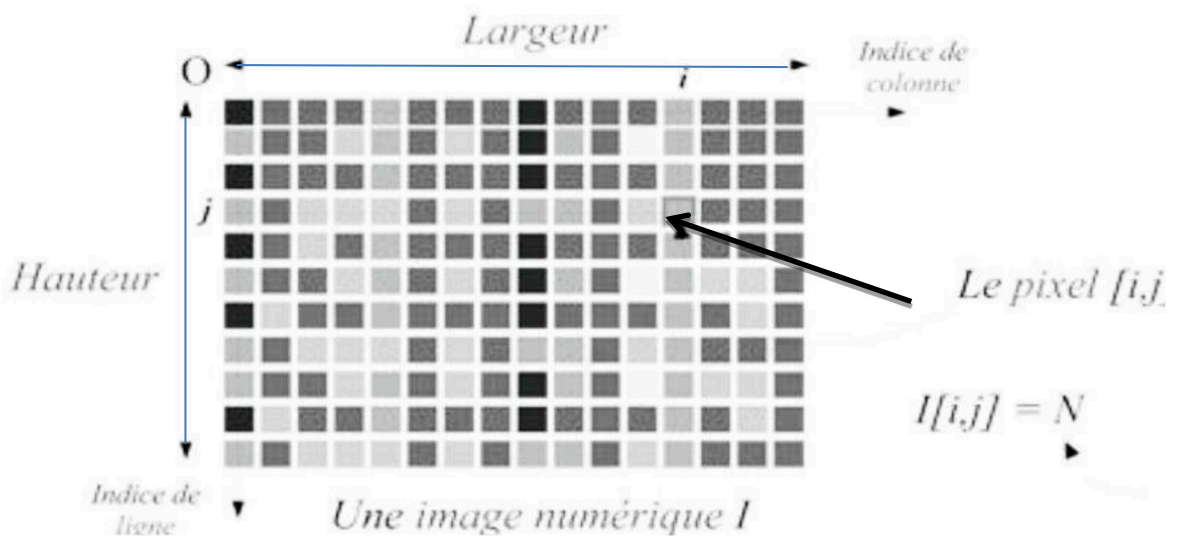


Figure1.4. Image numérique.

1.5.2. Echantillonnage et quantification [1]

L'**échantillonnage** est le procédé de discrétisation spatiale d'une image consistant à associer à chaque zone rectangulaire $R(x, y)$ d'une image continue une unique valeur $I(x, y)$. On parle de sous l'échantillonnage lorsque l'image est déjà discrétisée et qu'on diminue le nombre d'échantillons.

Une image numérique est une image échantillonnée et quantifiée. La **quantification** désigne la limitation du nombre de valeurs différentes que peut prendre $I(x, y)$.

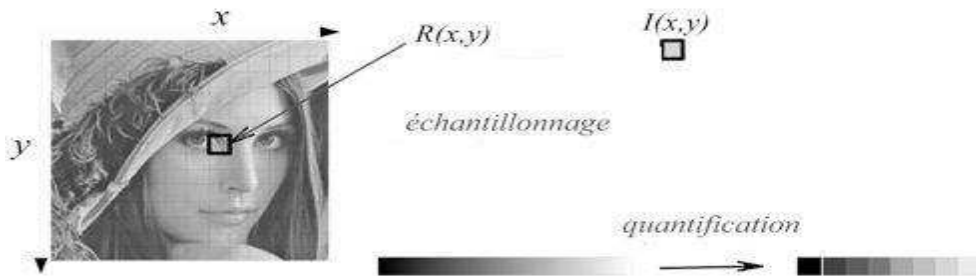


Figure1.5. Représentation de Echantillonnage et quantification

1.5.3. Le codage de pixel

Une image est divisée en points ou pixels. Considérons une image de 10 cm sur 10 cm avec une résolution très faible de 10 pixels par cm. Elle est codée sur $100 \times 100 = 10000$ pixels.

Avec une résolution de 100 pixels par cm (un pixel mesure 0,1 mm), elle est codée sur $1000 \times 1000 = 1000000$ pixels = 1 Million pixels.

Pratiquement toujours, la valeur d'un pixel est un mot binaire de longueur K bits, par conséquent un pixel peut prendre l'une des valeurs de l'intervalle $[0 \dots 2^K - 1]$. La valeur K est appelée profondeur de l'image.

L'ensemble de ces pixels est contenu dans un tableau à deux dimensions (une matrice) constituant l'image finalement obtenue.

1.5.4. La taille d'un images

La taille d'une image est le nombre de pixels de cette image, la taille d'une image qui est représenté par (328×456) dont 328 est le nombre de lignes, et 456 est le nombre de colonnes, est égale à: $328 \times 456 = 149568$ pixels.

1.5.5 Résolution d'une images:

La résolution d'une images est définie par le nombre de pixels par unité de longueur. Plus on a des points (ou pixels) par pouce (**ppp**) ou dote per inch (**dpi**) en anglais, meilleure est la qualité de l'mages.

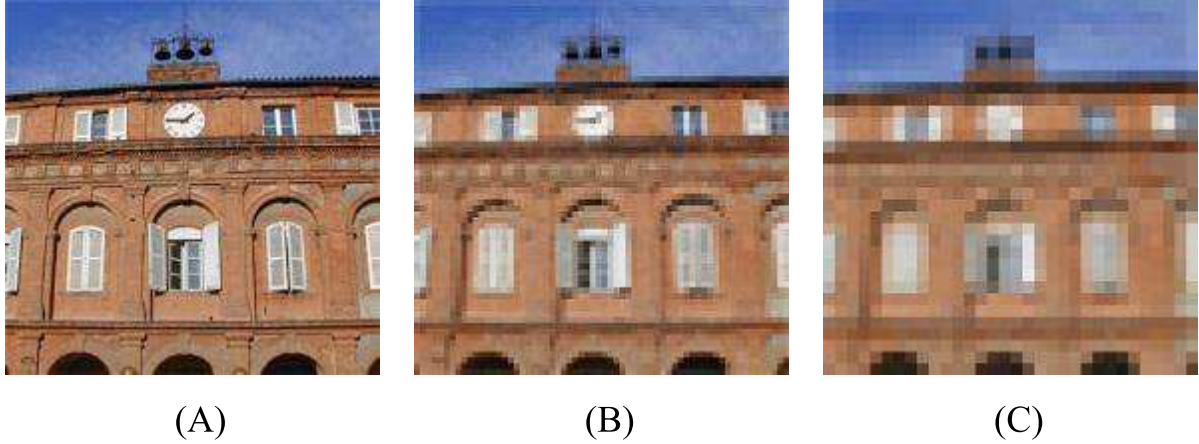


Figure1.6. La résolution d'une image (A) image acquise à 256 dpi, (B) image acquise à 64 dpi, (C) image acquise à 32 dpi.

1.5.6 La luminance [30]

Le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. La luminance est le degré de luminosité des pixels de l'image. Elle est définie aussi comme l'intensité d'une source étendue dans une direction donnée, divisée par l'aire apparente de cette source dans cette même direction

1.5.7. Le bruit

Un système d'enregistrement d'images ne restitue pas l'image de manière parfaite. En effet des informations parasites s'ajoutent de manière aléatoire aux détails de la scène d'origine, c'est ce qu'on appelle: **bruit**.

Le bruit n'a pas les mêmes origines, mais provoque des effets assez semblables comme la perte de netteté sur les détails ou l'apparition de grains.

1.5.8. Histogramme [31]

Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse. Par convention un histogramme représente le niveau d'intensité en abscisse en allant du plus foncé (à gauche) au plus clair (à droite).

Ainsi, l'historgramme d'une image en 256 niveaux de gris sera représenté par un graphique possédant 256 valeurs en abscisses, et le nombre de pixels de l'image en ordonnées. Prenons par exemple l'image suivante composée de niveaux de gris:

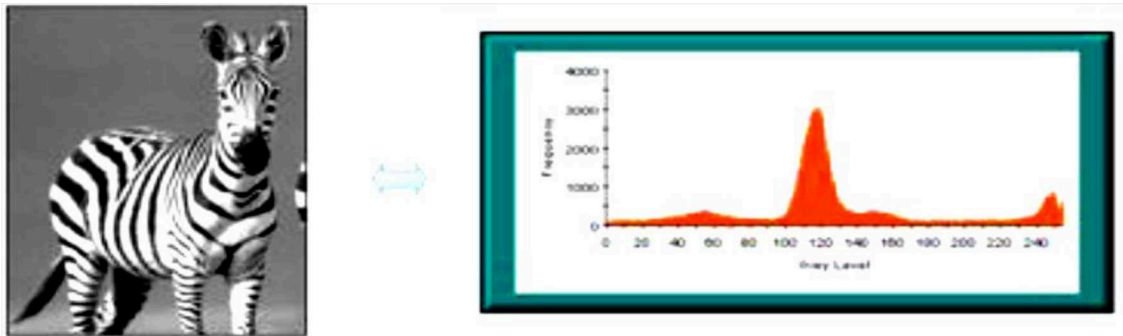


Figure1.7. L'historgramme et la palette associés à cette image.

Pour les images en couleur plusieurs histogrammes sont nécessaires. Par exemple pour une image codée en RGB:

- un histogramme représentant la distribution de la luminance,
- trois histogrammes représentant respectivement la distribution des valeurs respectives des composantes rouges, bleues et vertes.

1.6. Le codage des couleurs [2]

Une image est représentée par un tableau à deux dimensions dont chaque case est un pixel. Pour représenter informatiquement une image monocanal, il suffit donc de créer un tableau de pixels dont chaque case contient une valeur. La valeur stockée dans une case est codée sur un certain nombre de bits déterminant la couleur ou l'intensité du pixel, on l'appelle profondeur de codage (parfois profondeur de couleur). Alors la profondeur d' image représente le nombre de bits associés à chacun des pixels d'une images afin de coder les couleurs (Grand dictionnaire Terminologique). Il existe plusieurs standards de codage de la profondeur:

Cette information de couleur va dépendre du nombre de valeurs numérique que l'on veut attribuer à chaque pixel voir (Table 1.1).

Pour l'images les plus simples (noir et blanc): Le stockage d'un bit dans chaque cas permet de définir 2 couleurs (noir ou blanc).

- **Les images 16 couleurs ou 16 niveaux de gris.** En stockant 4 bits dans chaque case, il est possible de définir pour chaque pixel 16 dégradés de gris, du noir au blanc, ou bien 16 couleurs différentes.
- **Les images 256 couleurs ou 256 niveaux de gris.** En stockant un octet dans chaque case, il est possible de définir 256 dégradés de gris allant du noir au blanc ou bien 256 couleurs différentes.
- **Les images en couleurs:** Un pixel doit pouvoir prendre beaucoup plus de valeurs différentes. La valeurs de chacun d'entre-deux est habituellement enregistrée sous forme de trois nombres binaires à 8 bits (d'une valeur de 0 à 255), correspondant respectivement à chacune des couleurs primaires: **Rouge, Vert et Bleu (RVB)**. Cela donne une gamme de $256 \times 256 \times 256$ couleurs possibles (soit 16777216 couleurs). On obtient une image de couleur dit à 24 bits ou encore en couleurs variés.


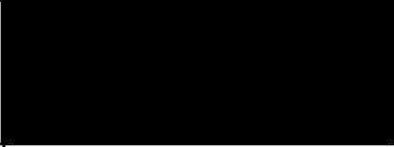


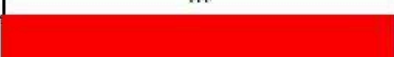


Valeur R	Valeur V	Valeur B	Couleur correspondante	Commentaire
0	0	0		noir
0	0	1		un peu moins noir (nuance impossible de détecter à l'oeil par rapport au noir)
...
0	0	255		bleu
...
0	255	0		vert
...
255	0	0		rouge
...
128	128	128		couleur intermédiaire correspondant à un gris
255	255	255		blanc

Table 1.1. L'information couleur de pixel est codée sur 3 octets.Ce qui entraine des fichiers volumineux.

Il existe plusieurs modèles de codage des couleurs:

- Le modèle RGB (ou RVB): système additif. Le modèle additif se prête aux couleurs qui seront restituées sur des périphériques de sortie tels que les moniteurs, les téléviseurs, les projecteurs.
- Le modèle CMYK (ou CMJN): système soustractif. Ce modèle est utilisé dans le domaine de l'impression.
- Le modèle HSL (ou TSL): teint – saturation (appelée aussi chrominance) – luminosité (parfois appelée luminance). Ce modèle se fonde sur la perception des couleurs par l'œil humain.

Type de codage	Nombre des bits par pixel	Valeurs possible	Interprétation de ces valeurs
Treint	1 bits	0 ou 1	0=noire, 1=blanc
Nuances de gris	8 bits	De 0 à 255	256 niveaux de gris, entre le noir (0) et le blanc (1)
Couleurs réelles (true color) RVB	24 bits (3×8)	3 valeurs de 0 à 255	Chaque valeurs représenter une intensité de rouge, de vert ou de bleu, 0=pas de couleur, 255=intensité maximale. On obtient ainsi 16 million de couleurs.
Couleurs indexées par une palette	8 bits	De 0 à 255	Une table appelée palette établit la correspondance entre ces 256 numéro de couleur et les couleurs réelles qu'ils représenter, codées dans le mode RVB

Table 1.2. Les principaux modes de codage.

Pour connaître le poids (en octets) d'une image, il est nécessaire de compter le nombre de pixels que contient l'image, cela revient à calculer le nombre de cases du tableau, soit la hauteur de celui-ci que multiplie sa largeur. Le poids de l'image est alors égal à son nombre de pixels que multiplie le poids de chacun de ces éléments [32].

Voici le calcul pour une image 640x480 en True color voire (**Table 1.3.**):

- Nombre de pixels: $640 \times 480 = 307200$
- Poids de chaque pixel: $24 \text{ bits} / 8 = 3 \text{ octets}$
- Le poids de l'image est ainsi égal à:

$$307200 \times 3 = 921600 \text{ octets} \text{ -----} 921600 / 1024 = 900 \text{ Ko}$$

Définition de l'image	Noire et Blanc	256 couleur (8bits)	65000 couleurs (16 bits)	True color (24 bits)
320×200	7.8 Ko	62.5 Ko	125 Ko	187.5 Ko
640×480	37.5 Ko	300 Ko	600 Ko	900 Ko
320×200	58.6 Ko	468.7 Ko	987.5 Ko	1.4 Mo
320×200	96 Ko	768 Ko	1.5 Ko	2.3 Mo

Table 1.3. Exemples de poids d'images avec différentes profondeurs de codage.

1.7. Les différents formats d'images

Un format d'image est une représentation informatique de l'image, incluant des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler.

La plupart des formats sont composés d'un en-tête contenant des attributs (dimension de l'image, type de codage, LUT, etc.), suivi des données (l'image proprement dite).

La structuration des attributs et des données diffère d'un format à un autre. Il existe plusieurs formats d'images, nous allons mentionner quelques-uns:

1.7.1. windows BitMaP (BMP)

Le format BMP est l'un des formats les plus simples. Il a été développé conjointement par Microsoft et IBM. Cette technologie a pour principal avantage la qualité des images fournies pas de compression (pas de perte de qualité). Cela fait de lui un format d'image très lourd, pas ou peu utilisé sur Internet.

1.7.2. Le JPEG (Joint Photographic Experts Group)

C'est le format le plus courant, on le rencontre sur Internet. Il occupe peu d'espace disque. C'est le format développé par les photographes pour transmettre des images de qualité photographique professionnelle. Il gère des millions de couleurs mais il ne possède pas de

palette de couleurs associée et donc les couleurs peuvent être différentes sur des machines et des systèmes différents.

1.7.3. GIF (Graphical Interchange Format)

Ce format est l'autre standard d'internet. Les fichiers GIF sont de petites tailles, ce qui est dû au fait que ces images ne peuvent enregistrer que 256 couleurs: le plus gros avantage du format est lié à son plus gros inconvénient. Le format GIF permet également la création d'animations et de détournement.

