

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A.MIRA de Bejaia

Faculté de la Technologie

Département d'Electronique

Mémoire

Présenté par : **MEKHMOUKH Abdenour**

Pour obtenir le Diplôme de Magister

Spécialité : Automatique et Traitement du Signal (ATS)

Option : Systèmes

Thème

*Segmentation d'images par classifieurs non supervisés :
Application à l'Imagerie par Résonance Magnétique
(IRM).*

Sous la direction de Mr : ***K.MOKRANI***

Soutenue publiquement le 27/05/2008

devant le jury composé de :

Mr. ZBOUDJ YUCEF	Professeur à l'Université de Bejaia , <i>Président du jury</i>
Mr. MOKRANI KARIM	Maitre de Conférences à l'Université de Bejaia , <i>Rapporteur</i>
Mr. CHIKOUCHE DJAMEL	Professeur à l'Université de Setif , <i>Examineur</i>
Mr. MENDIL BOUBEKEUR	Professeur à l'Université de Bejaia , <i>Examineur</i>

Mémoire préparé au sein du
Laboratoire de Technologie Industrielles et de l'Information (LTII).
(Bejaia, Route de Targua Ouzemour (06000), Algérie)

Remerciements

Je remercie Dieu , le tout puissant de m'avoir accordé santé , volonté , courage et patience qui m'ont été utiles tout le long de mon parcours.

Je tiens à remercier chaleureusement mon encadreur Mr. MOKRANI Karim pour sa rigueur scientifique, son esprit d'ouverture et sa disponibilité. Il a su m'encourager et me guider dans la bonne direction tous au long de ce mémoire.

Je remercie le président et les membres du jury pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail, et pour l'intérêt qu'ils ont porté à ce mémoire.

Merci également à ma famille et en particulier mes parents, pour l'encouragement constant et leur aide sans retenue tout au long de mes études.

Sans omettre mes collègue Abdelghani , Riad et Ahmed pour m'avoir soutenu et encouragé .

j'ai rencontré de nombreuses personnes que je ne citerai pas de peur d'en oublier , j'exprime toute ma gratitude à tous ceux qui ont participé de près ou de loin à la réussite de ce travail.

Table des matières

Remerciements	1
Liste des Figures	6
Liste des Algorithmes	7
Introduction Générale	8
1 Introduction à la segmentation d'images.	10
1.1 Notion d'image	11
1.2 Chaîne de traitement d'images	11
1.3 Définition de la segmentation d'images	13
1.4 Différentes approches	14
1.4.1 Approches contours	14
1.4.2 Approches régions	15
1.4.3 Approches classification	15
2 Segmentation par analyse des propriétés spatiales.	16
2.1 Introduction	17
2.2 Segmentation par division de régions	18
2.3 Segmentation par croissance de régions	19
2.3.1 Principe de la méthode LPE (Watershed)	20
2.3.2 LPE Approche Filtrage	22
2.3.2.1 Principe de la méthode	22
2.3.2.2 Diffusion Gaussienne (linéaire)	22
2.3.2.3 Filtre Alterné Séquentiel (morphologique)	23
2.3.2.4 Filtre non linéaire (Modèle de Malik et Perona)	24
2.4 Segmentation par fusion de régions	26
2.5 Segmentation par division-fusion	28
2.5.1 Tétra-arbre (Quadtree)	28
2.6 Conclusion	30
3 Segmentation par classifieurs non supervisés	31
3.1 Introduction	32
3.2 Généralités sur la classification	32
3.2.1 Classification	32

3.2.2	Méthodologie de la classification	33
3.2.2.1	Phase d'apprentissage	33
3.2.2.2	Phase de classement (ou de reconnaissance)	33
3.2.3	Différentes classifications	33
3.2.3.1	Classification supervisée	33
3.2.3.2	Classification non supervisée (clustering)	34
3.2.3.3	Classification avec apprentissage	34
3.2.3.4	Classification sans apprentissage	34
3.2.3.5	Paramétriques	34
3.2.3.6	Non paramétriques	34
3.2.4	Dissemblance et distance	34
3.3	Segmentation nette d'images par classification de pixels	36
3.3.1	Segmentation par la méthode des K-Means	36
3.3.1.1	Introduction	36
3.3.1.2	Principe de la méthode des K-Means [21]	36
3.3.1.3	Convergence et initialisation des K-Means	37
3.3.1.4	Choix du nombre K de classes (clusters)	37
3.3.1.5	Résultats et discussion	38
3.3.1.5.1	Initialisation des centres des classes	38
3.3.1.5.2	Choix du nombre K de classes	40
3.3.1.5.3	Immunité au bruit	41
3.3.2	Segmentation par la méthode "Mean Shift"	42
3.3.2.1	Estimation de la densité de probabilité	42
3.3.2.2	Algorithme de classification du Mean Shift [24]	44
3.3.2.3	Résultats et discussion	46
3.3.2.3.1	Influence du paramètre de lissage (h)	46
3.3.2.3.2	Immunité au bruit	47
3.4	Méthodes floues de classification	48
3.4.1	Principaux concepts de la logique floue	48
3.4.1.1	Définition d'un sous-ensemble flou	48
3.4.1.2	Caractéristiques d'un sous-ensemble flou	49
3.4.1.2.1	Support de A	49
3.4.1.2.2	Hauteur de A	49
3.4.1.2.3	Noyau de A	49
3.4.1.3	Exemple d'un sous-ensemble flou	49
3.4.2	Segmentation floue d'images par classification de pixels	50
3.4.2.1	C-Means (CM)	50
3.4.2.2	Fuzzy C-Means (FCM)	51
3.4.2.2.1	Analyse de l'algorithme FCM [33]	53
3.4.2.2.2	Paramètre de contrôle du flou (m)	53
3.4.2.2.3	Au niveau de l'initialisation	53
3.4.2.2.4	Forme des fonctions d'appartenance	54
3.4.2.2.5	Convergence du FCM[34]	55
3.4.2.2.6	Mise en évidence des Cols avec le test d'extremum	55
3.4.2.2.7	Résultats et discussion	56