1.7.4. Le PNG (Portable Network Graphics)

Le format PNG reprend le principe de codage du format GIF mais n'est pas limité à 256 couleurs, et offre une compression généralement plus efficace. Il permet donc contrairement à GIF d'enregistrer des photographies sans perte de qualité, mais avec un gain d'espace de stockage moindre comparativement au format JPEG.

1.7.5. Le PSD (PhotoShop Document)

Le PSD est le format utilisé par le logiciel Adobe Photoshop. Sa particularité est qu'il utilise des calques qui, à la manière de feuilles transparentes superposées, permet de travailler sur l'une ou l'autre des couches sans apporter de modifications aux autres.

1.7.6. Le format d'image TIFF (Tagged Image File Format) [33]

Le TIF / TIFF est le format de fichier le plus couramment utilisé pour stocker des images, des photographies. TIF est plus couramment utilisé dans les environnements professionnels et pour l'impression commerciale. Le format TIF est le format le plus largement pris en charge sur toutes les plateformes. Il est le format standard pour les images de haute qualité. Bien que de grande taille, le format TIF est considéré comme étant le format le plus fiable pour des impressions de haute qualité comme pour le textile, les tissus, etc.

1.7.8. WMF (Window Meta File)

Le format Window Meta File est la version de Microsoft du format PICT de Macintosh et peut contenir des objets comme des lignes, des cercles et ainsi de suite. Les WMF ont une entête spéciale définie par Aldus Corp. Pour permettre aux applications de lire, mettre à l'échelle et dessiner les fichiers WMF.

1.8. La structure d'un fichier BMP [3]

Le fichier se décompose en 4 parties dans l'ordre:

- L'entête du fichier (FileHeader)
- L'entête du Bitmap (BitMapHeader)
- Une palette de couleur (pour les images d'au plus 256 couleurs soit 8 bits par pixels).
- Les données du BitMap.

1.8.1. En-tête du fichier [4]

L'entête est constitué d'une série d'entiers, codés sur 16 ou 32 bits (respectivement 2 ou 4 octets).

Ces entiers sont disposés dans un ordre précis:

Nom du champ	Longueur en octets	Signification
Identifiant	2 octets	Contient toujours l'octet 'B' suivi de l'octet 'M'.
FileSize	4 octets	Taille totale du fichier en octets.
Reserved	4 octets	Champ réservé, doit être égal à 0.
DataOffset	4 octets	Nombre d'octets séparer.

Tab.1.4. Entête d'un fichier BMP.

1.8.2. L'entête du Bitmap (BitMapHeader)

L'entête fournit des information sur l'images, comme le montre de tableau suivant:

Nom du champ	Longueur en octets	Signification
HeaderSize	4 octets	La taille de l'entête de l'image en octets
Width	4 octets	Largeur de l'image en pixels
Height	4 octets	Hauteur de l'image en pixels
Planes	2 octets	Le nombre de plans. Cette valeur vaut toujours 1
BitsPerPixels	2 octets	Nombre de bits nécessaires pour représenter un pixel
Compression	4 octets	Type de compression
BitmapDataSize	4 octets	Taille en octets des données de l'image
HResolution	4 octets	Résolution horizontale de l'image en pixels par mètre
VResolution	4 octets	Résolution verticale de l'image en pixels par mètre
Colors	4 octets	Nombre de couleurs dans l'image
ImportantColors	4 octets	Nombre de couleurs importantes

Table.1.5. Entête de l'image BMP.

1.8.3. La palette [3]

Les couleurs sont codées sur 4 octets: Bleu, Vert, Rouge et alpha (sur 8 octets, ordre inversé par rapport au RGB). La palette n'est présente que pour les images d'au plus 256 couleurs. En gamme de gris, les valeurs sont de 00 00 00 00 à FF FF FF 00 (ou 256 valeurs, de 0 à 255, pour le RGB). La palette a donc 256 fois 4 octets soit 1024 octets.

1.8.4. Le BitMap [3]

L'image est codée de la dernière ligne à la première, de gauche à droite. La largeur de l'image en octets est toujours un multiple de 4. Si ce n'est pas le cas, la fin de la ligne est complétée avec de zéros. Les pixels sont stockés bits à bits: Pour un 256 couleurs ou Niveau de gris, un octet décrit un pixel et un octet est représenté par un caractère ayant la valeur ascii du pixel.

1.8.5. Les données du BitMap

L'image est codée de la dernière ligne à la première, de gauche à droite. La largeur de l'image en octets est toujours un multiple de 4. Si ce n'est pas le cas, la fin de la ligne est complétée avec de zéros. Les pixels sont stockés bits à bits: Pour un 256 couleurs ou grisés, un octet décrit un pixel et un octet est représenté par un caractère ayant la valeur ascii du pixel.

1.9. Quelques aspects du Traitement d'Image

1.9.1. Binarisation [5]

La binarisation (le seuillage) est la technique de classification la plus simple où les pixels de l'image sont partagés par un seul seuil s en deux classes: ceux qui appartiennent au fond et ceux qui appartiennent à la scène (l'objet). L'image est alors séparée en deux classes de façon à ce que l'information comprise entre 0 et s est retenue et l'autre non, ou vice-versa.

Soit l'image I ($M \times N$), supposons que $f(x, y)$ représente le niveau de gris du pixel aux coordonnées (x, y) , $0 \leq x \leq M$, $0 \leq y \leq N$ et s est le seuil choisi, les pixels de l'objet sont ceux ayant le niveau de gris inférieur à s et les autres ayant le niveau de gris supérieur à s sont des pixel du fond. Alors, l'image binarisée G est déterminée par les pixels (x, y) dont la valeur est:

$$g(x, y) = \begin{cases} 1 & \text{Si } f(x, y) > S \\ 0 & \text{Si } f(x, y) \leq S \end{cases} \quad 1.1$$

Il existe trois grandes techniques de sélection du seuil s : global, local et dynamique. Comme il y a des différentes façons de déterminer le seuil s , il peut être considéré comme une fonction sous forme de $s = t ((x, y), p (x, y), f (x, y))$ où $p (x, y)$ représente des propriétés locales du point (x, y) . Si s ne dépend que de la valeur $f (x, y)$ du point, le seuil est global, s'il dépend en plus de $p (x, y)$, s est un seuil local. Et si s dépend à la fois de (x, y) , de $p (x, y)$ et de $f (x, y)$, on dit le seuil dynamique ou bien adaptatif.

Dans la méthode de binarisation **globale** un seuil unique est calculé à partir d'une mesure globale sur toute l'image. Il nous permet de décider l'appartenance d'un pixel à l'objet ou au fond.

Pour la binarisation **locale**, la classification d'un pixel dépend non seulement du pixel soi-même mais aussi de ses informations locales.

1.9.2. Segmentation [6]

La segmentation d'images ainsi définie est un domaine vaste où l'on retrouve de très nombreuses approches.

- La détection de frontières.
- La recherche de régions.
- L'approche de la classification.

Toutes ces approches visent à l'extraction des indices visuels. Après de nombreuses années passées à rechercher la méthode optimale, les chercheurs ont compris que la segmentation idéale n'existait pas. On peut même montrer que le problème de la segmentation est le plus souvent un problème mal posé. Etant donnée une image, il existe toujours plusieurs segmentations possibles. Une bonne méthode de segmentation sera donc celle qui permettra d'arriver à une bonne interprétation. Elle devra donc avoir simplifié l'image sans pour autant en avoir trop réduit le contenu.

1.9.3. Squelettisation [7]

La squelettisation donne une représentation compacte des objets, même dans le cas de formes et de topologies complexes. En dimension 2, les squelettes représentent des lignes inter connectées au centre d'un objet, comme le montre la figure (**Figure1.8.**).

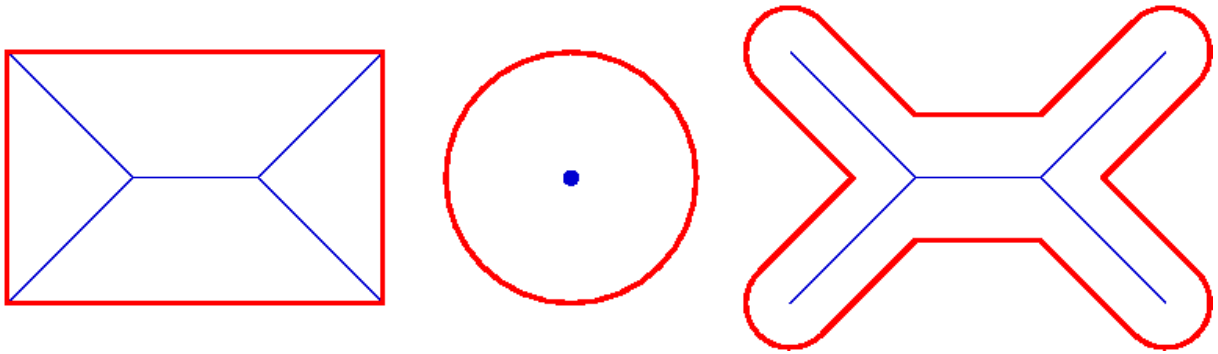


Figure 1.8. Exemples de squelettes pour des formes simples.

Ils sont très utilisés en analyse d'image et reconnaissance de forme, car ils permettent de décrire synthétiquement non seulement la forme, mais aussi certaines propriétés mathématiques des objets, comme par exemple la longueur ou la surface. Ils sont par ailleurs utilisés en compression d'image car ils décrivent les objets de façon compacte.

1.9.4. Convolution [8]

La convolution est le remplacement de la valeur d'un pixel par une combinaison de ses avoisinants. Elle consiste à faire balayer une fenêtre d'analyse (masque) sur l'ensemble des pixels de l'image.

L'opération de convolution est calculée en tout point de l'image source en 3 étapes:

1. On centre le masque (par exemple un masque carré 3*3) sur le pixel courant.
2. On calcule les 9 produits entre la valeur de l'image et la valeur du masque superposé.
3. Puis on somme les 9 produits pour obtenir la valeur du pixel de l'image filtrée.

On peut résumer l'opération de convolution par la formule suivante:

$$S(i, j) = \sum_{k=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} \sum_{l=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} E_{i+k, j+l} h_{k,l} \quad 1.2$$

Où E est l'image d'entrée, h le masque de convolution avec h (0, 0) le centre du masque et N la taille du filtre (dans notre exemple N=3).

1.9.5. Filtrage [35]

Le principe du filtrage est de modifier la valeur des pixels d'une image, généralement dans le but d'améliorer son aspect. En pratique, il s'agit de créer une nouvelle image en se servant des valeurs des pixels de l'image d'origine.

a. Le filtre de Moyenne

Le filtrage des bruits blancs conduit à des filtres de moyenne (ou de lissage) qui remplacent le pixel courant par la valeur moyenne des pixels sur une fenêtre 3x3 ou 5x5.

Ce filtre est bien compatible avec notre observation de départ selon laquelle le bruit est caractérisé par d'importantes hautes fréquences. Mais certaines fréquences (assimilées à de fortes variations dans les niveaux de gris) sont définitivement détruites, le lissage agissant comme un filtre passe-bas. L'image est non seulement lissée mais elle est aussi fortement altérée, les contours étant particulièrement dégradés par les filtres de moyenne.

Le filtre de moyenne est une version d'une gamme plus vaste de filtres de lissages dits linéaires.

Ici, le masque représentant ce filtre est le suivant:

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

Ce filtre, normalisé par un facteur 1/9 pour le ramener dans les valeurs normales, génère un lissage fort. Un lissage moyen peut être réalisé par le biais du filtre suivant:

$$\begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix}$$

Le facteur de normalisation est sans surprise 1/16 et le coefficient plus élevé du pixel central permet d'atténuer l'effet de lissage.

b. Le filtre Médian

Les inconvénient du filtre de Moyenne induisent une orientation vers le filtre dit Médian, qui préserve les contours. Ce filtre vise à remplacer la valeur du pixel central par la valeur médiane de la répartition (on trie les luminances dans l'ordre croissant) des niveaux de gris des pixels situés à l'intérieur de cette fenêtre (figure 3.8)

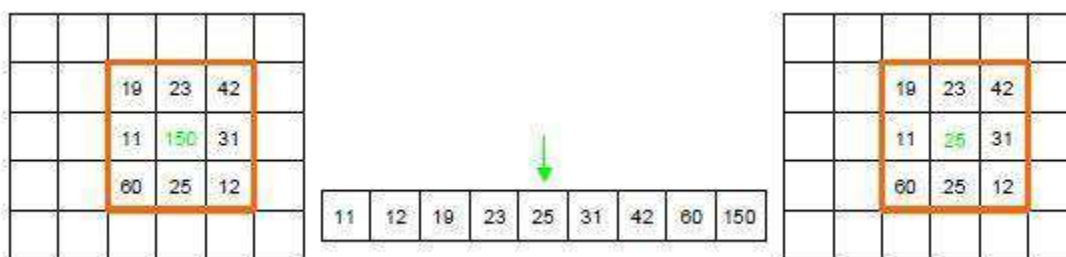


Figure1.9. Mise en œuvre du filtre Médian.

Ce filtre - non linéaire - permet d'éliminer des pixels isolés ou les traînées fines noyées dans des zones assez homogènes.

1.10. Morphologie mathématique [35]

La morphologie mathématique est une théorie d'analyse d'images née dans les années soixante lorsque Matheron et Serra effectuèrent des travaux sur les milieux poreux. A la base, cette technique s'appliquait à des images binaires, qui étaient considérées comme des ensembles.

Parmi les outils les plus importants de la morphologie mathématique on cite: érosion et dilatation.

- **Érosion:** qui consiste à enlever un pixel sur les pourtours de la forme, agrandir les trous.
- **Dilatation:** consiste à ajouter un pixel sur les pourtours de la forme, à boucher les trous.

1.11. La détection des contours [36]

Le but de la détection de contours est de repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. Ces changements de propriétés de l'image traduisent en général des événements importants ou des changements

dans les propriétés du monde. Ils incluent des discontinuités dans la profondeur, dans l'orientation d'une surface, dans les propriétés d'un matériau et dans l'éclairage d'une scène. La détection de contour est un champ de la recherche qui appartient au traitement d'image et à la vision par ordinateur, particulièrement dans le domaine de l'extraction de caractéristiques.

La détection des contours d'une image réduit de manière significative la quantité de données et élimine les informations qu'on peut juger moins pertinentes, tout en préservant les propriétés structurelles importantes de l'image. Il existe un grand nombre de méthodes de détection de l'image mais la plupart d'entre elles peuvent être regroupées en deux catégories. La première recherche les extremums de la dérivée première, en général les maximums locaux de l'intensité du gradient. La seconde recherche les annulations de la dérivée seconde, en général les annulations du laplacien ou d'une expression différentielle non-linéaire. Où laplacien est l'opérateur différentiel défini par l'application de l'opérateur gradient suivie de l'application de l'opérateur divergence.

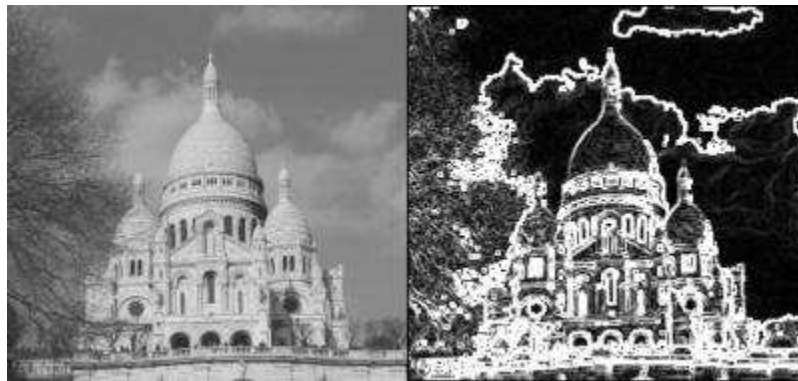


Figure1.10. De détection de contours (Sobel).

Le tableau (**Table 1.6**) illustre les différents types de masques de convolution utilisables pour l'obtention du contour d'une image.