3.4.2.2.7.1	Mise en évidence du problème de l'initialisation . . .	56
3.4.2.2.7.2	Fusion des classes	57
3.4.2.2.7.3	Influence du paramètre de fusion (fu)	60
3.4.2.2.7.4	Sensibilité du FCM au bruit	60
3.4.2.2.8	Avantages et inconvénients de l'algorithme (FCM)	62
3.4.2.3	C-Moyennes Possibilistes (Possibilistic C-Means)	62
3.4.2.3.1	Introduction	62
3.4.2.3.2	Principe général des PCM	62
3.4.2.3.3	Résultats et discussion	64
3.4.2.4	C-Moyennes Possibilistes Flous (Fuzzy Possibilistic C-Means (FPCM))	65
3.4.2.4.1	Résultats et discussion	67
3.5	Conclusion	68
4	Application à l'imagerie par résonance magnétique (IRM)	70
4.1	Introduction	71
4.1.1	Segmentation des tissus cérébraux (MB, MG, LCR)	71
4.1.1.1	Anatomie cérébrale [40]	71
4.1.1.1.1	Encéphale	71
4.1.1.1.2	Tissus du cerveau	72
4.1.1.1.2.1	Matière grise (MG)	72
4.1.1.1.2.2	Matière blanche (MB)	72
4.1.1.1.2.3	Liquide céphalo-rachidien (LCR)	72
4.1.1.2	Résultats et discussion	73
4.1.1.2.1	Présentation de la base de données utilisée (Brainweb) . . .	73
4.1.1.2.2	Résultats	73
4.1.1.2.2.1	Choix des paramètres des algorithmes utilisés	74
4.1.1.2.2.2	Image à 0% de bruit	74
4.1.1.2.2.3	Image à 3% de bruit	75
4.1.1.2.2.4	Image à 5% de bruit	76
4.1.1.2.2.5	Image à 9% de bruit	76
4.1.1.2.3	Amélioration du FPCM, PCM et FCM	78
4.1.1.2.4	Extraction des classes obtenues	79
	Conclusions et perspectives	81
	Annexe 1	83
	Annexe 2	86
	Bibliographie	93

Table des figures

1.1	Exemple d'une chaîne de traitement d'images sur une image aérienne.	12
1.2	Schéma d'un système imageur.	13
1.3	Exemple de segmentation par approche contour.	14
2.1	Image de synthèse.	18
2.2	Résultat de la segmentation par la méthode classique de division de régions.	19
2.3	Principe de la Ligne de Partage des Eaux.	20
2.4	Résultat de la segmentation par la LPE.	21
2.5	Résultat de la segmentation par la LPE (Diffusion Gaussienne).	22
2.6	Résultat de la segmentation par la LPE (Filtre Morphologique).	23
2.7	Résultat de la segmentation par la LPE (Filtre non linéaire).	25
2.8	Résultat de la segmentation par la méthode de fusion de régions	27
2.9	Structure pyramidale du Tétra-arbre.	28
2.10	Segmentation par division-Fusion.	29
2.11	Résultats de la segmentation par le Tétra-arbre.	29
3.1	Principe de la classification.	32
3.2	Exemple illustratif du déroulement de l'algorithme des K-Means.	37
3.3	Images synthétiques utilisées pour les tests (K-Means).	38
3.4	Influence de l'initialisation des centres des classes (Image synthétique 1)	38
3.5	Influence de l'initialisation des centres des classes (Image synthétique 2).	39
3.6	Résultats de la classification pour $K = 3$ et $K = 4$ (Image synthétique 1).	40
3.7	Résultats de la classification pour $K = 2 \dots 6$ (Image synthétique 2).	40
3.8	Résultat de la classification pour un bruit Gaussien ($m = 0, \sigma = 0,05$) (Image synthétique 1).	41
3.9	Résultat de la classification pour un bruit Gaussien ($m = 0, \sigma = 0,05$) (Image synthétique 2).	41
3.10	Exemples de fonctions noyaux utilisées.	44
3.11	Résultat de la classification des pixels de l'image synthétique 1 pour $h = 0,1 \dots 0,5$	46
3.12	Résultat de la classification des pixels de l'image synthétique 2 pour $h = 0,1 \dots 0,5$	46
3.13	Résultat de la classification des pixels de l'image synthétique 1 bruitée avec un bruit Gaussien ($m = 0, \sigma = 0,05$) et pour différentes valeurs de h	47
3.14	Résultat de la classification des pixels de l'image synthétique 2 bruitée avec un bruit Gaussien ($m = 0, \sigma = 0,05$) et pour différentes valeurs de h	47
3.15	Détail de l'image originale.	48

3.16	Support, Noyau, Hauteur d'un sous-ensemble flou.	49
3.17	Exemple d'un sous-ensemble flou.	50
3.18	Exemple illustratif du problème hyper-sphérique représentant les classes.	53
3.19	Fusion de formes sphériques élémentaires.	54
3.20	Mise en évidence du problème classification du FCM , la valeur de la fonction d'appartenance est différente pour x_A et x_B par rapport à C_1 alors que les points sont équidistants du centre de la classe.	55
3.21	Exemple de point selle.	55
3.22	Mise en évidence du problème de l'initialisation (Image synthétique 1)	56
3.23	Mise en évidence du problème de l'initialisation (Image synthétique 2)	56
3.24	Images synthétiques modifiées.	57
3.25	Représentation des classes dans l'espace RVB (Image synthétique 1)	58
3.26	Représentation des classes dans l'espace RVB en initialisant K=8 classes	58
3.27	Résultat de la segmentation initiale par les FCM	58
3.28	Résultat de la segmentation finale après fusion ($fu = 0,05$)	59
3.29	Résultats de la segmentation pour différente valeur du paramètre de fusion (fu)	60
3.30	Résultats de la segmentation pour un bruit Gaussien ($m = 0, \sigma = 0,001$) et ($fu = 0,34$)	61
3.31	Résultats de la segmentation pour un bruit Gaussien ($m = 0, \sigma = 0,05$) et ($fu = 0,34$).	61
3.32	Résultats de la segmentation par l'algorithme des PCM (Image synthétique 1)	64
3.33	Résultats de la segmentation par l'algorithme des PCM (Image synthétique 2)	65
3.34	Résultats de la segmentation par l'algorithme des FPCM (Image synthétique 1)	67
3.35	Résultats de la segmentation par l'algorithme des FPCM (Image synthétique 2)	67
4.1	Cerveau humain	72
4.2	Localisation du LCR.	73
4.3	Résultats obtenus sur une coupe axiale pondérée en T_1 à 0% de bruit	74
4.4	Résultats obtenus sur une coupe axiale pondérée en T_1 à 3% de bruit	75
4.5	Résultats obtenus sur une coupe axiale pondérée en T_1 à 5% de bruit	76
4.6	Résultats obtenus sur une coupe axiale pondérée en T_1 à 9% de bruit	76
4.7	Comparaison du taux d'erreur de classification (MCR) des algorithmes.	77
4.8	Chaîne de prétraitement de l'image.	78
4.9	Résultats obtenus sur une coupe axiale pondérée en T_1 à 9% de bruit en utilisant le prétraitement	78
4.10	Classification de référence (MB,MG,LCR) de la base de données Brainweb.	79
4.11	Classes obtenues (MB,MG,LCR) avec le FCM.	79
4.12	Classes obtenues (MB,MG,LCR) avec le PCM.	80
4.13	Classes obtenues (MB,MG,LCR) avec le FPCM.	80
4.14	Fusion de classes.	86
4.15	Mouvement de spin	88
4.16	Mouvement de précession d'un ensemble de moments magnétiques	89
4.17	Décomposition du moment magnétique M	90
4.18	Signal FID recueilli par les bobines réceptrices.	90
4.19	Plans sur une acquisition en T_1	91
4.20	Une même coupe obtenue sous différents paramètres d'acquisition.	92

Liste des algorithmes

1	Algorithme des K-Means	37
2	Algorithme des Mean-Shift, la procédure est répétée pour chaque $x \in X$	45
3	Algorithme général d'exécution du FCM	52
4	Algorithme général d'exécution du PCM	64
5	Algorithme général d'exécution du FPCM	66
6	Algorithme "Watershed" de Vincent-Soille	83
7	Algorithme "Watershed" de Vincent-Soille (Suite).	84

Introduction Générale

Au cours des dernières décennies, le domaine de la vision artificielle s'est énormément développé et de nombreux systèmes de vision sont de nos jours utilisés dans des domaines d'applications très variés tels que le contrôle qualité, la cartographie aérienne, le domaine médical, etc...

L'un des principaux problèmes de la vision artificielle : la segmentation d'images. Celle-ci consiste à localiser dans une image les régions (ensembles de pixels) appartenant à une même structure (objet ou scène imagée). Aujourd'hui, vu la simplicité de la définition de l'objectif de la segmentation d'images et la richesse des travaux effectués dans ce domaine, on pourrait penser que la segmentation d'images est un problème en grande partie résolu. Il n'en est rien, probablement à cause de la très grande diversité des besoins et des applications mais également parce que des approches mathématiques (analyse multirésolution, champs de Markov, classification floue,...) permettent de nouveaux développements.

La recherche d'une méthode performante, pour une application donnée, passe la plupart du temps par la comparaison de quelques méthodes disponibles et bien maîtrisées et par la modification d'une méthode existante afin de l'adapter. Beaucoup de méthodes de segmentation existent dans la littérature et peuvent être séparées en trois grandes familles :

- Les méthodes de segmentation par contours sont basées sur la recherche des discontinuités locales présentes dans l'image.
- Les méthodes de segmentation en régions homogènes consistent à trouver des ensembles de pixels qui partagent des propriétés similaires. Les régions homogènes sont construites à partir des parties connexes de ses ensembles.
- Les méthodes de classification fournissent une partition de l'image en regroupant des pixels ayant des niveaux de gris similaires dans une même classe de pixels. Les régions sont définies par les ensembles maximaux de pixels connexes appartenant à la même classe.

Les techniques de classification dites non supervisées issues de l'intelligence artificielle et de la logique floue (Fuzzy logic), des probabilités et des statistiques, connaissent un très grand succès proportionnel aux flux de données de plus en plus importants. Dans cette optique, nous nous proposons d'étudier différentes méthodes de classification et de les confronter afin de les appliquer sur des images médicales en vue d'une segmentation.

L'imagerie par résonance magnétique (IRM), qui constitue une des avancées les plus significatives en médecine, permet au praticien de faire un diagnostic et un suivi de plusieurs pathologies. Vu la taille des données à analyser, le recours aux techniques de traitement d'images peut offrir un outil d'aide au diagnostic. Pour cela, nous nous proposons d'appliquer les méthodes de seg-

mentation d'images par classification afin de segmenter des images cérébrales IRM en vue d'une classification des différentes régions correspondant aux différents tissus et structures.

Ce manuscrit est organisé en quatre chapitres :

Le premier chapitre introduit la segmentation d'images, différentes définitions que nous pouvons trouver dans la littérature y sont rappelées.

Le second chapitre est une revue des principales méthodes de segmentation d'images. L'objectif commun de l'ensemble des méthodes présentées est de découper les images en un ensemble de régions homogènes. Ces méthodes considèrent une région comme un ensemble de pixels connexes dont les niveaux de gris sont proches les uns des autres. Ces techniques procèdent par un balayage de l'image afin de construire des régions et font intervenir l'information spatiale pour la prise de décision.

Nous abordons ensuite quelques généralités dans le domaine de la classification pour converger vers la théorie des ensembles flous indéniablement liée aux fonctions d'appartenances , différentes méthodes de classification non supervisées sont proposées et comparées dans le troisième chapitre.

Dans le quatrième chapitre, le résultat de cette confrontation, nous permet de justifier le choix des méthodes que nous utilisons afin de segmenter des images cérébrales IRM en vue d'une classification des différentes régions correspondant aux différents tissus et structures.

Une conclusion générale nous permet de synthétiser les résultats obtenus, et d'envisager quelques perspectives.

Chapitre 1

Introduction à la segmentation d'images.

Dans ce premier chapitre, nous présentons dans un premier temps la notion d'image qui correspond au support sur lequel nous avons travaillé. Puis, dans un deuxième temps, une définition fonctionnelle communément admise de la segmentation au sein de la chaîne de traitement d'images est donnée. Ensuite, nous présentons un aperçu des différentes approches de segmentation que nous pouvons trouver dans la littérature.

1.1 Notion d'image

Avant de présenter en détail la chaîne de traitement d'une image, nous avons jugé nécessaire de définir la notion d'"image".

Sur le plan fonctionnel, la notion d'image est définie comme étant la "représentation d'une personne ou d'une chose par les arts graphiques ou plastiques, la photographie, le film". D'un point de vue plus technique, il s'agit alors d'un "ensemble de points lumineux formé par des rayons émanants, par réflexion ou par réfraction, des différents points d'un objet"[1].

Nous pouvons distinguer deux catégories principales d'images :

- Les collections de mesures physiques.
- Les images numériques.

Une image peut tout d'abord correspondre à une collection de mesures physiques d'un support donné $X = \mathbb{R}^n$. Par exemple, si $X = \mathbb{R}^2$, il peut s'agir de mesures physiques sur un support plan, ou de mesures d'un signal selon une direction avec sa donnée temporelle.

Afin d'exploiter ces collections de données sur ordinateur, il faut les stocker sous forme d'images numériques $2D$ qui correspondent à une structure de données informatique. Ainsi, les données contenues dans ces images vont permettre de représenter soit une discrétisation de mesures physiques pouvant être continues, soit le résultat d'un traitement quelconque d'une ou plusieurs autres images. On nomme chaque point d'une image numérique $2D$ un "pixel".

Ces images peuvent être le résultat d'une acquisition numérique directe tel que le ferait un appareil photo numérique avec un capteur CMOS ou CCD, ou encore un capteur aérien ou spatial comme SPOT comme présenté dans la figure (1.1) D'autres exemples d'images numériques $2D$ pourraient être obtenus à partir de photographies scannées provenant d'appareils photo argentiques ou encore des cartographies cérébrales de tissus obtenues par IRM.

1.2 Chaîne de traitement d'images

La notion d'image numérique $2D$ que nous venons de voir correspond au support étudié tout au long d'une chaîne de traitement d'images. La figure (1.1) présente un exemple de chaîne de traitement d'une image SPOT de la "forêt de Paimpont"[1].