Type de masque	Gradient partiel	Extraction de l'amplitude	Extraction de la direction																								
Masque de Roberts <table border="1"> <tr><td colspan="2">Gradient selon x</td></tr> <tr><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td></tr> </table> <table border="1"> <tr><td colspan="2">Gradient selon y</td></tr> <tr><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	Gradient selon x		-1	0	0	1	Gradient selon y		0	-1	1	0	$G_x G_y$ Substitution du pixel supérieur gauche	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \frac{\pi}{4} + \arctan\left(\frac{G_x}{G_y}\right)$												
Gradient selon x																											
-1	0																										
0	1																										
Gradient selon y																											
0	-1																										
1	0																										
Masque de Sobel <table border="1"> <tr><td colspan="3">Gradient selon x</td></tr> <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> <table border="1"> <tr><td colspan="3">Gradient selon y</td></tr> <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>	Gradient selon x			1	0	-1	2	0	-2	1	0	-1	Gradient selon y			1	2	1	0	0	0	-1	-2	-1	$G_x G_y$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_x}{G_y}\right)$
Gradient selon x																											
1	0	-1																									
2	0	-2																									
1	0	-1																									
Gradient selon y																											
1	2	1																									
0	0	0																									
-1	-2	-1																									
Masque de Prewitt <table border="1"> <tr><td colspan="3">Gradient selon x</td></tr> <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> <table border="1"> <tr><td colspan="3">Gradient selon y</td></tr> <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>	Gradient selon x			1	0	-1	1	0	-1	1	0	-1	Gradient selon y			1	1	-1	0	0	0	-1	-1	-1	$G_x G_y$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_x}{G_y}\right)$
Gradient selon x																											
1	0	-1																									
1	0	-1																									
1	0	-1																									
Gradient selon y																											
1	1	-1																									
0	0	0																									
-1	-1	-1																									

Table 1.6. les différents types de masques de convolution utilisée pour la détection du contour d'une image.

1.12. Conclusion

Le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dont le but est de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations utiles.

Dans ce chapitre nous avons présenté les différents concepts liés à l'image numérique, on a parlé sur les types d'une image, ces caractéristiques et quelques formats de fichiers d'images, on a

expliqué aussi la structure d'un fichier bitmap et finalement nous avons présenté quelques traitements d'images usuels.

CHAPITRE 2
SEGMENTATION D'IMAGES

2. 1. Introduction

Une image est une représentation planaire d'une scène ou d'un objet situé en général dans un espace tridimensionnel. L'information élémentaire associée à chaque point de l'image est transcrite en niveau de gris ou en couleur. L'analyse pertinente de l'image en regard de l'application concernée, les traiter et les interpréter. La segmentation est un traitement de bas-niveau qui consiste à créer une partition de l'image en sous-ensembles appelés régions. Une région est un ensemble connexe de points images (pixel) ayant des propriétés communes (intensité, texture, . . .) qui les différencient des pixels des régions voisines. Les connaissances utilisées sont le plus souvent des domaines de l'image numérique et du traitement du signal, donc sémantiquement assez pauvres. Il n'y a pas de méthode unique de segmentation d'une image, le choix d'une technique est lié à plusieurs facteurs. Du fait de cette diversité, il est difficile de définir de manière absolu une « bonne » segmentation. La segmentation n'est pas une fin en soi, sa qualité est fonction des résultats obtenus par les traitements situés en aval qui utilisent les primitives extraites.

La segmentation est une des étapes critiques de l'analyse d'images qui conditionne la qualité des mesures effectuées ultérieurement. Elle permet d'isoler dans l'image les objets sur lesquels doit porter l'analyse, de séparer, dans une image, les régions d'intérêt du fond.

2. 2. Définition de la segmentation

La segmentation d'image est une opération de traitement d'images qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis. Les pixels sont ainsi regroupés en régions, qui constituent un pavage ou une partition de l'image [37].

La segmentation est un processus de synthèse, visant à extraire les caractéristiques géométriques des images en faisant abstraction des nuances de couleur, des reflets, des transparences, des ombres, des textures et autres éléments qui constituent l'incroyable richesse de l'information visuelle [9].

Formellement, Soit X le domaine de l'image et f la fonction qui associe à chaque pixel une valeur $f(x,y)$. Si nous définissons un prédicat P sur l'ensemble des parties de X , la segmentation de X est définie comme une partition de X en C sous-ensemble $\{R_1, \dots, R_c\}$ tels que:

1. $X = \sqcup_{i=1}^c R_i$

2. $\forall i \in \{1, \dots, c\}$ R_i est connexe

3. $\forall i \in \{1, \dots, c\}$ $P(R_i) = \text{vrai}$

4. $\forall i, j \in \{1, \dots, c\}^2$ R_i est adjacent à R_j et $i \neq j \Rightarrow P(R_i \cup R_j) = \text{faux}$ Ou \sqcup représente une union d'ensemble disjoints.

Le prédicat P est utilisé pour tester l'homogénéité des ensembles R_i . Ces sous-ensembles constituent les régions de l'image. Une segmentation de l'image est donc sa décomposition en un ensemble de régions homogènes, le critère d'homogénéité P restant à déterminer. La première condition implique que tout pixel de l'image appartienne à une région et une seule. Cela signifie que l'algorithme de segmentation ne doit pas se terminer avant d'avoir traité tous les points. La seconde condition implique que toute région doit être connexe. La connexité des régions étant induite par le voisinage défini sur l'image. La troisième condition implique que chaque région doit être homogène. Enfin, la quatrième condition est une condition de maximalité indiquant que la fusion de deux régions ne doit pas être homogène. Il est important de remarquer que le nombre n de régions formant la partition de l'image reste indéterminé. Il peut donc exister plusieurs segmentations possibles pour un prédicat P donné.

2. 3. Méthodes de la segmentation

Après de nombreuses années passées à rechercher la méthode optimale, les chercheurs ont compris que la segmentation idéale n'existait pas. On peut même montrer que le problème de la segmentation est le plus souvent un problème mal posé. Etant donnée une image, il existe toujours plusieurs segmentations possibles. Une bonne méthode de segmentation sera donc celle qui permettra d'arriver à une bonne interprétation. Elle devra donc avoir simplifié l'image sans pour autant en avoir trop réduit le contenu.

La segmentation d'images ainsi définie est un domaine vaste où l'on retrouve de très nombreuses méthodes de segmentation, que l'on peut regrouper en quatre principales classes:

- Segmentation fondée sur les régions (en anglais: region-based segmentation).
On y trouve par exemple: la croissance de région (en anglais: region-growing),
décomposition/fusion (en anglais: split and merge).
- Segmentation fondée sur les contours (en anglais: edge-based segmentation).
- Segmentation fondée sur classification ou le seuillage des pixels en fonction de leur intensité (en anglais: classification ou thresholding).

2. 3. 1. Méthodes basées sur l'approche Région

Les approches de segmentation régions visent à créer une partition de l'image en un ensemble de régions homogènes au sens d'un ou plusieurs critères. On cherche donc à rassembler les pixels par critères de similarité.

La segmentation d'une image vis à vis d'un critère d'homogénéité H (par exemple "tous les pixels sont du même niveau de gris" ou "la variation de niveau de gris n'excède pas n niveaux") peut s'exprimer de la manière suivante:

La segmentation d'une image I en regard du critère H est une partition de l'image I en n régions homogènes X_1, \dots, X_n telles que:

- $\bigcup_{i=1}^n X_i = I$.
- Pour tout i , X_i est connexe.
- Pour tout i , $H[X_i]$ est vrai.
- Pour tout couple (X_i, X_j) de régions voisines, $H[X_i, X_j]$ est faux.

Cette définition conduit à deux remarques très importantes. Tout d'abord, une segmentation dépend du critère employé. Le choix du critère est donc primordial. Ensuite, la décomposition obtenue n'est pas unique. Pour un critère donné, il existe plusieurs solutions

2. 3. 1. 1. Méthodes de type division-fusion "split-marge"

Le processus de segmentation est itératif et alterne deux phases: une phase de division de toutes les régions non homogènes et une phase de fusion de toutes les régions adjacentes de sorte que la région résultante respecte toujours le critère d'homogénéité. En d'autres termes, cette approche consiste à utiliser des propriétés globales pour diviser l'image en zones homogènes puis à fusionner certaines de ces régions grâce à des caractéristiques locales.

L'algorithme de division-fusion consiste, dans un premier temps, à découper l'image en zones de taille moyenne (le plus souvent en quatre quadrants), puis à parcourir chacune des régions en cherchant si elle vérifie un critère d'homogénéité. Les régions non homogènes

sont à leurs tours divisées en sous-blocs. Ce processus de découpage de régions non homogène est pour suivit jusqu'à obtenir que des régions homogènes. Une étape de fusion permet par la suite de regrouper les zones contigües correspondant à la même région i. e. si deux régions adjacentes peuvent être fusionnées pour former une zone homogène, elles le sont.

Ces méthodes font généralement appel à la théorie des graphes, ainsi elles peuvent être classées selon la structure du graphe utilisé. La figure (2. 1) illustre le principe de la segmentation basée sur les quadtree.

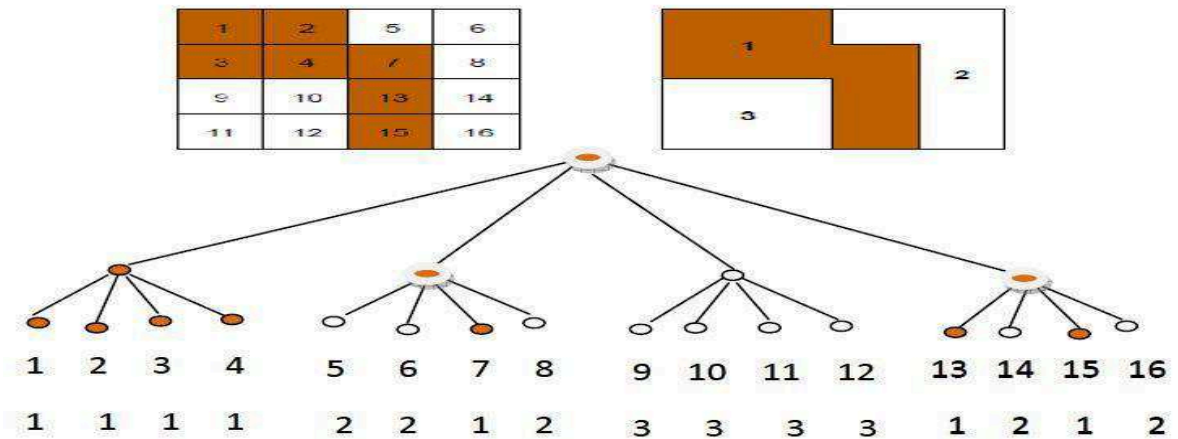


Figure 2. 1 Algorithme division/ fusion par les quadtree.

2. 3. 1. 2. Méthodes de type croissance des régions

Également appelées agrégation de pixels (les pixels forment la base du processus). Ces méthodes intègrent implicitement l'information spatiale dans le processus de segmentation. Les régions sont créées les unes après les autres (critère d'adjacence) avec pour chaque région, une phase d'initialisation et une phase itérative. La phase d'initialisation est la phase du choix d'un nouveau germe (point de départ d'une nouvelle région). La phase itérative est la phase d'agrégation des pixels voisins au germe, autrement dit, une région est construite à partir d'un germe par agrégation de pixels autour de ce germe selon un critère d'homogénéité. Ce germe doit être considéré comme une région, c'est-à-dire satisfaire au critère d'homogénéité (on peut choisir un pixel unique comme germe) [10], Un agrégat est un assemblage hétérogène de substances ou d'éléments qui adhèrent solidement entre eux.

Un prédicat classique d'homogénéité pour une région peut être par exemple la variance $\delta^2(R)$ des niveaux de gris de l'image associés aux points de cette région. Un point de l'image appartient à R si sa variance est inférieure à un seuil fixé. Le processus est initialisé par un

germe choisi par seuillage sur un attribut simple (comme le niveau de gris). Il s'arrête lorsqu'il n'y a plus de région qui vérifie le prédicat.

La qualité de la segmentation peut également être un critère de segmentation. Si l'on possède une référence permettant de construire un indicateur global de la qualité de la segmentation, cette mesure peut servir localement pour savoir si une fusion l'accroît significativement [11].

Une telle mesure peut être la probabilité de bonne interprétation des régions et des contours. Il est également possible d'évaluer la qualité de fusion en considérant une région comme homogène si aucune fusion locale ne peut l'améliorer. Ces méthodes peuvent intégrer des informations locales (histogramme local pour être plus robuste à la présence du bruit). L'agrégation de pixels donne d'assez bons résultats même si la localisation des contours reste moins précise que celle obtenue par les techniques de détection de contours. Notons aussi que les seuils d'agrégation influent beaucoup sur la qualité du résultat.

2. 3. 2. Méthodes basé sur l'approche Contours [12]

Contrairement aux approches régions, qui cherchent à former des zones homogènes, les approches contours se basent sur les discontinuités des images pour déterminer les contours des régions. On peut distinguer les méthodes dérivatives et les méthodes fondées sur

Les modèles déformables. Les méthodes dérivatives permettent de détecter de manière simple les ruptures dans une image ou un volume. Les contours (ou surfaces en 3D) sont assimilés à des points de forts gradients ou de dérivées secondes nulles.

2. 3. 3. Méthodes basé sur la classification

Cette méthode consiste à regrouper et à classer les pixels d'une image en classes en fonction de leurs propriétés. A chaque point de l'image est associé un vecteur d'attributs. La classification est alors effectuée sur ces vecteurs d'attributs de façon à aboutir à un nombre restreint de régions homogènes au sein de l'image.

Donc la classification est définie comme une procédure dans laquelle les pixels similaire d'une image sont identifiés et regroupés dans une même classe. Il existe deux grandes tendances: *La classification non supervisée* et *La classification supervisée*.

2.3.3.1. Méthodes basé sur la classification Supervisées

Les méthodes supervisées nécessitent un ensemble d'images ou de régions d'intérêt dans l'image déjà étiquetée. Soit, les données à analyser $X = \{X_j \in R^p, 1 \leq j \leq N\}$, où X_j est un vecteur forme représentant le pixel j . Cet ensemble de données est décomposé en deux sous-ensembles: un ensemble d'apprentissage et un ensemble de test. L'ensemble d'apprentissage est construit par étiquetage manuel des images ou des régions d'intérêt en C classes de tissus (C_1, \dots, C_i) par un ou plusieurs experts (radiologues). Chaque classe C_i . Se voit donc affecter un ensemble d'apprentissage E_i , et les données de l'ensemble de test sont segmentées en fonction des E_j . Parmi ces méthodes on peut citer: la segmentation Bayésienne, la segmentation par les champs de Markov, réseaux de neurones, segmentation bayésiennes, etc.

a. Méthodes bayésiennes

La segmentation bayésienne consiste à calculer, pour chaque vecteur forme X_j , la probabilité conditionnelle $P(C_i | X_j)$ pour chacune des C classes (C_1, \dots, C_C) à l'aide de la règle Bayes:

$$\forall i \in \{1..C\} P(C_i | X_j) = \frac{P(C_i) P\left(\frac{X_j}{C_i}\right)}{\sum_{k=1}^c P(C_k) \cdot P\left(\frac{X_j}{C_k}\right)} \quad (2.1)$$

La segmentation bayésienne admet:

- Soit une approche paramétrique, dans laquelle l'intensité d'un voxel (est un point à trois dimensions) est considérée comme la combinaison linéaire de probabilités d'appartenance attachées à chaque class (en effectuant l'hypothèse que les données sont conformes à des distributions paramétrique). La probabilité conditionnelle $P(X_j/C_i)$
- Est modélisée par une fonction dépendant d'un vecteur de paramètre θ . Le problème est alors pour chaque classe C_i d'estimer le meilleur paramètre θ connaissant l'ensemble d'apprentissage E_i de cette classe.
- Soit une approche non paramétrique, dans ce type d'approche, les probabilités conditionnelles $P(X_j/C_i)$ sont supposées quelconques. La forme non paramétrique doit permettre de rendre compte le plus fidèlement possible da la réelle distribution statistique des niveaux de gris dans l'image.

b. Réseaux de Neurones [13]

Un réseau de neurones est un réseau d'unité élémentaire (les nœuds) interconnectés, à fonction d'activation linéaires ou non linéaire. Ces nœuds sont regroupés pour les réseaux multicouches en sortie et éventuellement un ensemble de neurones cachés. De nombreux modèles de réseaux existent (réseaux de Hopfield, perceptrons multicouche, etc.), les différents nœuds étant complètement ou partiellement interconnectés aux autres. L'ensemble des liens convergeant vers un nœud constitue les connexions. Ceux qui divergent vers d'autres nœuds sont les connexions sortantes. A chaque connexion, entre des nœuds i et j , est associé un poids W_{ij} représentant la force de l'influence du nœud i sur le nœud j . L'ensemble des poids est regroupé dans un vecteur de poids synaptiques W . Un vecteur de scalaires a présenté à tous les nœuds d'entrée est appelé exemple. A cet exemple sont aussi associées les valeurs Y (le vecteur de sortie) que l'on désire apprendre. Les poids des connexions sont éventuellement modifiés au cours d'un cycle d'apprentissage.