Une chaîne de traitement d'images commence par l'acquisition des données physiques sous la forme d'une image. Ces données peuvent correspondre tout simplement à l'intensité lumineuse des éléments d'une scène ou bien à des données plus complexes comme des mesures de champs magnétiques ou autres... Ces données peuvent être acquises soit dans un espace quasi continu (comme des photos sur support pelliculaire par exemple) soit dans un espace discret (mesures par pas de temps ou d'espace). Dans le premier cas, une étape supplémentaire de numérisation de l'image $2D$ est alors nécessaire.

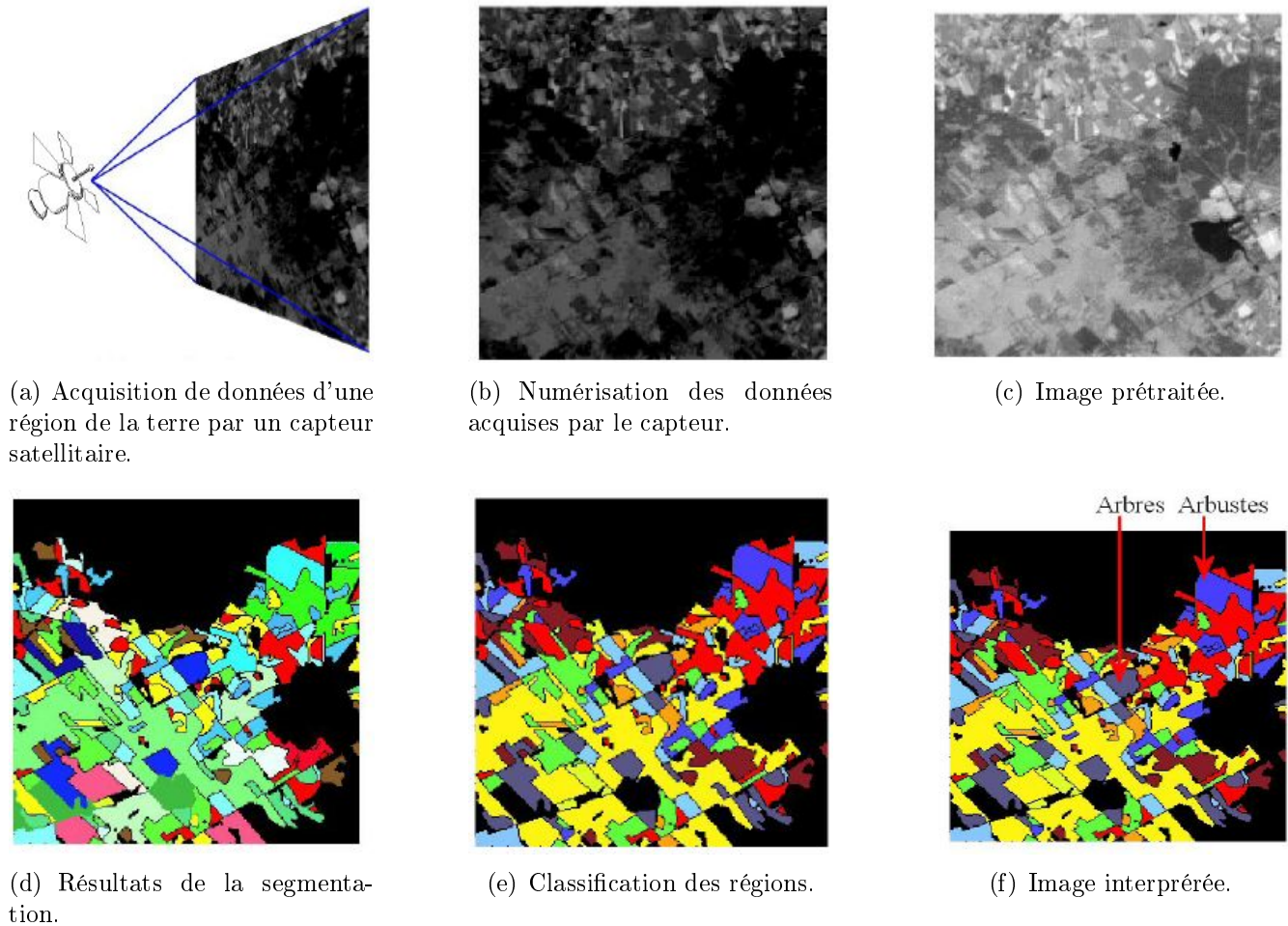


FIG. 1.1 – Exemple d'une chaîne de traitement d'images sur une image aérienne.

L'étape suivante concerne des prétraitements possibles afin d'améliorer la qualité de l'image acquise. Il peut s'agir par exemple de rehausser le contraste d'une image ou encore d'atténuer un bruit dû au capteur lors de l'acquisition de l'image. L'étape de segmentation consiste alors à partitionner une image en régions ou à détecter leurs frontières. Nous allons détailler par la suite cette étape.

Des post-traitements sont ensuite possibles. A ce niveau, diverses opérations peuvent être effectuées comme la classification d'un résultat de segmentation en régions afin de regrouper les régions ayant des statistiques voisines dans une même classe, ou encore la relaxation consistant à rendre le résultat de segmentation plus cohérent .

La dernière étape est l'interprétation qui consiste à étiqueter les régions détectées dans l'étape de segmentation. Cette étape nécessite l'utilisation de connaissances a priori.

1.3 Définition de la segmentation d'images

Il n'est pas aisé de trouver une seule définition de la segmentation car cette tâche est souvent confondue avec la classification ou l'étiquetage. Segmenter une image signifie trouver ses régions homogènes et ses contours. Ces régions et contours sont supposés très pertinents, c'est à dire que les régions doivent correspondre aux parties significatives des objets du monde réel, et les contours leurs contours apparents.

Une définition formelle d'un algorithme de segmentation a été donné par Horowitz et Pavlidis [2] en 1975. (Voir chapitre 2)

Les psychophysiciens et les gestaltistes sont d'accord sur le fait qu'un processus de segmentation se produit aux toutes premières étapes du processus de perception visuelle [3]. De plus, ils ont prouvé que ces premières étapes sont tout a fait indépendantes d'un quelconque apprentissage ou d'une connaissance a priori sur le monde (le domaine de l'image). Cela signifie qu'un algorithme programmé, traitant des images numériques, est capable de faire la même chose.

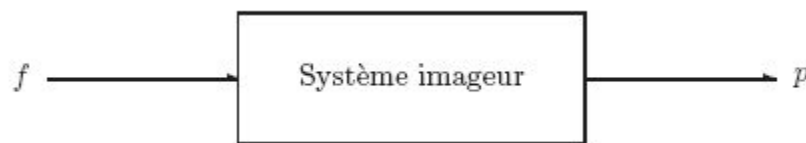


FIG. 1.2 – Schéma d'un système imageur.

L'image d'origine (notée f) est relié à l'image observée (notée p) par une transformation qui fait intervenir deux types de phénomènes.

Les premiers sont déterministes (projection, défauts du système imageur, etc...) et les seconds sont aléatoires, connus uniquement en termes de statistiques (bruit de mesure).