Modifier la sortie des nœuds à partir de leurs entrées consiste tout d'abord à calculer l'activation présente à l'entrée du nœud puis à calculer la sortie du nœud suivant la fonction d'activation qu'elle possède. Un réseau de neurones peut ainsi être défini pour chaque nœud par quatre éléments:

- La nature de ses entrées, qui peuvent être binaires ou réelles.
- La fonction d'entrée totale e , qui définit le pré-traitement $e(a)$ effectué sur les entrées. Généralement, e est une combinaison linéaire des entrées pondérées par les poids synaptiques des connexions entrées pondérées par les poids synaptiques des connexions entrantes.
- La fonction d'activation f du nœud qui définit son état de sortie en fonction de la valeur de e . Toute fonction croissante et impaire convient et la fonction sigmoïde est souvent utilisée. La valeur de f en $e(a)$ est redirigée vers l'extérieure ou vers d'autres nœuds ou elle contribue à calculer leur état d'activation.
- La nature de ses sorties, qui peuvent être binaires ou réelles.

Les réseaux de neurones sont également utilisés pour obtenir une classification de l'image en tissus cérébraux. Ils sont organisés autour d'un ensemble de cellules (ou neurones) interconnectées. On dispose d'une base de connaissances constituée de couples (entrées, Sorties) et on utilise cette base pour entraîner une « mémoire » informatique à raisonner en prenant comme référence cette base empirique. Ainsi, en fonction de la base d'apprentissage, le réseau de neurones détermine la sortie pour chaque nouvelle entrée ; une mesure d'erreur

est calculée pour chaque sortie obtenue, on cherche donc la meilleure valeur de sortie (c'est elle qui minimise l'erreur). Ici encore, la base d'entraînement est d'images segmentées par un expert, ou d'un échantillon de l'image à segmenter.

La méthode de segmentation supervisée offre l'avantage d'être plus rapide et plus reproductible que les méthodes manuelles. Toutefois, elles ont le désavantage de rester très dépendantes de la base d'apprentissage. C'est pourquoi il est intéressant de développer des algorithmes entièrement automatiques «non supervisée», qui présenteront l'avantage d'offrir un résultat reproductible et indépendant des actions de l'opérateur.

c. segmentation par les champs de Markov [14]

Les champs de Markov sont largement utilisés en analyse d'images. Nous les avons utilisés dans la troisième étape pour régulariser notre image par l'introduction de l'information contextuelle. Les champs de Markov permettent de modéliser les propriétés globales en utilisant des contraintes locales.

Nous considérons une image (ou un volume) comme un ensemble S de n sites. L'ensemble S est défini par: $S = \{S_i, 1 \leq i \leq n\}$.

À chaque site S est associée une variable aléatoire X_s . L'image est alors considérée comme la réalisation d'un vecteur aléatoire $X = \{X_s, s \in S\}$, appelé champ des observables. La classification markovienne a alors pour objectif d'estimer le champ des étiquettes $Y = \{Y_s, s \in S\}$.

Les champs X et Y sont respectivement associés aux réalisations x et y définies par:

$$\mathbf{x} = \{\mathbf{x}_s, \mathbf{s} \in S\} \quad \text{et} \quad \mathbf{y} = \{\mathbf{y}_s, \mathbf{s} \in S\} \quad (2.2)$$

Si Y est un processus aléatoire sur un ensemble S , alors on dira que Y est un champ de Markov relativement au système de voisinage V si et seulement si: la probabilité conditionnelle locale en un site n est fonction que de la configuration du voisinage du site considéré.

Ce qui s'exprime de la façon suivante:

$$P(Y_s = y_s) > 0, \forall s \in S \quad (2.3)$$

$$P(Y_s = y_s | Y_t = y_t, t \neq s) = P(Y_s = y_s | Y_t = y_t, t \in V_s) \quad (2.4)$$

2. 3. 3. 2. Méthodes basé sur la classification non-Supervisées

L'intérêt des méthodes non supervisées est qu'elles ne nécessitent aucune base d'apprentissage et par là même aucune tâche préalable d'étiquetage manuel n'est requise. La seule intervention de l'expert se situe à la fin du processus pour identifier les tissus en comparant les classes calculées avec les classes biologiques.

Les algorithmes non supervisés les plus répandus tendent à minimiser une fonction coût, dépendant de la distance de chaque pixel aux prototypes (ou noyaux) des classes. Le prototype d'une classe étant un point connu dont l'appartenance à la classe est garantie et où chaque pixel est assigné à la classe qui lui est la plus proche. Selon la certitude de la classification que nous voulons obtenir, et la relation entre les classes, nous pourrions distinguer plusieurs méthodes de classification.

a. L'algorithme K-Maens

L'algorithme k-means est l'algorithme de clustering le plus connu et le plus utilisé, du fait de sa simplicité de mise en œuvre. Il partitionne les données d'une image en K clusters. Contrairement à d'autres méthodes dites hiérarchiques, qui créent une structure en « arbre de clusters » pour décrire les groupements, k-means ne crée qu'un seul niveau de clusters. L'algorithme renvoie une partition des données, dans laquelle les objets à l'intérieur de chaque cluster sont aussi proches que possible les uns des autres et aussi loin que possible des objets des autres clusters. Chaque cluster de la partition est défini par ses objets et son centroïde [15].

Soit $X = \{x_k / K=1, \dots, n\}$ un sous-ensemble d'un espace n-dimensionnel. Une partition de X en C classes peut être représentée par des sous-groupes mutuellement disjoints $C_1 \dots C_c$, tel que $C_1 \cup \dots \cup C_c = X$.

Chaque vecteur x_k est assigné à une et seulement une classe qui lui est la plus proche.

La fonction objectif correspondante s'exprime sous la forme:

$$J = \sum_{i=1}^c \sum_{k=1}^n U_{ik} d^2(x_k, v_i) \quad (2.5)$$

Ou:

➤ v_i désigne le prototype (centroïde) de cette classe.

- d mesure la similarité entre le prototype et le vecteur de données.
- U_{ik} est un indicateur binaire $\in \{0, 1\}$ désignant l'appartenance ou non du vecteur x_k à la classe C_i ($U_{ik} = 1$ si $x_k \in C_i$ et $U_{ik} = 0$ sinon, $\forall i=1 \dots C$ et $x_k \in X$).

L'indicateur d'appartenance U_{ik} s'exprime comme suit:

$$U(x_k, v_i) = \{1 \text{ si } d^2(x_k, v_i) = \min C(x_k, v_c) \text{ ou } 0 \text{ sinon}\}.$$

L'optimisation se fait de manière itérative. Les étapes de cette classification pouvant être résumé comme suit:

- 1- Initialiser l'ensemble des centroides avec des éléments, soit pris au hasard de l'ensemble de données, soit d'après des connaissances a priori.
- 2- Assigner chaque point au centroïde le plus proche.
- 3- Calculer les nouveaux centroides des classes construites. Les centroïdes étant les centres des classes, ils sont calculés par la moyenne des éléments de la classe.
- 4- Générer une nouvelle partition de l'ensemble de données en assignant chaque point au centroïde le plus proche.
- 5- S'il y a changement dans la partition, réitérer depuis l'étape 3 [16].

b. L'algorithme Fuzzy C-Means [17]

La modélisation de l'imprécision s'effectue en considérant des frontières graduelles au lieu de frontières nettes entre les classes. L'incertitude s'exprime par le fait qu'un pixel possède aussi bien des attributs qui l'assignent à une classe qu'à une autre. La classification floue assigne donc, non pas à un pixel une étiquette relative à une classe unique, mais son degré d'appartenance à chacune des classes. Ces valeurs expriment l'appartenance incertaine d'un pixel à une région et sont appelées degrés d'appartenance. Le degré d'appartenance se situe dans l'intervalle $[0, 1]$ et les classes obtenues ne sont pas forcément disjointes. Dans ce cas, les données X_j ne sont plus assignées à une classe unique, mais à plusieurs par l'intermédiaire de degrés d'appartenance U_{ij} du vecteur X_j à la classe i . Le but des algorithmes de classification est non seulement de calculer les centres de classe B mais aussi l'ensemble des degrés d'appartenance des vecteurs aux classes.

Si U_{ij} est le degré d'appartenance de X_j à la classe i , la matrice $U_{C \times N} [U_{ij}]$ est appelée matrice de C-partitions floues si et seulement si elle satisfait aux conditions:

$$\forall i \in \{1..C\}, \forall j \in \{1 \dots N\} \begin{cases} u_{ij} \in [0,1] \\ 0 < \sum_{j=1}^N u_{ij} < N \end{cases} \quad (2.6)$$

$$\forall j \in \{1 \dots N\} \sum_{i=1}^c u_{ij} = 1 \quad (2.7)$$

La fonctionnelle à minimiser (2.8), et les solutions (2.9), (2.10), au problème du FCM sont décrites par les formules suivantes:

$$J(B, U, X) = \sum_{i=1}^c \sum_{j=1}^N (u_{ij})^m d^2(x_j, b_i) \quad (2.8)$$

$$b_i = \frac{\sum_{k=1}^n u_{ik}^m \cdot x_k}{\sum_{k=1}^n u_{ik}^m} \quad (2.9)$$

$$u_{ij} = \left[\sum_{k=1}^c \left(\frac{d^2(x_j, b_i)}{d^2(x_j, b_k)} \right)^{\frac{2}{m-1}} \right]^{-1} \quad (2.10)$$

c. L'algorithme Possibilités C-Maens [18][14]

Krishnapuram et Keller ont proposé une approche possibiliste des c-moyennes appelée Possibilistic C-Means, ou PCM. Leur approche est censée conduire à une meilleure performance en présence de bruit. Mais leur travail est motivé essentiellement par le désir de remédier au caractère relatif des degrés d'appartenance générés par les FCM, qui est dû à la contrainte probabiliste que la somme de tous les degrés d'appartenance d'un objet aux différentes classes soit égale à 1. Une condition nécessaire pour passer à des degrés d'appartenance "absolus" est donc de lever cette contrainte, ou, plutôt, de l'assouplir afin que les degrés d'appartenance ne soient plus dépendants les uns des autres. Cependant, bien qu'incarnant cette notion de typicalité, l'algorithme PCM présente des performances plutôt médiocres comparées à celles de l'algorithme FCM. Dans le cas général, l'algorithme PCM a tendance à produire des classes concentriques, i. e les classes obtenues ont des centres très proches, voire confondus.

L'algorithme PCM

L'algorithme PCM permet de partitionner un ensemble $X = \{x_1; x_2, K, x_n\} \subset R^p$ en C classes, en minimisant la fonction objective:

$$J(B, U, X) = \sum_{i=1}^c \sum_{j=1}^N (u_{ij})^m d^2(x_j, b_i) + \sum_{i=1}^c \eta_i \sum_{k=1}^N (1 - u_{ik})^m \quad (2.11)$$

$$u_{ij} \in [0,1], 1 \leq i \leq c, 1 \leq k \leq N, \quad (2.12)$$

$$0 < \sum_{k=1}^N u_{ik} < N, 1 \leq i \leq c \quad (2.13)$$

- U_{ik} : C'est la distance euclidienne entre l'élément k et le centre de la classe i .
- $V = \{V_1, K, V_c\}$ est l'ensemble des prototypes des classes C_k dont la distance aux x_i est noté d_{ik} .
- $\eta = \{\eta_1, K, \eta_c\}$ est l'ensemble des termes de pénalité des données atypiques associés à chacune des classes. w_i Est à déterminer par l'utilisateur, et représente une évaluation de l'extension de la classe k . La minimisation de $J(U, V, \eta)$ consiste à minimiser pour toutes les classes:

$$J(B, U, \eta) = \sum_{i=1}^c \sum_{j=1}^N (u_{ij})^m d^2(x_j, b_i) + \sum_{i=1}^c \eta_i \sum_{k=1}^N (1 - u_{ik})^m \quad (2.14)$$

En dérivant la formule (2.14) par rapport à U_{ik} , et en annulant cette dérivée on obtient:

$$u_{ij} = \left[1 + \left(\frac{d_{ik}^2}{\eta_i} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (2.15)$$

Où le paramètre w_i représente le diamètre de la classe et peut être défini a priori par initialisation. Les points aberrants (les points très éloignés de toutes les classes) se voient alors dotés de petits poids pour toutes les classes et dont l'influence dans la phase de classification est réduite. Le PCM souffre, cependant, du problème de coïncidence de classes. On peut avoir, dans certains cas, des classes confondues tandis que des sous-groupes de données normaux fonction objectif est obtenue quand toutes les classes sont coïncidentes.

Des résultats plus ou moins satisfaisants ont été obtenus avec le PCM parce que l'optimisation mène à un minimum local mais pas à un minimum global. Cette propriété n'est pas toujours satisfaite, d'un point de vue théorique.

2. 4. Exemples d'applications de la segmentation

La segmentation est utilisée dans de nombreuses applications dont certaines concernent:

- La reconnaissance de la parole: Pal et Majumder [19] ont été les premiers à introduire le concept de sous-ensemble flou dans ce domaine. Ils décrivent des techniques permettant la discrimination de voyelles et de différentes voix d'orateurs appliquées à la langue Telugu (une des nombreuses langues en usage en Inde).
- La reconnaissance des caractères écrits peut avoir de nombreuses applications, telles que, par exemple, le traitement automatique des documents.
- Le traitement des eaux usées.
- La segmentation d'images RMN (Résonance Magnétique Nucléaire).

2. 5. Conclusion

Dans cet chapitre nous nous intéressons aux techniques de la classification automatique, ou segmentation (clustering). Nous avons vu que la segmentation (clustering) permet de regrouper des objets (individus ou variables) en un nombre limité de groupe ou de classes (segmentes, ou cluster).

La segmentation d'images est l'un des problèmes phares du traitement d'images. Elle consiste à partitionner l'image en un ensemble de régions connexes. L'intérêt de ces régions est de pouvoir être manipulées ensuite via des traitements de haut niveau pour extraire des caractéristiques de forme, de position, de taille, etc.

Le problème est évidemment très mal posé, car on ne sait jamais dire quelle est la segmentation idéale. L'idée est bien sûr que la région se rapproche de la notion d'objet, au sens courant du terme. Néanmoins, on peut dégager des propriétés plus raisonnables qu'on cherche à obtenir dans un algorithme de segmentation, en particulier:

- Stabilité: la segmentation obtenue ne doit pas varier beaucoup lorsque les conditions d'acquisition varie légèrement (bruit, illumination, point de vue, . . .).

- Régularité: les régions obtenues doivent être simples à manipuler (taille suffisante, forme régulière, . . .).