Le terme "problème inverse" désigne les problèmes consistant à inverser cette transformation : retrouver l'image f à partir de p , ce qui n'est pas toujours possible en pratique. En effet, la plupart des problèmes inverses en traitement d'image sont par nature mal posés.

Au sens d'Hadamard [4], un problème est mal posé si une des conditions suivantes n'est pas respectée :

1. Il existe une solution au problème.
2. Cette solution est unique.
3. Cette solution dépend continûment des données.

La seule connaissance des données observées est insuffisante pour assurer l'existence, l'unicité et la stabilité d'une solution. En général, on introduit un a priori sur la solution, que l'on appelle régularisation.

La segmentation consiste donc à trouver une approximation de f suffisamment correcte pour effectuer les phases de travail en aval.

1.4 Différentes approches

La segmentation est un vaste sujet d'étude et fait partie des grands thèmes de l'imagerie numérique. A ce titre, de nombreuses publications font état de segmentations. Comment préférer l'une ou l'autre est un débat ouvert qui fait rage dans bien des laboratoires.

En effet, pour valider correctement une segmentation d'objets naturels, comme en imagerie médicale, il faut disposer de la vérité terrain ; ce qui est bien difficile dans le cas de la segmentation, car comment définir de façon précise où commencent et où s'arrêtent les objets sur une image ? Il n'y a donc pas une mais des segmentations possibles sur une même image et elles sont bien souvent subjectives. De même, selon ce que nous voulons segmenter, certaines techniques seront plus à même d'y parvenir.

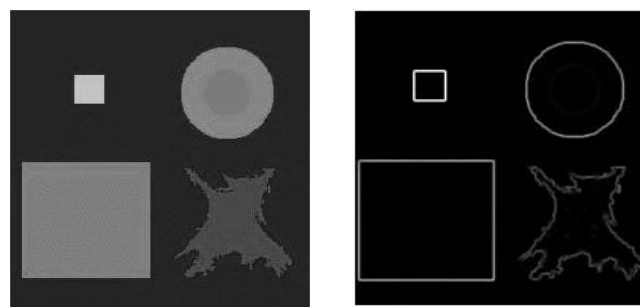
Beaucoup de méthodes de segmentation existent dans la littérature et peuvent être séparées en trois grandes familles.

1. Les méthodes de segmentation par contours.
2. Les méthodes de segmentation en régions homogènes.
3. Les méthodes de classification.

1.4.1 Approches contours

Les approches frontières détectent les zones de fortes discontinuités de l'image. C'est la frontière des objets qui permet de définir les objets. Les contours des objets dans les images sont localisés par les maxima locaux du gradient ou les zéros du Laplacien de la fonction de niveaux de gris. Les deux approches fournissent des contours sensiblement équivalents bien qu'elles ne localisent pas tout à fait les mêmes discontinuités.

Les difficultés de la détection de contours proviennent du bruit présent dans les images, car il perturbe fortement le gradient. Les approches contours n'aboutissent pas directement à une segmentation, car les contours obtenus ne sont généralement pas connexes et fermés. On utilise alors des techniques de fermeture de contours qui sont utilisées en aval de la détection de contours. Pour éviter ce type d'inconvénients, d'autres approches plus sophistiquées ont été développées et regroupées sous le nom de modèles déformables. Le principe général consiste à déformer une courbe ou une surface afin de l'adapter au contour de l'objet à segmenter.



(a) Image originale.

(b) Contours obtenus.

FIG. 1.3 – Exemple de segmentation par approche contour.

1.4.2 Approches régions

Les approches régions (duales des approches frontières) ont pour but de mettre en évidence les régions homogènes de l'image. Il s'agit de rechercher des ensembles de pixels partageant des propriétés communes. Les régions sont différenciées entre elles par des propriétés élémentaires basées sur des critères locaux tels que le niveau de gris de chaque pixel, ou bien sur un attribut estimé dans le voisinage du pixel tel que la valeur moyenne, la variance ou des paramètres de texture. L'ensemble des regroupements de pixels constitue une segmentation de l'image. Nous nous intéressons plus loin et particulièrement aux méthodes par analyse des propriétés spatiales (**Chapitre 2**).

1.4.3 Approches classification

Les méthodes par classification ont pour but de partitionner les images en plusieurs classes (comme leur nom l'indique) en regroupant des pixels ayant des niveaux de gris similaires dans une même classe de pixels. Les régions sont définies par les ensembles maximaux de pixels connexes appartenant à la même classe. Cependant, leur utilisation dans les méthodes de segmentation étant très répandue, il nous a paru judicieux d'en expliquer les ressorts. Nous verrons plus particulièrement et en détails les méthodes de classification non supervisées dans les prochains chapitres (**Chapitre 3**).

Chapitre 2

Segmentation par analyse des propriétés spatiales.



Contrairement à la segmentation par contours dont le principe est la recherche des points essentiels qui donnent la forme des objets composant l'image, la segmentation en régions consiste à décomposer l'image en des régions homogènes. Dans ce chapitre, nous aborderons les techniques utilisées pour aboutir à une segmentation de l'image en régions homogènes telle que une région soit composée de l'ensemble des pixels connexes possédant les mêmes propriétés au sens d'un prédicat d'homogénéité donné.

2.1 Introduction

La segmentation d'images en niveaux de gris ou couleur en régions est un traitement de bas niveau qui consiste à créer une partition d'une image I en sous-ensembles R_i appelés régions [5]. Chaque région doit respecter des critères d'homogénéité et de connexité.

L'homogénéité d'une région R_i est définie par un prédicat d'uniformité noté $Pred(R_i)$. Ce prédicat est vrai si R_i est homogène, faux dans le cas contraire.

Les régions doivent respecter les quatre conditions suivantes :

1. $I = \bigcup_i R_i$.
 2. R_i est constituée de pixels connexes pour tout i .
 3. $Pred(R_i) = \text{vrai}$ pour tout i .
 4. $Pred(R_i \cup R_j) = \text{faux}$ pour tout $i \neq j$, R_i et R_j étant adjacents dans I .
- La première condition implique que chaque pixel de l'image doit appartenir à une région R_i et l'union de toutes les régions correspond à l'image entière.
 - La deuxième condition est relative à la structure des régions. Elle définit une région comme un ensemble de pixels qui doivent être connexes.
 - La troisième condition exprime que chaque région doit respecter un prédicat d'uniformité.
 - La dernière condition implique la non-réalisation de ce même prédicat pour la réunion de deux régions adjacentes.

La vérification de ces conditions est une condition nécessaire et suffisante pour qu'une partition d'une image soit une segmentation. Rien, toutefois, n'implique l'unicité de cette segmentation. D'une part le nombre de régions obtenues est variable en fonction des prédicats d'homogénéité ainsi qu'aux seuils de tolérance contrôlant leurs formations. D'autre part ces critères d'homogénéité doivent s'adapter, à la nature de l'image à segmenter.

Dans ce type de méthodes de segmentation, en plus de l'information niveau de gris (ou couleur), l'information spatiale est automatiquement intégrée dans le processus de segmentation pour la formation de régions connexes et homogènes au sens d'un prédicat prédéfini [5].

Nous pouvons distinguer quatre approches :

1. Segmentation par division de régions.
2. Segmentation par croissance de régions.
3. Segmentation par fusion de régions.
4. Segmentation par division-fusion de régions.

Nous décrivons dans les paragraphes suivants chacune de ces approches, ainsi que les résultats obtenus sur une image de synthèse.

2.2 Segmentation par division de régions

L'approche segmentation par division de régions consiste à diviser l'image originale en régions homogènes au sens d'un critère donné. Ce processus est récursif et considère que la région initiale correspond à l'image à analyser. Si une région ne respecte pas un prédicat d'homogénéité, elle est divisée en quatre sous-régions de taille égale. Chaque sous-région est ensuite analysée. L'algorithme récursif s'arrête lorsque toutes les régions respectent le prédicat d'homogénéité [6].

A cause des divisions en quatre des régions, cette méthode est plutôt adaptée à des images carrées ayant un nombre de lignes et de colonnes égal à une puissance de deux, et dans lesquelles les régions sont de forme rectangulaire. D'autre part, cette méthode a tendance à faire apparaître des effets de blocs.

Pour illustrer cette approche, nous nous proposons de segmenter l'image de synthèse (en niveaux de gris) de la figure 2.1 par division de régions.

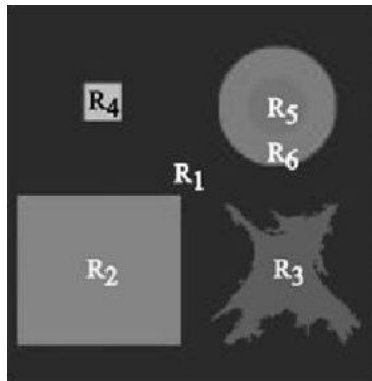


FIG. 2.1 – Image de synthèse.

Dans l'image de synthèse de la figure 2.1, nous distinguons six régions qui sont :

- Un fond R_1 .
- Un grand rectangle R_2 .
- Une forme irrégulière R_3 .
- Un petit carré R_4 .

Un disque composé de deux disques concentriques :

- Un disque central R_5 .
- Un anneau périphérique R_6 .

Cette méthode consiste à diviser l'image, qui constitue la région initiale, en régions de plus en plus homogènes. Le processus est réitéré pour chacune des régions produites jusqu'à ce qu'une certaine homogénéité soit atteinte .

Nous considérons qu'une région doit être divisée si sa variance est supérieure à un certain seuil. La figure (2.4.b) montre le résultat obtenu avec un seuil fixé.

Le résultat de la segmentation dépend essentiellement de ce seuil qui est difficilement ajustable.

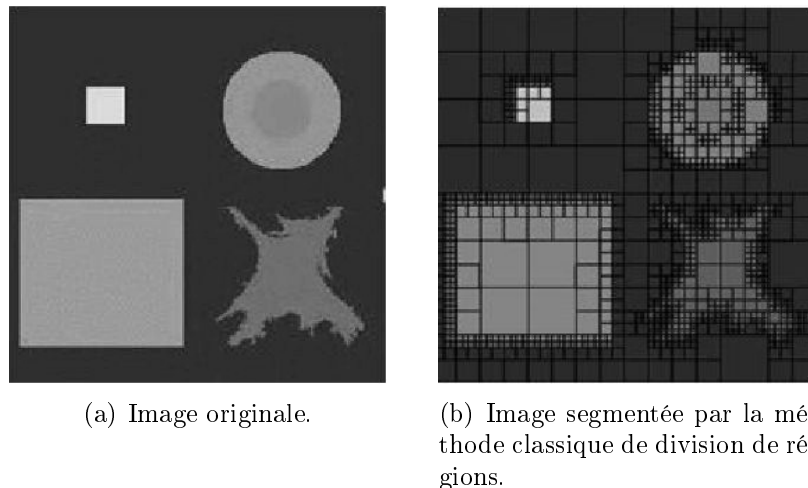


FIG. 2.2 – Résultat de la segmentation par la méthode classique de division de régions.

2.3 Segmentation par croissance de régions

Ce type de segmentation consiste à faire croître des régions en y ajoutant successivement les pixels adjacents qui satisfont un critère d'homogénéité. La croissance s'arrête lorsque tous les pixels ont été traités.

L'étape initiale consiste à sélectionner les germes des régions qui correspondent généralement à un pixel. Puis, les régions sont construites en y ajoutant successivement les pixels qui leur sont connexes et qui vérifient un critère de similarité . La plupart des algorithmes de croissance de régions sont régis par un parcours de l'image selon le balayage de gauche à droite et de haut en bas [7].

L'avantage de la croissance de régions est de préserver la forme de chaque région de l'image. Cependant, une mauvaise sélection des germes ou un choix du critère de similarité mal adapté peuvent entraîner des phénomènes de sous-segmentation ou de sur-segmentation.

A. Trémeau et N. Borel [7] proposent un algorithme de segmentation qui combine une croissance de régions suivie d'un processus de fusion de régions. Cet algorithme procède par un balayage séquentiel de l'image et considère le premier pixel comme un germe. Il tente alors de faire croître ce germe le plus longtemps possible en y agrégeant les pixels voisins.

Cette méthode repose sur plusieurs critères prenant en compte la proximité aux sens spatial et colorimétrique des régions adjacentes. La méthode proposée nécessite le réglage de nombreux seuils, parmi lesquels :

- S_{LHC} : seuil du critère d'homogénéité local. Ce seuil est utilisé pour garantir que deux pixels voisins appartenant à une même région ont des niveaux de gris proches.
- S_{AHC1} : seuil du critère d'homogénéité moyen. Ce seuil est utilisé pour s'assurer que le niveau de gris d'un pixel ajouté à une région soit proche du niveau de gris moyen de ses pixels voisins appartenant à cette région.
- A_P : nombre minimal de pixels voisins nécessaire pour qu'un pixel soit ajouté à une région. Un pixel doit avoir au moins A_P pixels voisins appartenant à une région pour qu'il puisse y être ajouté.

L'approche présentée fournit des résultats dépendant du sens de balayage de l'image. Pour éviter ce problème, L. Shafarenko et M. Petrou [8] proposent une méthode de croissance de régions topographique qui est fondée sur la ligne de partage des eaux (LPE).