CHAPITRE 3

LES ALGORITHMES POSSIBILISTE D'IMAGES

3.1. Introduction

La théorie des ensembles flous et la théorie des possibilités représentent un cadre de raisonnement permettant de modéliser et de gérer l'information imprécise et incertaine. Ce type d'imperfection étant fréquemment présent dans les concepts et les informations manipulés par l'homme, les chercheurs ont de plus en plus recours à l'utilisation de ces théories. C'est ainsi que leurs domaines d'application se sont élargis. On peut en citer : la médecine et la biologie, l'ingénierie industrielle, la défense, la robotique, les systèmes d'aide à la décision et bien d'autres. L'application de la théorie des ensembles flous en traitement d'image, a permis de développer des méthodes de segmentation et de classification d'images présentant des avantages particuliers.

3.2. La classification floue

Le principe de la classification floue est d'affecter un élément à classer non pas à une classe comme dans les approches classiques (dont fait partie la méthode des k-means), mais à toutes les classes avec un certain degré.

Ce principe découle de ce qu'on appelle la logique floue, qui est venue compléter la logique classique.

3.2.1. La logique floue

Dans la vie de tous le jours, nous nous trouvons dans des situations où les informations dont nous disposons ne sont pas toujours précises. Dans le domaine de la science aussi, des imprécisions peuvent être vues ; par exemple lorsqu'on veut établir une valeur qui caractérise le degré de similarité de deux formes.

L'être humain est habitué à utiliser des informations entachées d'incertitudes et d'imprécision dans la vie de tous les jours, il utilise ces informations incomplètes, raisonne avec elles et prend des décisions. Dans le domaine scientifique, il a été nécessaire de créer une logique qui admette des valeurs de vérité en dehors de l'ensemble $\{ \text{vrai}, \text{faux} \}$ pour pouvoir tenir compte et manipuler ce genre d'information incomplète.

Lukasiewicz propose en 1920 une logique ayant les trois valeurs de vérité suivantes : "vrai", "faux" et "doute". Ces valeurs ont été représentées par l'ensemble $\{ 0, 1, 0.5 \}$, et ont été ensuite étendues à l'intervalle $[0, 1]$.

3.2.2. La théorie des possibilités [20]

La théorie des possibilités constitue un outil qui permet la représentation, le traitement et la fusion d'informations entachées d'imperfection du type possibiliste. Cette forme particulière d'imperfection est rencontrée lorsqu'une information est entachée d'incertitude et où les connaissances disponibles pour décrire cette incertitude sont de nature ambiguë.

Tout en permettant d'exploiter les avantages de la théorie des ensembles flous (pour la représentation des connaissances ambiguës), la théorie des possibilités apporte des mesures ensemblistes permettant de délimiter la probabilité de réalisation d'un événement $A \subseteq \Omega$ (A étant un sous-ensemble algébrique de Ω , ou un ensemble flou défini sur Ω). Ces mesures sont appelées : mesure de possibilité et mesure de nécessité. Dans ce paragraphe, nous allons présenter brièvement cette théorie.

3.2.2.1. La théorie des sous-ensembles flous [20]

La théorie des sous-ensembles flous et les outils de raisonnement qui en découlent, proposent un cadre formel qui permet de modéliser le langage naturel et de gérer l'imprécis et l'incertain.

La notion de sous-ensemble flou permet de garder l'appartenance d'un élément à une classe, C'est-à-dire de permettre à cet élément d'appartenir plus ou moins fortement à cette class. Par abus de langage, On parle généralement d'ensemble flou pour désigner un sous-ensemble flou.

Cette théorie est basée sur la notion de degré d'appartenance. Ces degrés sont des valeurs qui expriment l'appartenance incertaine d'un pixel à une région. Le degré d'appartenance se situe dans l'intervalle $[0, 1]$ et les classes obtenues ne sont pas forcément disjointes.

Dans ce cas, les données X_j ne sont plus assignées à une classe unique, mais à plusieurs par l'intermédiaire de degrés d'appartenance u_{ij} du vecteur X_j à la classe i .

3.2.2.2. Approches floues et possibilistes en segmentation d'images

L'application de la théorie des ensembles flous en traitement d'image, a permis de développer des méthodes de segmentation et de classification d'images présentant des avantages particuliers. De telles approches :

- permettent de retenir, à partir de l'image traitée, plus d'informations que les méthodes traditionnelles (i.e. seuillage, croissance des régions.....) ;

- produisent des régions segmentées plus homogènes que les autres méthodes ;
- sont moins sensibles au bruit ;
- offrent des modèles numériques capables de mieux gérer l'attribution des pixels à telle ou telle région selon leurs situations. Des scores de confiances sont en effet attribués aux pixels, pour mieux définir leurs degrés d'appartenance aux régions en croissance.

3.3. Algorithmes les C- moyennes floue FCM (Fuzzy C- Means)

La modélisation de l'imprécision s'effectue en considérant des frontières graduelles au lieu de frontières nettes entre les classes. L'incertitude s'exprime par le fait qu'un pixel possède aussi bien des attributs qui l'assignent à une classe qu'à une autre. La classification floue assigne donc, non pas à un pixel une étiquette relative à une classe unique, mais son degré d'appartenance à chacune des classes. Ces valeurs expriment l'appartenance incertaine d'un pixel à une région et sont appelées degrés d'appartenance. Le degré d'appartenance se situe dans l'intervalle [0, 1] et les classes obtenues ne sont pas forcément disjointes [21].

3.3.1. Formulation du Problème

L'algorithme de référence utilisé aujourd'hui dans denombreux domaines, est celui des c-moyennes floues (FCM), dans ce cas, les X_i ne sont plus assignés à une unique classe, mais à plusieurs par l'intermédiaire de degrés d'appartenance U_{ij} du vecteur X_j à la classe i . Le but de l'algorithme est alors non seulement de calculer les centres de classe B mais aussi l'ensemble des degrés d'appartenance des vecteurs aux classes. Si U_{ij} est le degré d'appartenance de X_j à la classe i , la matrice $U_{C \times N} [U_{ij}]$ est appelée matrice de C-partitions floues si et seulement si elle satisfait aux conditions :

$$(\forall i \in \{1..C\})(\forall j \in \{1..N\}) U_{ij} \in [0,1], \quad (3.1)$$

$$0 < \sum_{j=1}^N U_{ij} < N \quad (3.2)$$

$$(\forall j \in \{1..N\}) \sum_{i=1}^C U_{ij} = 1 \quad (3.3)$$

Bezdek a montré que le problème de partition de X en C classes floues pouvait être formulé comme la minimisation d'une fonctionnelle $J (B, U, X)$ définie par [22].

$$J(B, U, X) = \sum_{i=1}^C \sum_{j=1}^N U_{ij}^m d^2(x_j, b_i) \quad (3.4)$$

Le prototype B donne :

$$B_i = \frac{\sum_{k=1}^n U_{ik}^m x_k}{\sum_{k=1}^n U_{ik}^m} \quad (3.5)$$

La matrice d'appartenance U donne :

$$U_{ij} = \left[\sum_{k=0}^c \left(\frac{d^2 (x_j, b_i)}{d^2 (x_j, b_k)} \right)^{2/(m-1)} \right]^{-1} \quad (3.6)$$

L'algorithme des FCM consiste alors en l'application itérée de (3.5) et (3.6) jusqu'à stabilité des solutions. Le critère d'arrêt des itérations, définissant cette stabilité, peut par exemple consister en l'étude de la norme de la matrice U ou en la stabilité des centres de classe sur deux itérations successives.

Dans ce qui suit nous présentons les étapes de l'algorithme FCM :

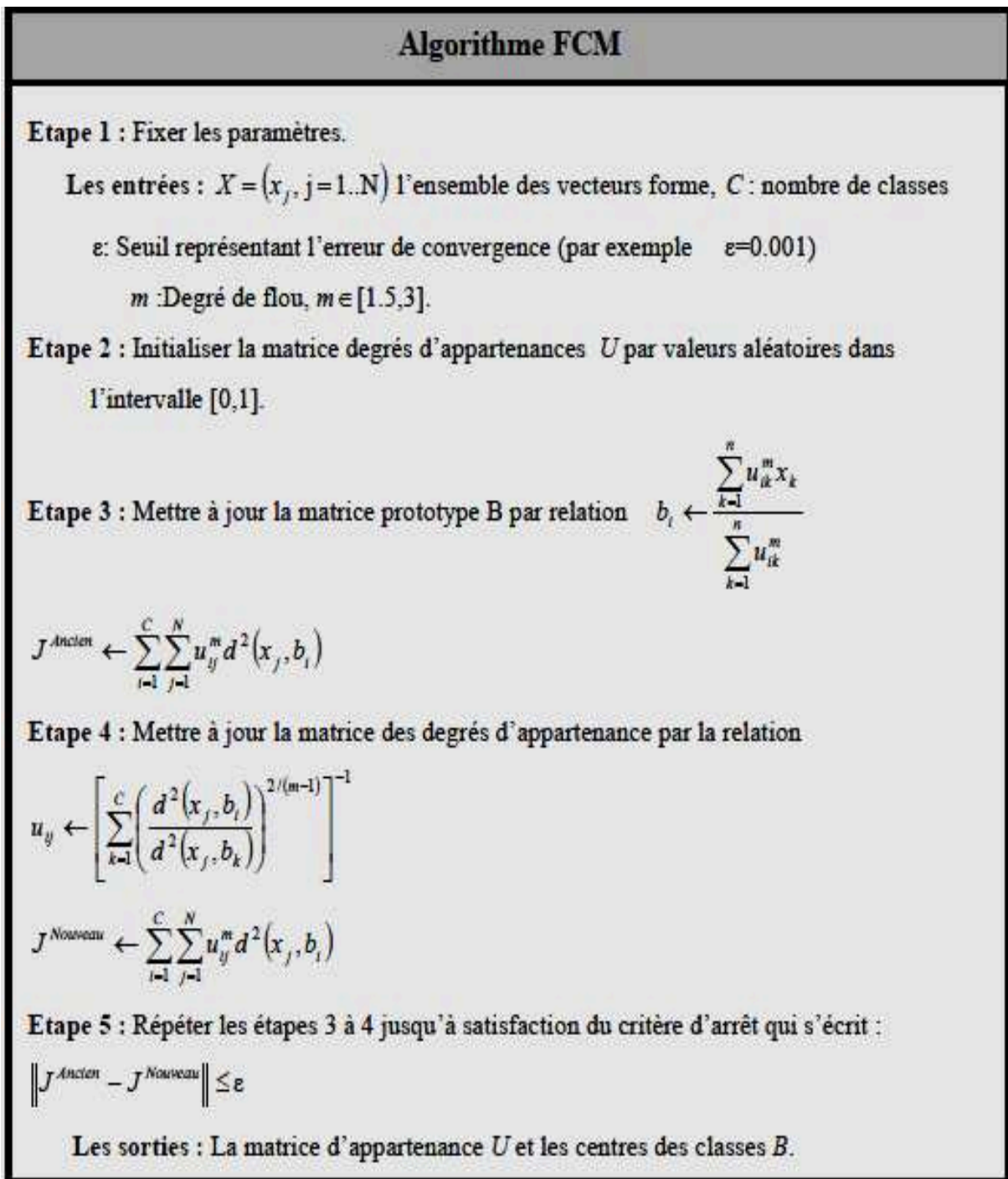


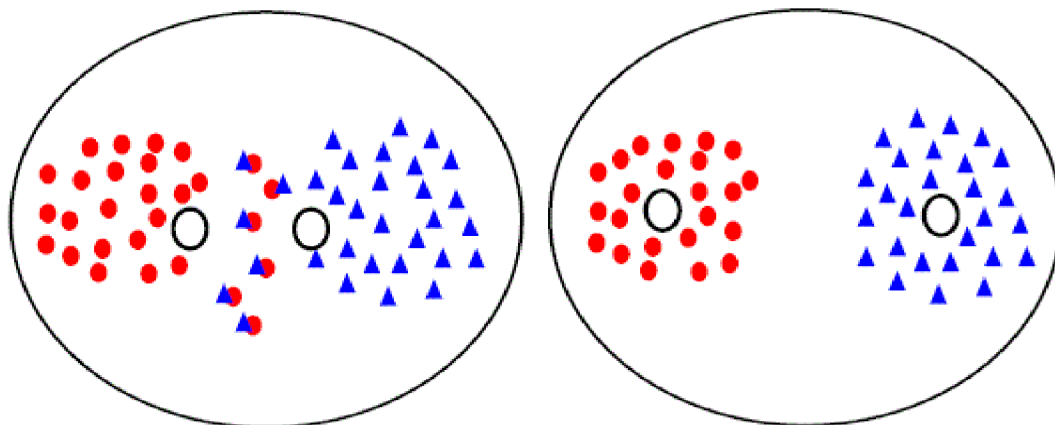
Figure 3.1. Algorithme de FCM.

3.3.2. Les limites du FCM

Un grand nombre d'algorithmes est dérivé du FCM, utilisant des métriques différentes ou des prototypes de dimensions supérieures. Le FCM souffre des plusieurs inconvénients :

- Les degrés d'appartenance sont des degrés relatifs. Autrement dit, l'appartenance d'un individu à une classe dépend de l'appartenance de cet individu aux autres classes. Les fonctions d'appartenance construites sont donc interdépendantes. Aussi, les estimations des centres des classes ne correspondent pas aux centres réels ou typiques.
- Les points aberrants (éloignés) peuvent avoir des valeurs d'appartenance élevées et ils peuvent affecter de façon significative l'estimation des centres des classes.
- Les algorithmes floue modélisent dans la phase de classement l'hésitation ou l'ambiguïté entre classes à partir de règles de décisions floues fixées à priori. Dans la phase de classification, les points ambigus ou non représentatifs des classes peuvent alors affecter la position des centres.

Pour pallier aux deux premiers problèmes voire (**Figure 3.2**), Krishnapuram et Keller [23] ont proposé un nouvel algorithme de classification appelé possibilistic c-means (PCM).



(a) Classification floue classique.

(b) Classification floue Avec rejet des points ambigus.



Figure 3.2.L'influence des points ambigus sur les centres des classes

3.4. Algorithmes les C-moyennes possibilistes PCM (Possibilistic C-means)

Krishnapuram et Keller ont proposé une approche possibiliste des c-moyennes appelée Possibilistic C-Means, ou PCM [24]. Leur approche est censée conduire à une meilleure performance en présence de bruit. Mais leur travail est motivé essentiellement par le désir de remédier au caractère relatif des degrés d'appartenance générés par les FCM.

3.4.1. L'algorithme PCM

Rappelons que l'algorithme des FCM minimise une somme pondérée des carrés des distances entre les vecteurs à regrouper et les centres des classes [22] :

$$J = \sum_{i=1}^C \sum_{j=1}^N U_{ij}^m d^2 (x_j, b_i) \quad (3.7)$$

Le degré d'appartenance d'un élément quelconque à une classe donnée doit naturellement être d'autant plus élevé que le vecteur en question est un élément typique de cette classe. La levée de cette condition conduirait à des fonctions d'appartenance triviales identiquement nulles dans le cas des FCM. L'élimination de l'interférence entre les différents prototypes dans l'approche possibiliste, qui est beaucoup moins contraignante, et qui exige seulement qu'un, au moins, des différents degrés d'appartenance d'un vecteur quelconque ne soit pas nul, nécessite donc la définition d'un nouveau critère d'optimisation. Une nouvelle fonction erreur, à minimiser, a été proposée par Krishnapuram et Keller [24]:

$$J = \sum_{i=1}^C \sum_{j=1}^N U_{ij}^m d^2 (x_j, b_i) + \sum_{i=1}^C \eta_i \sum_{j=1}^N (1 - U_{ij})^m \quad (3.8)$$

Où les paramètres η_i détermination des paramètres de pondération. Sont des nombres positifs homogènes à des distances carrées. Plus précisément, η_i est le carré de la distance séparant le centre de la classe i de l'ensemble des points dont le degré d'appartenance à cette même classe est égal à (0.5).

Krishnapuram et Keller proposent de choisir ce paramètre égal à la distance moyenne floue intra-classe, *i.e.* pour tout i dans $\{1...C\}$ [24] :

$$\eta_i = \frac{\sum_{j=1}^N u_{ij}^m d^2 (x_j, b_i)}{\sum_{ij}^m u_{ij}^m} \quad (3.9)$$

L'algorithme PCM offre la possibilité de fixer, avant itérations, des valeurs pour ces paramètres, ou de faire recalculer les h_i au cours des itérations. Dans ce dernier cas, l'algorithme PCM peut évoluer vers des situations d'instabilité. Les auteurs recommandent alors de fixer des valeurs avant itérations puis, si c'est nécessaire, de recalculer les h_i après convergence avec les nouvelles valeurs de U et B pour appliquer une seconde fois PCM (dans le cas de données très bruitées par exemple).

C'est pourquoi nous retenons dans la relation (3.8) comme formule d'initialisation des paramètres η_i et fixons les valeurs avant itérations.

3.4.2. Nouvelle formule itérative

La convergence de l'algorithme est encore assurée à la suite d'un certain nombre d'itérations dans lesquelles les degrés d'appartenance sont mis à jour à l'aide de la formule suivante :

$$U_{ij} = \frac{1}{1 + \left(\frac{d^2(x_j, b_i)}{\eta_i} \right)^{\frac{1}{m-1}}} \quad (3.10)$$

D'autres choix sont bien sur possibilité. mais n'apportent pas d'amélioration significative dans la classification finale. De fixer, avant itérations, des valeurs pour ces paramètres, ou de faire recalculer les η_i au cours des itération. Dans ce dernier cas, l'algorithme PCM peut évoluer vers des situations d'instabilité. Les auteurs recommandent alors de fixer des valeurs avant itération puis, si c'est nécessaire, de recalculer le h_i après convergence avec les valeurs de U et B pour appliquer une second fois PCM (dans le cas de données très bruitées par exemple).

Dans ce qui suit nous présentons les étapes de l’algorithme PCM :

Algorithme PCM

Etape 1 : Fixer les paramètres.
 Les entrées : $X = (x_j, j=1..N)$ l’ensemble des vecteurs forme, C : nombre de classes
 ε : Seuil représentant l’erreur de convergence (par exemple $\varepsilon=0.001$)
 m : Degré de flou, $m \in [1.5, 3]$, η_i : Degré de pondération.

Etape 2 : Initialiser la matrice degrés d’appartenance U par valeurs aléatoires dans l’intervalle $[0,1]$.

Etape 3 : Mettre à jour la matrice prototype B par relation $b_i \leftarrow \frac{\sum_{k=1}^n u_{ik}^m X_k}{\sum_{k=1}^n u_{ik}^m}$

$J^{Ancien} \leftarrow \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m d^2(x_j, b_i) + \sum_{i=1}^C \eta_i \sum_{j=1}^N (1 - u_{ij})^m$

Etape 4 : Mettre à jour la matrice degrés d’appartenance par la relation

$u_{ij} \leftarrow \frac{1}{1 + \left(\frac{d^2(x_j, b_i)}{\eta_i} \right)^{\frac{1}{m-1}}}$

$J^{Nouveau} \leftarrow \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m d^2(x_j, b_i) + \sum_{i=1}^C \eta_i \sum_{j=1}^N (1 - u_{ij})^m$

Etape 5 : Répéter les étapes 3 à 4 jusqu’à satisfaction du critère d’arrêt qui s’écrit :

$\|J^{Ancien} - J^{Nouveau}\| \leq \varepsilon$

Les sorties : La matrice d’appartenance U et les centres des classes B .

Figure 3.3. Algorithme de PCM.

Avec tous les avantages et les limites des algorithmes de classification automatique, nous avons opté pour la coopération de méthodes pour tirer parti des avantages de chacune, l’intérêt de telle approche est qu’elle exploite la complémentarité d’informations en proposant un système de classification complet.

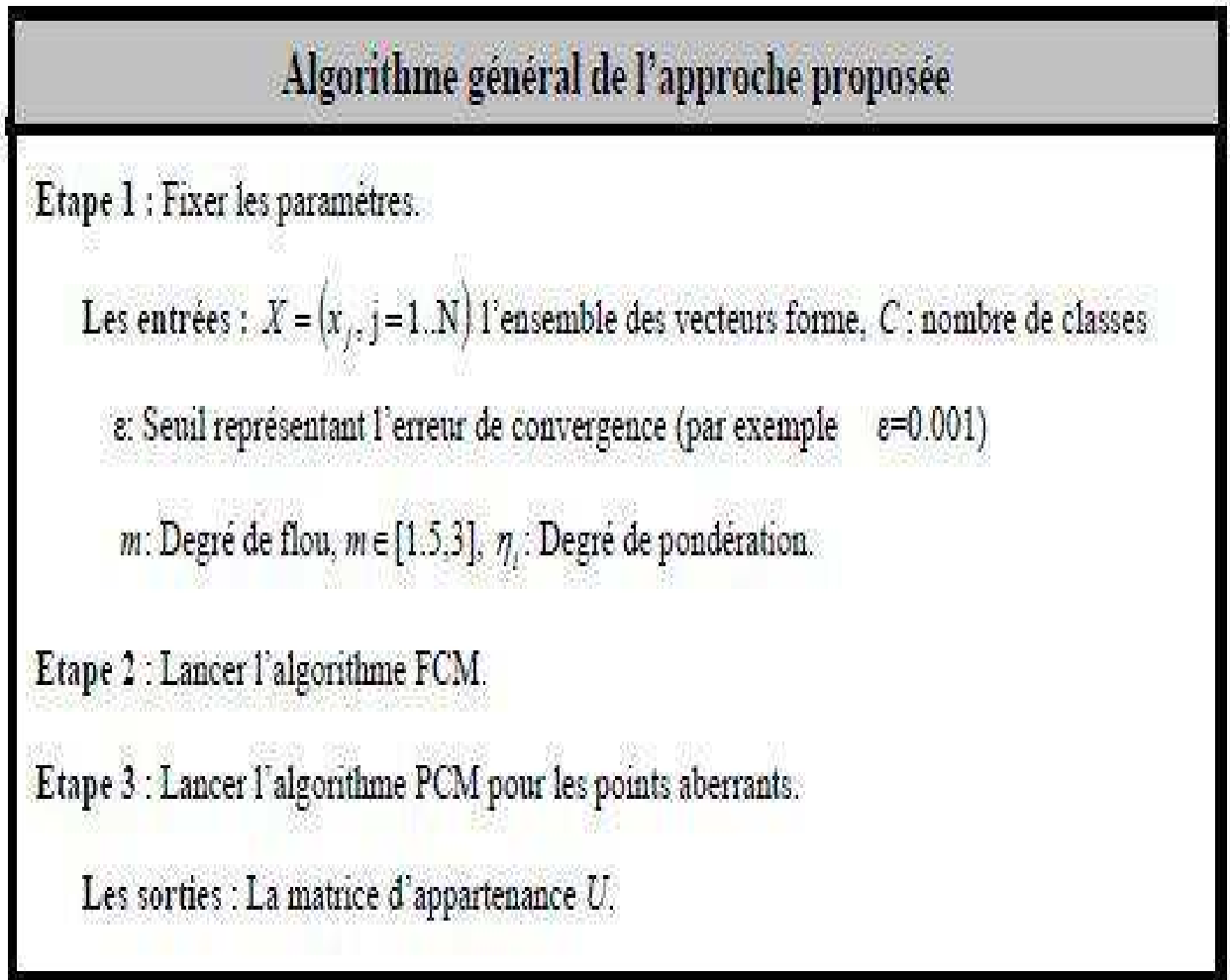


Figure 3.4. Algorithme de coopération FCM et PCM.

3.5. L'avantage et L'inconvénient algorithme possibiliste

- **L'avantage de cet algorithme possibiliste :** c'est qu'il peut regrouper des classes s'il ne trouve pas suffisamment de différence pour les séparer: les éléments classifiés appartiennent fortement à plusieurs classes à la fois, et celles-ci sont centrées au même endroit. Ainsi, si on propose de trouver trop de classes, l'algorithme les regroupe.
- **L'inconvénient de cet algorithme possibiliste :** c'est que les paramètres ne doivent pas être suffisamment discriminants pour les cellules (quel que soit la métrique utilisée), et celles-ci appartiennent autant à chaque classe lorsqu'on utilise l'algorithme: il n'y a en fait qu'une classe de cellules.

3.6. Choix des paramètres de l'algorithme

Nous utilisons l'algorithme de segmentation possibiliste pour segmenter les image bmp il faut définir les différents paramètres gouvernant la méthode, à savoir les valeurs de m et c , le choix des poids η_i , la métrique utilisée et enfin les valeurs forme représentant les pixels des images.

3.6.1. Initialisation de l'algorithme

L'étape initialisation, conditionnant la recherche du minimum, est donc fondamentale. Plusieurs stratégies ont été proposées dans la littérature. La plus simple consiste à demander à un expert de déterminer des régions d'intérêt représentatives des centres des classes.

Tout algorithme peut donc être utilisé, et FCM consiste une excellente manière d'initialiser les données, puisqu'il donne accès à une première estimation de U et de B .

L'algorithme de segmentation possibiliste PCM est donc utilisé dans la suite en initialisation les données avec un FCM ($m=2$ pour une question de rapidité de calcul) et un critère d'arrêt n'autorisant que peu d'itérations.

3.6.2. Choix des vecteurs forme

Le choix des vecteurs forme est fondamental puisque leur pertinence va permettre de discriminer les pixels entre eux. Ce choix est défini suivant le type de modalité. L'image anatomique que nous utilisons est une est bmp une image constituée d'un ensemble de points : les pixels. Chaque point porte des informations de position et de couleur.

3.6.3. Choix de paramètre m

La paramètre m contrôle le degré de flou de partition floue U . Si m est proche de 1, la partition résultante est quasiment non floue, chaque vecteur X_j est assigné à une classe i et une seule avec un degré d'appartenance $U_{ij} = 1$. Une valeur comprise dans l'intervalle $[1.5 ; 2 ; 3]$ est généralement acceptée afin d'assurer la convergence de l'algorithme.

3.6.4. La métrique utilisée [25]

La métrique utilisée conditionne la forme des nuages de points à séparer. D'une manière générale, la distance d^2 est la distance euclidienne et la structure des nuages de points est sphérique. D'autres choix sont possibles, permettant de détecter des nuages de forme plus complexe. Dans notre cas, nous n'avons supposé aucun a priori quant à la forme des nuages de

points à séparer dans l'espace des vecteurs forme. Nous avons également opté pour la distance la plus usuelle et la plus rapide à calculer, à savoir la distance euclidienne (définition initiale du FCM et PCM). Nous supposons donc que la forme des nuages de points représentant les classes de tissus est quasi-sphérique.

3.7. Conclusion

Dans ce chapitre nous avons présenté les algorithmes de segmentations floue et possibiliste des images, Nous avons remarqué à partir de cette étude que le problème majeur de l'algorithme FCM (C-moyennes floues) est sa sensibilité aux points aberrants (éloignés) qui peuvent avoir des valeurs d'appartenance élevées et ils peuvent affecter de façon significative l'estimation des centre des classes, pour éliminer le problème de l'algorithme de FCM nous avons utilisé l'algorithme de possibiliste PCM (C-moyennes possibiliste) pour les points aberrante. Ou faire la coopération entre l'algorithme FCM et PCM.

CHAPITRE 4

Réalisation et évaluation des résultats

4.1. Introduction

Après avoir présenté un état de l'art sur les algorithmes de la segmentation possibiliste d'images. Dans ce chapitre nous appliquons les algorithmes étudiés dans les chapitres précédents. Nous allons présenter notre travail qui consiste concevoir et à réaliser un système de segmentation d'images. Et évaluons des résultats de segmentation pour comparer les performances des algorithmes. Pourtant, l'évaluation est très souvent réalisée de manière qualitative par une inspection visuelle alors qu'il serait souhaitable d'utiliser une approche objective et quantitative.

4.2. Utilisation de l'environnement de développement Visual C#

Pour l'implémentation de notre application, nous avons opté pour le langage C# en utilisant l'EDI (Environnement de développement intégrée) Microsoft visual studio 2010 sous Windows 7.

Microsoft visual studio 2010 Garantisiez des résultats de qualité de la conception au déploiement et Simplifiez votre processus de développement dans un environnement intégré au système de développement Microsoft Visual Studio. Accélérez le processus de codage en travaillant à partir d'un nombre croissant de plateformes, notamment la technologie Microsoft SharePoint et les applications informatiques en nuage.

Les développeurs et les testeurs peuvent travailler de manière plus efficace avec des outils de test et de débogage intégrés pour corriger rapidement et aisément des bogues. la figure (Figure 4.1) présente l'interface de Microsoft visual studio 2010.

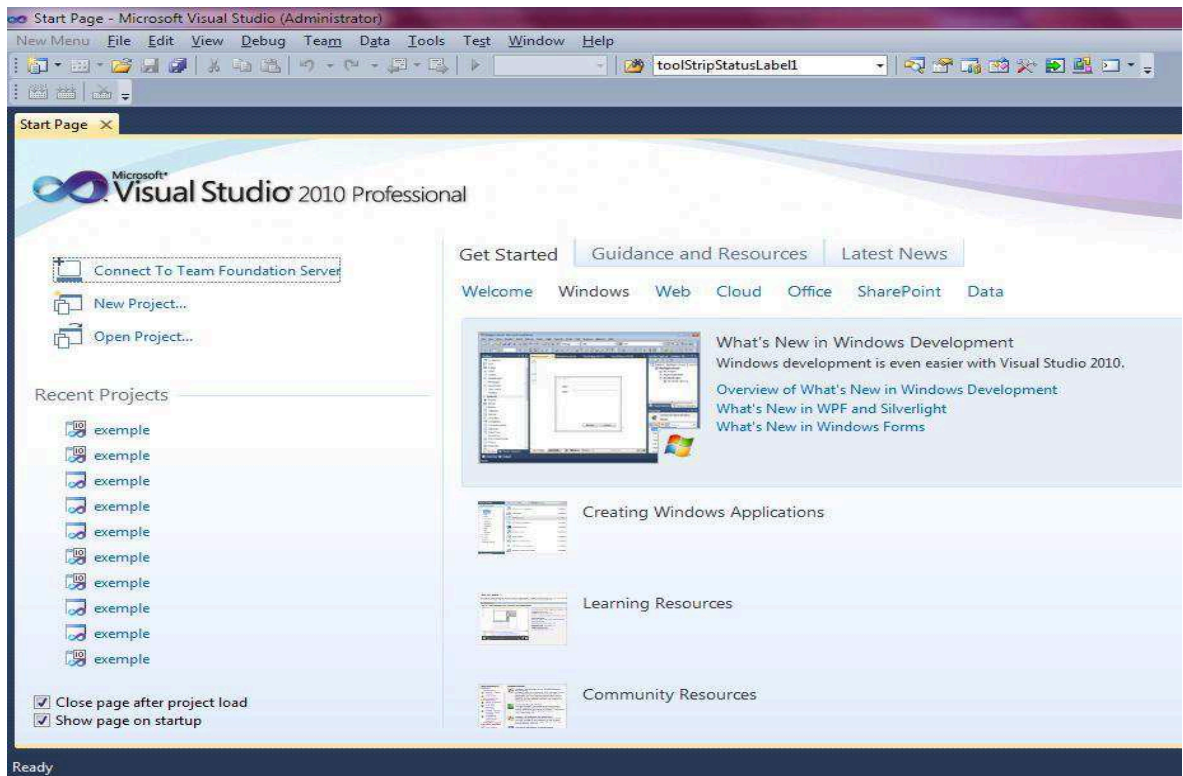


Figure 4.1. L'interface de Microsoft Visual Studio 2010.

4.3. Présentation et implémentation du système

Le système qu'on a réalisé est consisté consiste un système de segmentation floue et possibiliste, et faire une coopération entre eux, guidée par ces connaissances a priori. Une application a été développée pour la segmentation des images de format BMP, qui est le format le plus simple, le plus standard et le plus répandu.