2.3.1 Principe de la méthode LPE (Watershed)

La ligne de partage des eaux (LPE) utilise la description des images en termes géographiques. Une image peut en effet être perçue comme un relief si l'on associe le niveau de gris de chaque point à une altitude (figure 2.3). Il est alors possible de définir la ligne de partage des eaux comme étant la crête formant la limite entre deux bassins versants

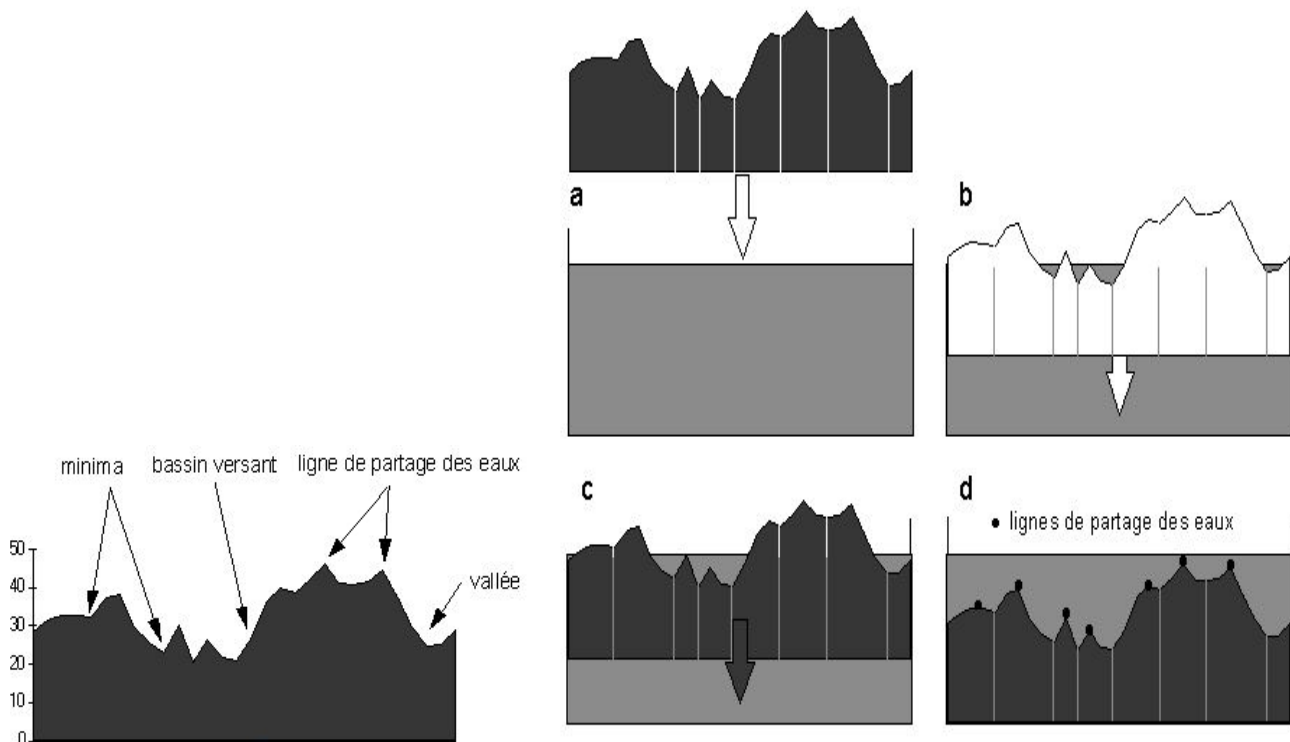


FIG. 2.3 – Principe de la Ligne de Partage des Eaux.

Pour l'obtenir, il faut imaginer l'immersion d'un relief dans de l'eau, en précisant que l'eau ne peut pénétrer dans les vallées que par ses minima. La ligne de partage des eaux est représentée par les points où deux lacs disjoints se rejoignent au cours de l'immersion.

Il est à noter que la ligne de partage des eaux n'est pas locale. En particulier, elle n'est pas liée à des structures locales (lignes de crête, escarpement, . . .). La LPE n'est pas un concept local. Nous ne pouvons pas, avec la seule connaissance locale du voisinage d'un point, répondre à la question : ce point appartient-il à une ligne de partage des eaux ?

Une des difficultés à la mise en œuvre de cette analogie intuitive est qu'elle laisse beaucoup de liberté quant à sa formalisation. Il existe deux classes principales d'implémentation : l'une est basée sur un algorithme récursif d'immersion [9] et une autre basée sur l'utilisation de fonctions de distances géodésiques [10]. Nous utiliserons l'algorithme de Vincent – Soille (voir Annexe 1) pour illustrer cette méthode :

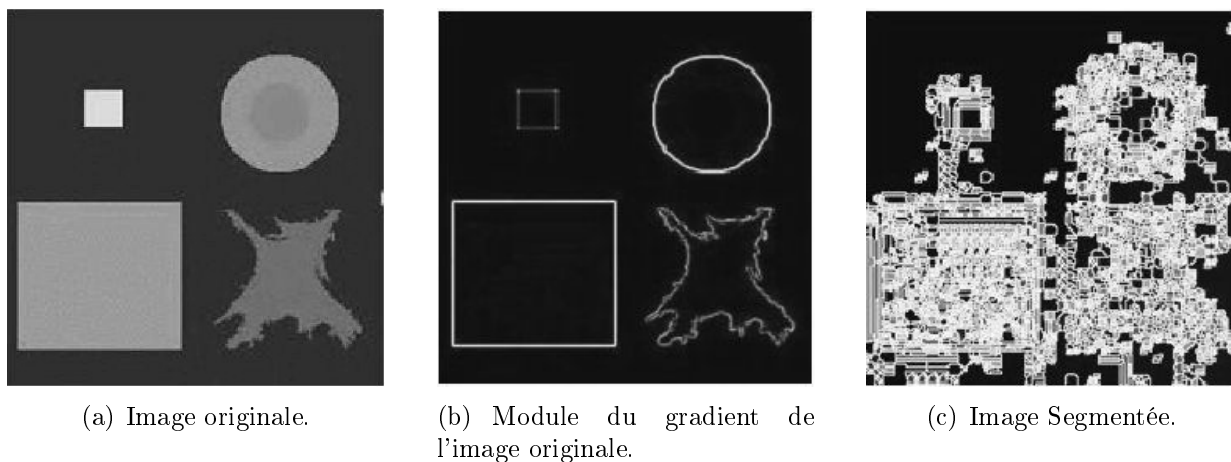


FIG. 2.4 – Résultat de la segmentation par la LPE.

La LPE du gradient est sur-segmentée (figure (2.4.c)). Ceci est dû au fait que les images gradient des images naturelles contiennent un grand nombre de minima (dû au bruit du système de prise de vue par exemple ou à variations locales non significatives des niveaux de gris ou encore aux textures des régions). Chaque minimum génère un bassin versant dans la LPE.

Il existe plusieurs méthodes qui permettent de pallier à cette sur-segmentation, nous pouvons citer :

1. La méthode qui permet de supprimer tous les minima non significatifs en filtrant l'image originale : c'est l'approche filtrage (que nous allons décrire dans le paragraphe suivant).
2. La méthode qui permet de choisir le nombre de minima locaux et donc le nombre de zones que l'on souhaite mettre en évidence grâce à la LPE : c'est l'approche marqueurs (swamping).