4.4. Système de segmentation d'image

La structure de notre Système est présentée par la figure (Figure 4.2) :

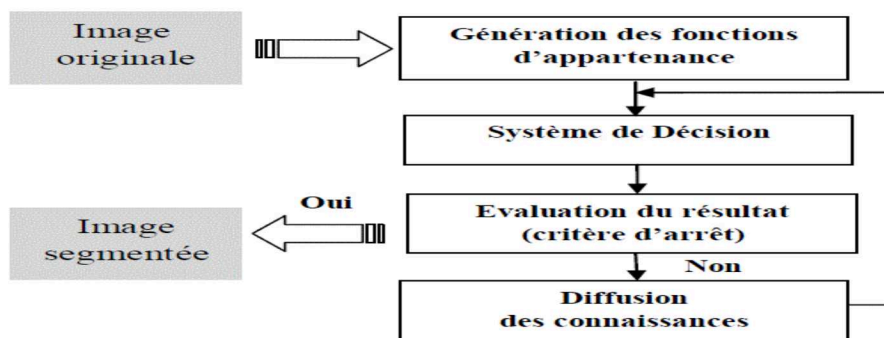


Figure 4.2. les étape de segmentation d'une image.

4.5. Critères de validité

Les résultats de la segmentation obtenus dépendent fortement du nombre de classes fixé. Il est donc primordial de choisir le nombre exact de classes pour espérer avoir une bonne qualité de segmentation. Ceci n'est pas toujours simple, surtout en présence de bruits. Pour les algorithmes FCM et PCM nous avons obtenu des résultats différents selon le nombre de classes k fixé par l'utilisateur. Pour des classes bien séparées, les algorithmes de segmentation retrouvent généralement le même nombre de clusters. Dans le contexte de la segmentation, il est naturel de s'interroger sur la validité de la partition obtenue. Les groupes découverts correspondent-ils à nos connaissances a priori ? Correspondent-ils vraiment à l'ensemble d'objets dont on dispose ? De ces algorithmes de segmentation, laquelle est la plus pertinente ? Ces différentes questions permettent de distinguer deux catégories de critères : les critères de «contraste» et les critères d'«adéquation à un modèle». Les premiers recherchent une variabilité inter-région, alors que les seconds recherchent une uniformité en intensité ou en couleur à l'intérieur des régions. On va représenter quelque critère de validité. Parmi les critères de contraste, nous présentons ci-dessous le critère de Rosenberger, et le critère de Zeboudj. Parmi les seconds, nous présentons le critère Borsotti et al, et critère d'uniformité.

Si f représente un attribut du pixel (en général son intensité ou sa couleur), on notera m_i (resp. σ_i) la moyenne (resp. l'écart type) de f dans la région R_i ou le $i \in \{1 \dots c\}$ le nombre des classes.

4.5.1. Critère contraste de Rosenberger [25]

Rosenberger commence par caractériser chaque région en région texturée ou uniforme, Il calcule ensuite la disparité intra-région, notée \underline{D} et la disparité inter-région, notée \bar{D} . La première correspond à l'écart-type des intensités pour une région uniforme et à un ensemble d'attributs de texture pour une région texturée. \bar{D} est égale à la différence des moyennes pour deux régions uniformes, à la distance euclidienne entre attributs de texture pour deux régions texturées et à 1 pour une région texturée et une région uniforme. La disparité intra-région globale est égale à la moyenne pondérée des disparités calculées pour chaque région (R_i):

$$D = \frac{1}{N} \sum_{R_i} \frac{A_i}{A} D_i \quad (4.1)$$

Où D_i : Disparités intra-région calculées pour chaque région.

Et de même pour la disparité inter-région globale.

$$\bar{D} = \frac{1}{N} \sum_{R_i} \frac{A_i}{A} \bar{D}_i \quad (4.2)$$

Où \bar{D}_i : Disparités inter-région calculées pour chaque région.

Finalement le critère de Rosenberger est égal à :

$$\frac{\bar{D}-D}{2} \quad (4.3)$$

4.5.2. Contraste de Zeboudj [26]

Cet indice prend en compte le contraste intérieur et le contraste extérieur aux régions, mesurés sur un voisinage $W(s)$ du pixel s .

Soit $c(s, t) = \frac{|f(s)-f(t)|}{L-1}$ le contraste entre deux pixels s et t , avec f représentant l'intensité et L le maximum des intensités.

Le contraste intérieur d'une région R_i est :

$$E_i = \frac{1}{l_i} \sum_{s \in F_i} \max\{c(s, t), t \in W(s), t \in R_i\} \quad (4.4)$$

Où F_i est la frontière de R_i et l_i la longueur de F_i .

Le contraste se R_i est :

$$C(R_i) = \begin{cases} 1 - \frac{l_i}{E_i} & \text{si } 0 < l_i < E_i \\ E_i & \text{si } l_i = 0 \\ 0 & \text{Sinon} \end{cases} \quad (4.5)$$

Le contraste global est enfin

$$\frac{1}{A} \sum_i A_i \cdot C(R_i) \quad (4.6)$$

Cet indice a été utilisé dans [26] pour comparer des segmentations en régions sur des images réelles et de synthèse. Cet indice n'est pas adapté aux images trop bruitées ou texturées.

4.5.3. Mesure de dissimilarité de Liu et Yang [27]

Ce critère est basé sur le nombre de régions, l'aire des régions et la couleur moyenne, dans l'espace RGB :

$$\frac{1}{10000 \times A} \sqrt{N} \sum_{i=1}^N \left(\frac{e_i^2}{\sqrt{A_i}} \right) \quad (4.7)$$

Où e_i est la somme des distances euclidiennes entre les vecteurs couleur des pixels de la région R_i et le vecteur couleur attribué à la région R_i dans l'image segmentée (en général la moyenne des couleurs de la région).

Le critère doit être faible. Les termes \sqrt{N} au numérateur et $\sqrt{A_i}$ au dénominateur de l'expression pénalisent la sur-segmentation.

4.5.4. Critère d'uniformité de mesure [28]

Ce critère simple est basé sur la somme des variances des régions.

$$u = 1 - 2 * c * \frac{\sum_{j=0}^c \sum_{i \in R_j} (f_i - \mu_j)^2}{N * (f_{max} - f_{min})} \quad (4.8)$$

Ou

- C : est le nombre de classes ;
- R_i : La région de chaque classes j ;
- f_i : La valeurs dans chaque pixel i ;
- μ_j : La moyennes de division la valeur de pixel i sur la somme de pixel i ;
- N : est le nombre totale des pixels dans un image donnée ;
- f_{min} et f_{max} : représenter la valeur minimale et maximale en respective dans une image donnée.

La valeur d'uniformité varie entre 0 et 1. Dans notre cas nous avons utilisé la critère d'uniformité de mesure. L'avantage de ce critère est d'être facilement de comparer entre les algorithmes de segmentation, l'algorithme qui donne le plus grand valeurs de l'uniformité est le mieux pour la segmentation d'images.

4.6. Les résultats de segmentation d'images par FCM, PCM et FCM-PCM

Dans la littérature, les auteurs qui abordent le problème de l'évaluation sont peu nombreux. En 1996, Zhang a dressé un état de l'art des méthodes utilisées pour valider les segmentations, mais la plupart d'entre elles sont appliquées aux images en niveaux de gris.

Les trois algorithmes FCM, PCM et FCM-PCM sont implémentés sur un PC portable Acer Aspire 5735Z avec un processeur Intel (R) Pentium (R) CPU T3400 @ 2.16GHz 2.17 GHz et avec mémoire installée (RAM) 3,00 Go.

4.6.1. Présentation des résultats obtenus par FCM

Le tableau (Table 4.1) présenter les résultats obtenus après segmentation des différents d'images (a), (b), (c) et (d) par FCM.


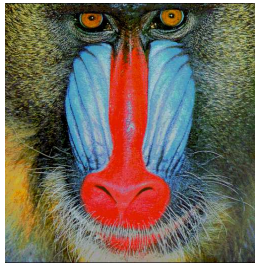

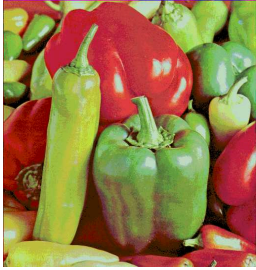
Image Originale	 <p>(a) Lena</p>	 <p>(b) Baboon</p>	 <p>(c) Camera man</p>	 <p>(d) Peppers</p>
Nombre de classes	C=2.	C=2.	C=2.	C=2.
C=3.	C=3.	C=3.	C=3.	C=3.
C=4.	C=4.	C=4.	C=4.	C=4.
C=5.	C=5.	C=5.	C=5.	C=5.

Table 4.1. Les résultats de segmentation d'images (a), (b), (c) et (d) par FCM.

4.6.1. Présentation des résultats obtenus par PCM

Le tableau. (table 4.2) présenter les résultats obtenus après segmentation des différents d'images (a), (b), (c) et (d) par PCM.


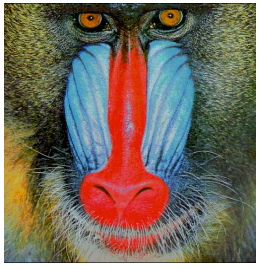

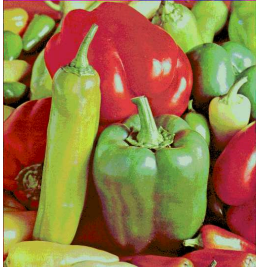

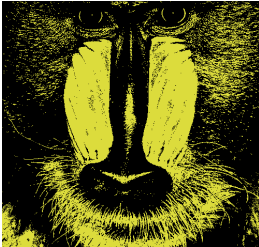



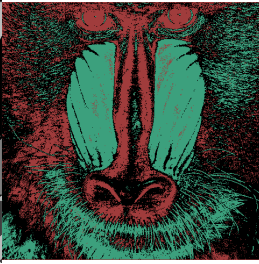

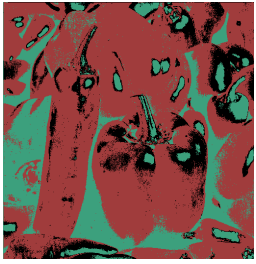



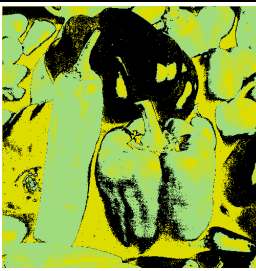




Image Originale Nombre de classes	 (a) Lena	 (b) Baboon	 (c) Camera man	 (d) Peppers
C=2.				
C=3.				
C=4.				
C=5.				

Table 4.2. Les résultats de segmentation d'images (a), (b), (c) et (d) par PCM.

4.6.2. Présentation des résultats obtenus par la coopération FCM-PCM

Le tableau (Table 4.3) présenter les résultats obtenus après segmentation des différents d'images (a), (b), (c) et (d) par l'algorithme coopérative FCM et PCM.


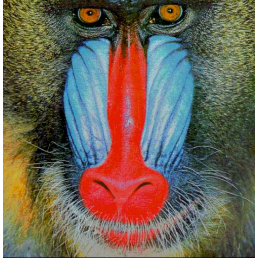

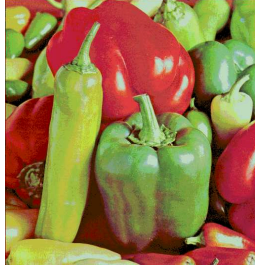







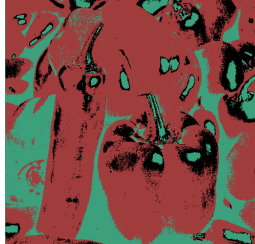





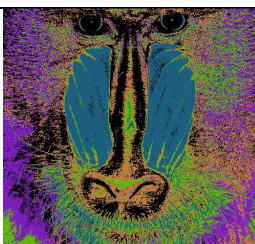

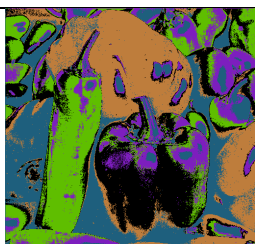
Image Originale				
Nombre de classes	(a) Lena	(b) Baboon	(c) Camera mane	(d)Peppers
C=2.				
C=3.				
C=4.				
C=5.				

Table 4.3. Les résultats de segmentation d'images (a), (b), (c) et (d) par FCM et PCM.

4.7. Evaluation des résultats obtenus par FCM, PCM et FCM-PCM

Le tableau (**Table.4.4**) présente les résultats expérimentales de segmentation d'images (a), (b), (c) et (d) obtenues par les trois algorithmes par différentes valeurs des paramètres (temps CPU l'unité est le second, valeur minimum de la fonction objectif, ainsi que la valeur de critère de validation d'uniformité) pour chaque classe fixer par l'utilisateur.

4.8. Discussion des résultats obtenus par FCM, PCM et FCM-PCM

A partir de tableau (**Table 1.4**) nous avons remarqué que :

- Les trois algorithmes fournissent une bonne segmentation soit en terme visuellement soit en terme critère d'uniformité (le critère d'uniformité est supérieure à 0.820162)
- Les valeurs des centres de classes obtenues pour chaque algorithme et pour le même nombre de classe choisi sont presque les même
- Les valeurs du critère d'uniformité sont plus grandes (proche de 1) pour l'algorithme de coopération, par contre les valeurs dans les deux algorithmes FCM et PCM sont plus proche.
- Les valeurs du critère d'uniformité sont proches de la valeur 1 (donc meilleure segmentation) lorsque sera mieux dans le nombre de classe est égale à 4.
- Le temps CPU de segmentation pour les trois algorithmes (FCM, PCM et FCM_PCM) augmente avec la croissance du nombre de classes choisi.
- La valeur minimale de la fonction objectif décroît avec la croissance de nombre de classes car la distance entre le centre de chaque classe et les points appartenant à cette classe devient petite.
- La taille de l'image à segmenter est un facteur très important pour la segmentation d'images c'est-à-dire, si la taille d'image est grande l'opération de la segmentation consomme beaucoup de temps machine par exemple la segmentation de l'image baboon prend à peu près 68 second.
- La segmentation d'images utilisant les deux algorithmes FCM et FCM-PCM.

Chapitre 4 - Réalisation et évaluation des résultats

<i>Thme</i>	<i>Image originale</i>	<i>C</i>	<i>Les centres</i>	<i>Fonction objectif</i>	<i>Le temps de CPU (Seconds)</i>	<i>Uniformité</i>
<i>FCM</i>	(a) Lena	2	158, 80	1096439	1.42	0.885367
		3	68, 133, 86	687400.1	7.64	0.918157
		4	146, 106, 194, 60	456425.7	16.22	0.966776
		5	133, 159, 101, 200, 59	348461.1	11.45	0.958342
	(b) Baboon	2	65, 211	7079746	6.40	0.820162
		3	94, 15, 224	2949006	18.54	0.879030
		4	164, 238, 89, 14	1509705	26.51	0.994712
		5	96, 14, 164, 239, 81	1169849	1.07.10	0.981880
	(c) Camera man	2	24, 156	1076606	2.11	0.920971
		3	18, 169, 121	573231.4	5.24	0.953157
		4	113, 179, 17, 155	407567.7	9.22	0.974836
		5	15, 157, 179, 123, 72	327978.9	12.80	0.964923
	(d) Peppers	2	15, 115	4673523	13.83	0.867144
		3	99, 9, 183	2377615	10.08	0.965000
		4	113, 84, 183, 9	1171541	54.83	0.992810
		5	183, 9, 83, 118, 105	872070.2	33.13	0.985307
<i>PCM</i>	(a) Lena	2	159, 80	1102204	1.06	0.885396
		3	68, 186, 133	687465.4	5.73	0.918157
		4	147, 107, 60, 195	456525.6	5.99	0.966541
		5	165, 199, 129, 98, 58	345850.8	27.38	0.957842
	(b) Baboon	2	65, 210	7094067	10.11	0.820162
		3	94, 15, 225	2915559	15.07	0.879030
		4	238, 164, 89, 14	1509311	24.42	0.994712
		5	6, 21, 239, 89, 164	1346925	47.77	0.977734
	(c) Camera man	2	156, 24	1076442	2.46	0.921022
		3	18, 120, 168	572760.6	8.00	0.953157
		4	17, 112, 177, 153	407585.8	9.05	0.974313
		5	69, 156, 121, 15, 178	329686.2	18.72	0.964906
	(d) Peppers	2	15, 115	4673523	15.12	0.867144
		3	183, 99, 9	2369712	25.10	0.965000
		4	25, 9, 183, 113	1156482	19.48	0.994899
		5	112, 84, 9, 180, 192	1063654	1.02.73	0.983911
<i>FCM et PCM</i>	(a) Lena	2	81, 159	1101968	2.84	0.885417
		3	129, 182, 66	679941.4	8.95	0.918244
		4	194, 106, 146, 60	456456.1	16.09	0.972609
		5	58, 156, 129, 199, 99	344863	17.10	0.967920
	(b) Baboon	2	210, 65	7106438	7.63	0.820162
		3	15, 94, 224	2948518	20.86	0.879029
		4	164, 238, 14, 89	1509342	17.95	0.994712
		5	14, 164 81, 239, 96	1168869	46.62	0.981880
	(c) Camera man	2	156, 24	1076442	2.47	0.921022
		3	18, 121, 169	573211.2	3.92	0.953268
		4	16, 172, 141, 103	421894	33.42	0.975289
		5	93, 16, 182, 130, 161	321599.4	31.61	0.965289
	(d) Peppers	2	115, 15	4665659	22.10	0.867145
		3	183, 99, 9	2369712	17.41	0.965000
		4	9, 183, 84, 113	1172117	31.79	0.994631
		5	100, 116, 82, 9, 183	85088.4	27.42	0.986581

Table 4.4. les résultats expérimentales de segmentation d'images obtenues par les trois algorithmes par différentes valeurs de nombre de classes.

4.9. Quelques interfaces de l'application

L'interface principale de notre logiciel est présentée par la figure suivante (voir Figure 4.13).



Figure 4.3. L'interface segmentée.

Dans le menu fichier on peut charger, arrêter l'opération de la segmentation d'image, et sortir de l'application (voir Figure 4.4).



Figure 4.4. Le menu de Fichier.

Dans le menu Segmentation (voir Figure 4.5.) on peut appliquer les traitements suivants sur l'image:

- Segmentation C-moyennes floue (FCM);
- Segmentation C-moyennes possibiliste (PCM);
- Segmentation C-moyennes floue et possibiliste (FCM-PCM).



Figure 4.5. Le menu segmentation.

4.10. Conclusion

L'évaluation des algorithmes de traitement d'images et notamment ceux de segmentation constitue un problème d'importance, tant pour le choix d'un algorithme et de son paramétrage pour un utilisateur que pour la comparaison avec l'existant d'un nouvel algorithme par un chercheur. Le nombre important de critères quantitatifs témoigne d'un besoin de toute la communauté en traitement d'images.

Dans ce chapitre, Nous avons présenté les résultats de notre travail qui consiste à utiliser plus d'un algorithme pour segmenter des images en vue d'évaluer la qualité de la segmentation pour chaque algorithme. On utilise pour valider les résultats le critère d'uniformité, nous avons vu quelque remarque sur les résultats de segmentation obtenus par les trois algorithmes utilisés, puis nous avons présenté quelque interface de notre application.

CONCLUSION GENERALE

CONCLUSION GENERALE

La segmentation consiste à créer une partition de l'image observée en un certain nombre de régions qui pourront par la suite être traitées de manière différentes. Comme nous avons pu le voir, la plupart des techniques de segmentation nécessitent le choix des valeurs de certains paramètres dont les résultats du segmentation peuvent dépendre très fortement. Nous nous sommes intéressés, en particulier, aux c-moyennes possibilistes, qui sont l'une des très nombreuses généralisations dérivant des algorithmes bien connus que sont les c-moyennes floues.

Dans ce travail nous avons démontré l'importance et les difficultés de ces travaux de segmentation floue.

Trois raisons expliquent ces difficultés :

- Il existe une très grande variété d'images qui diffèrent par leur, type, taille, leur forme, leur position, leurs couleur, leur composition...etc.
- La seconde raison vient de ce que la donnée issue de l'acquisition d'image est sensible au bruit de fond et à l'échantillonnage.
- Le problème majeur de l'algorithme FCM (C-moyennes floues) est sa sensibilité aux points aberrants.

Pour éliminer le difficultés de l'algorithme de FCM nous avons utilisé l'algorithme de possibiliste PCM (C-moyennes possibiliste). Nous nous sommes pour cela placés dans le cadre de la coopération entre l'algorithme flou FCM et l'algorithme possibiliste PCM afin de rendre l'algorithme plus robuste face aux imprécisions et aux données aberrantes. En second lieu, nous avons fusionné différentes données. Nous avons présenté et évalué les résultats de notre travail qui consiste à utiliser les trois algorithmes (FCM, PCM, FCM-PCM) pour segmenter des images en vue d'améliorer la qualité de la segmentation.

La bonne qualité de nos solutions tient au fait que :

- ✓ Elle est robuste en présence des points aberrants.
- ✓ C'est un système à base de la théorie floue et possibiliste.

Plusieurs perspectives peuvent être dégagées :

- Une étude plus approfondie resterait nécessaire pour l'évaluer par rapport aux diverses techniques de regroupement existantes. Une étude mathématique nous semble aussi nécessaire pour la mise au point d'une méthodologie concernant le choix des paramètres et de la partition initiale.
- Les PCM ne sont qu'un cas particulier de technique possibiliste du regroupement, et que d'autres fonctions objectifs peuvent être à la base d'autres formules itératives. Notre travail futur concernera la généralisation de notre approche à d'autres méthodes prenant en compte la différence de volume ou de forme des groupes recherchés, en utilisant, par exemple, des matrices de covariance.
- Introduire les informations globales de l'image dans le processus de segmentation..
- Réduire le temps de segmentation qui est trop élevé et ceci en utilisant les méthodes d'optimisation.
- Effectuer une étude approfondie pour le choix des critères d'évaluation.

REFERENCES BIBLIOGRAPHIQUES

BIBLIOGRAPHIE

- [1] M. BERGOUNIOUX, 'Quelques méthodes mathématiques pour le traitement d'image', cel-00125868, version 4 - 4 Jan 2009.
- [2] M. COLINET, 'Le Traitement Des Images', Centre pour la formation à l'informatique dans le Secondaire, 5.86 Mai 2004.
- [3] C. PARISEL, 'Structure du format BMP', sa lecture, sa construction et son écriture, Mars 2003.
- [4] V. BENONY, A. SEDOGLAVIC, 'Stéganographie élémentaire avec un fichier BMP', Université des Sciences et Technologies de Lille, 2003–2004.
- [5] N. THI OANH, 'Binarisation d'images de documents graphiques', Maître de conférences à Université de Nancy 2 Chercheur à l'équipe QGAR, INRIA Lorraine Nancy, France, juin - novembre 2004.
- [6] MR M. MAAMAR, 'Segmentation d'image par Coopération Régions-Contours', Magister en informatique, Ecole nationale supérieure en informatique, ESI 2012.
- [7] J.P. GASTELLU-ETCHEGORRY, 'Acquisition Et Traitement D'image Numérique', Université Paul Sabatier, Avril 2008.
- [8] M. EL HASSANI, 'Segmentation basée-région pour l'amélioration de séquences vidéo : Algorithme-Architecture dans un contexte temps réel', Doctorat, L'université De Caen. 07 Aout 2006.
- [9] P. ANDRES, A.ESCALANTE, 'Une approche métrique pour la segmentation d'images', Université Paris Dauphine, Doctorat, 24 novembre 2005.
- [10] Melle M.LEHAMEL, 'Segmentation d'images texturées à partir des attributs fractals', Magister, Université Mouloud Mammeri de Tizi-Ouzou, .
- [11] Y. YAKIMOVSKY, 'Boundary and objet detection in real world images,' ACM, vol. 23, no. 4, pp.599-618, 1976.
- [12] R. DERICHE, ' International journal of computer vision', pp. 167-187, 1987.
- [13] Z. HAKIMA, 'Clustering par fusion floue de données appliqué à la segmentation d'images IRM', Magister, Université M'hamed Bougrara De Boumerdes, 2007/2008.
- [14] M. ABDELOUAHAB,H. CHELLAKH 'La Segmentation des IRM Cérébrales Pathologiques par une combinaison Floue Possibiliste Markovienne', Ecole Doctorale Nationale – Département d'Informatique- Université Farhet Abbas de Sétif (Algérie).

- [15] F. D'HONDT, B. EL KHAYATI, 'Etude de méthodes de Clustering pour la segmentation d'images en couleurs', Faculté Polytechnique de Mons.
- [16] PR L. BOUMGHAR, 'Déploiement sur une plateforme de visualisation, d'un algorithme coopératif pour la segmentation d'images IRM basé sur les systèmes multi-agents', Projet de Fin d'Étude, Université des Sciences et de la Technologie Houari Boumediene.
- [17] M. SEMCHEDINE, L. TOUMI, A. MOUSSAOUI, 'Un Système Multi-agents pour La Classification Coopérative D'images IRM Cérébrales', Ecole doctorale nationale STIC – SETIF, Université Ferhat Abbas - Setif.
- [18] T. LAYZID, S. MOUSSA, M. ABDELOUAHAB, 'Une Approche Robuste Semi-Supervisée Pour la Détection des Tumeurs Cérébrales', Université Ferhat Abba, JIG'2007.
- [19] K. SANKAR PAL AND DWIJESH D. MAJUMDER, 'Fuzzy Sets and Decisionmaking Approaches in Vowel and Speaker Recognition', IEEE Trans. Syst. Man Cybern., pp. 625-629, August, 1977.
- [20] E. WAEL, 'Segmentation Itérative D'images Par Propagation De Connaissances Dans Le Domaine Possibiliste : Application Á La Détection De Tumeurs En Imagerie Mammographique', Doctorat, tel-00741996, version 1 - 15 Oct 2012.
- [21] H. ZOUAOUI, A. MOUSSAOUI, 'Clustering par fusion floue de données appliqué à la segmentation d'images IRM cérébrales', Université M'hamed Bougrara de Boumerdes.
- [22] J.C. BEZDEK, 'Pattern Recognition with Fuzzy Objective Function Algorithms', Plenum Press, New- York, 1981.
- [23] M. SEMCHEDINE, L. TOUMI, A. MOUSSAOUI, 'Classification Adaptative Robuste pour la Segmentation d'Images IRM Cérébrales', Université Ferhat Abbas de Sétif, JIG'2007.
- [24] R. KRISHN, 'The Possibilistic C-Means Algorithm: Insights and Recommendations'. IEEEtransactions on Fuzzy Systems, 1996.
- [25] S. CHABRIER, C. ROSENBERGER, H. LAURENT, B. EMILE, P. MARCHÉ, 'Evaluating the segmentation result of a gray-level image, In 12th EUSIPCO', Vienne, Austria, september 2004.
- [26] J.P COCQUEREZ, S. EDITORS PHILIPP, 'Analyse d'images: filtrage et segmentation', Masson, Paris, 1995.

[27] J. LIU, Y.H. YANG, 'Multiresolution color image segmentation', IEEE Trans, on PAMI, 16(7), p. 689-700, 1994.

[28] PENG-YENG YIN, 'A fast scheme for optimal thresholding using genetic algorithms', Université Ming Chuan, Tapei, Taiwan, 14 October 1987.

Sites web

[29] Toulouse, <http://www.map.toulou.archi.fr/works/panoformation/imagenum.htm>, consulter le 01/05/2013

[30] wikipedia, <http://www.wikipedia.org/>, consulté le : 01/05/2013

[31] Commentcamarche, <http://www.commentcamarche.net/contents/1216-traitement-d-images>, consulté le : 06/05/2013.

[32] Commentcamarche, <http://www.commentcamarche.net/contents/1191-infographie>, consulté le : 10/05/2013.

[33] Webmarketing, <http://www.webmarketing-com.com/2012/11/06/16580-quels-sont-les-9-formats-differents-pour-une-image>, consulté le : 12/05/2013.

[34] Développez, http://xphilipp.developpez.com/articles/filtres/?page=page_3, consulté le : 20/05/2013.

[35] Misfu, <http://www.misfu.com/cours/tutoriel/morphologie-mathematique-642.html> consulté le : 25/05/2013.

[36] Wikipedia, http://fr.wikipedia.org/wiki/D%C3%A9tection_de_contours, consulté le : 26/05/2013.

[37] Wikipedia, http://fr.wikipedia.org/wiki/Segmentation_d'image, 27/05/2013

Résumé:

Le travail suivante présente un système de segmentation d'images et ces performances. Ce système est basé sur la segmentation par l'algorithmes flous et possibilistes de classification non supervisée connus sous le nom des c-moyennes floues (FCM) et c- moyennes possibiliste (PCM). Les images utilisées dans notre système sont de format BMP puisqu'il est le plus simple, le plus standard et le plus répandu, Le but de ce système est de segmenter l'images par le FCM qui modélise l'incertitude et l'imprécision, et le PCM qui génère un degré d'appartenance absolu reflétant de manière exacte la réalité de distribution des pixels. Un autre point de vue consiste à faire la coopération entre l'algorithme FCM (Fuzzy C-Means) et l'algorithme possibiliste PCM (Possibilistic C-means) pour tirer parti des avantages de chacun. Pour l'évaluation des résultats des trois algorithmes de segmentation flous implémentés on utilise le critère d'uniformité. Ce critère simple est basé sur la somme des variances des régions. Les résultats obtenus par les trois algorithmes sont très satisfaisantes en termes du critère d'uniformité et le meilleur c'est l'algorithme de coopération FCM-PCM qui possède les avantages des deux algorithmes.

Mot clef :

Classification, sous-ensembles flous, classification floue non supervisée, c-moyennes floues, c-moyennes possibilistes.

Summary :

This work presents a system for segmentation of images and its performance, this system is based on segmentation by unsupervised algorithms fuzzy and possibility known as fuzzy c-means (FCM) and possibilistic c-means (PCM) classification. The images using in our system are BMP format because it is the simplest, most standard and most widely, the goal of this system is to segment the images by FCM with models the uncertainty and imprecision, and the PCM which generates a degree of membership absolute accurately reflect reality pixel distribution. Another point of view is to make cooperation between the FCM (Fuzzy C-Means) algorithm and PCM (Possibilistic C-Means) to reap the benefits of each. To evaluate the results of the three algorithms implemented of fuzzy segmentation we use the uniformity criterion. This simple criterion based on the sum of the variances of region. The results obtained by the three algorithms are very satisfactory in terms of the criterion of uniformity and the best is the algorithm of cooperation FCM-PCM which combines the advantages of both algorithms.

Key words:

Clustering, fuzzy sets, fuzzy clustering, fuzzy c-means (FCM), possibilistic c-means (PCM).

المخلص:

العمل التالي يعرض نظاما لتجزئة الصور واهميته، ويستند هذا النظام على اساس التجزئة بواسطة الخوارزميات الضبابية و الخوارزميات الممكنة، هذه الخوارزميات معروفة بالمتوسطات الممكنة، في نظامنا هذا استعملنا الصور ذات النوع BMP لأنها ابسط واكثر القياسية والاكثر استخداما على نطاق واسع، الهدف من هذا النظام هو تجزئة الصورة عن طريق خوارزمية المتوسطات الضبابية التي تحسب درجة الانتماء بطريقة نسبية، وخوارزمية المتوسطات الممكنة للبيكسل المبعثرة. وجهة نظر اخرى هو التنسيق بين خوارزمية المتوسطات الضبابية والممكنة للحصول على خوارزمية لها فوائد الاثنين معا. ولتقييم نتائج الخوارزميات الثلاثة استخدمنا معيار التماثل، يستند هذا المعيار على مجموعة تباينات المناطق، النتائج المتحصل عليها عن طريق الخوارزميات الثلاثة مرضية جدا من حيث معيار التماثل والافضل هو خوارزمية التنسيق بين الخوارزميتين المتوسطات الضبابية والمتوسطات الممكنة الذي يجمع بين مزايا كل واحدة.

الكلمات المفتاح:

التجميع، الضبابية، التجميع الضبابي، المتوسطات الضبابية، المتوسطات الممكنة.