2.3.2 LPE Approche Filtrage

2.3.2.1 Principe de la méthode

Dans cette approche nous cherchons à éliminer tous les minima non significatifs de l'image gradient. En ce sens nous cherchons donc d'une certaine manière à moyennner localement l'image. Alors qu'avec l'approche marqueur nous remplaçons les minima du gradient par des images de marquages, ici nous allons plutôt chercher à supprimer les minima non-significatifs dans l'image elle-même

2.3.2.2 Diffusion Gaussienne (linéaire)

Nous avons vu que l'une des principales sources du phénomène de sur-segmentation est le bruit présent dans les images naturelles. Afin de pallier à l'effet du bruit, un moyennage spatial à l'aide d'un filtre gaussien est effectué.

Soient $U(x)$ le niveau de gris en un point x de l'image à traiter et G_σ la gaussienne d'écart type σ donnée par :

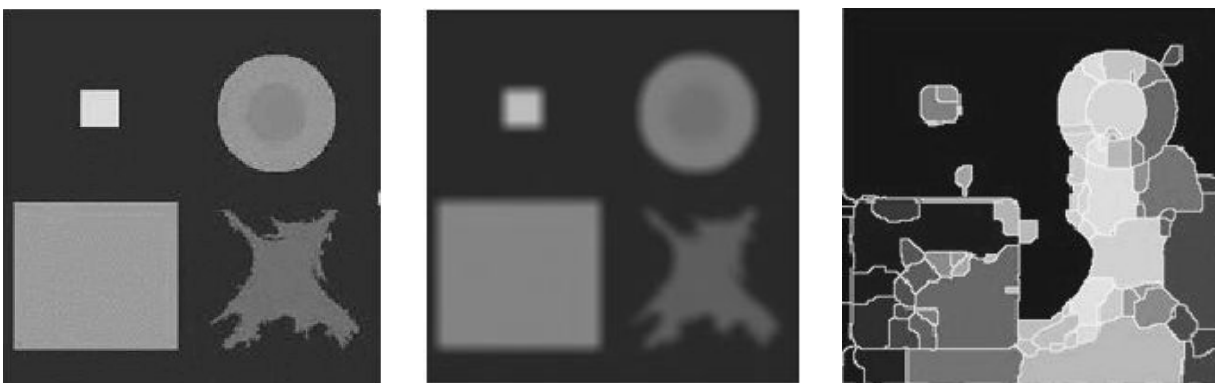
$$G_\sigma = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|x|^2}{2\sigma^2}\right) \quad (2.1)$$

Le filtrage gaussien de l'image résulte de la convolution de la fonction $U(x)$ avec des gaussiennes en chaque point de l'image :

$$U(x, \sigma) = (G_\sigma U_0)(x) = \int_{\mathbb{R}^2} G_\sigma(x - y) U_0(y) dy \quad (2.2)$$

Cette convolution est une opération régularisante. En pratique, cette régularisation de U permet de lisser de manière grossière, en affaiblissant l'information qui présente des variations spatiales sur des échelles inférieures à σ . Un inconvénient évident du filtrage gaussien est qu'il ne lisse pas uniquement le bruit, mais il gomme aussi les contours, les rendant difficilement identifiables.

Observons l'influence d'un tel filtrage sur l'image de la figure (2.1) qui est lissée avec une gaussienne de dimension 10 pixels et d'écart-type de 10.



(a) Image originale.

(b) Image lissée.

(c) LPE sur l'image lissée.

FIG. 2.5 – Résultat de la segmentation par la LPE (Diffusion Gaussienne).

La détection de contours obtenue reste très sur-segmentée (figure(2.5.c)), même s'il existe une amélioration quand le gradient est calculé sur l'image lissée. Le gradient est toujours meilleur s'il est calculé sur une image lissée par diffusion linéaire.

2.3.2.3 Filtre Alterné Séquentiel (morphologique)

Ce filtre est un filtre morphologique qui consiste à faire une succession d'ouvertures et de fermetures par des sphères de rayons croissants. Plus précisément, en désignant par I l'image considérée, la boucle itérée de base du Filtre Alterné Séquentiel (ASF) est définie par :

$$ASF(I) = Ouverture(fermeture(I)) \quad (2.3)$$

la première opération est une fermeture afin de privilégier la suppression des minima surnuméraires.

Les paramètres sont :

- la taille de l'élément structurant avec lequel on effectue les opérations morphologiques,
- le nombre d'itérations.

Nous utiliserons pour l'élément structurant du gradient un cercle de diamètre 5 pixels et un élément structurant de même type de taille initiale 3. Nous visualiserons les résultats sur l'image de la figure (2.1) :

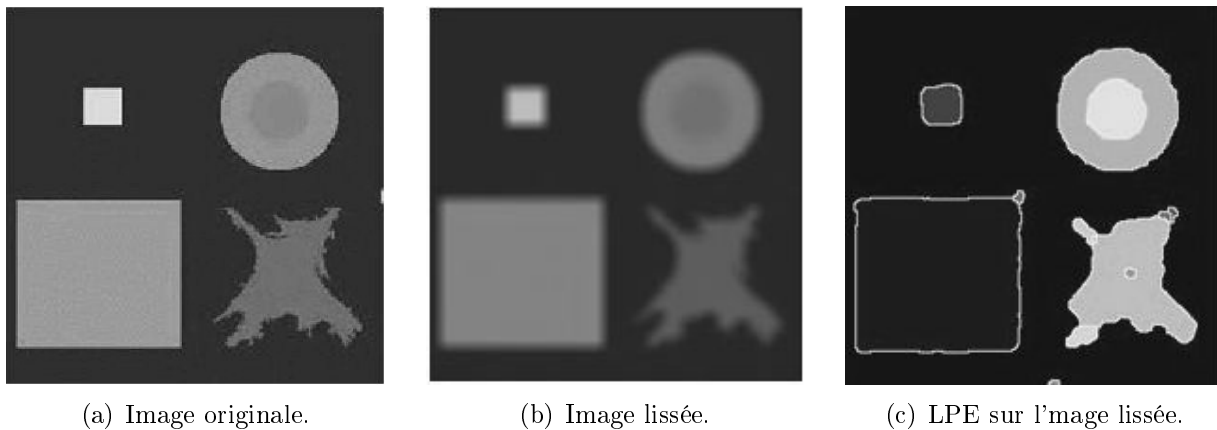


FIG. 2.6 – Résultat de la segmentation par la LPE (Filtre Morphologique).

Ce filtrage réduit sensiblement la sur-segmentation originelle mais ses performances restent néanmoins en deçà des autres prétraitements déjà testés.

2.3.2.4 Filtre non linéaire (Modèle de Malik et Perona)

Pour résoudre les problèmes issus d'une diffusion isotrope, la première idée de diffusion anisotrope a été proposée par Malik et Perona [11]. La motivation essentielle de l'utilisation de modèles basés sur la diffusion et la construction d'un opérateur de diffusion dépendant des propriétés locales de l'image. L'idée est de créer un filtre qui conserve, voire rehausse les contours et qui floute les zones à faible gradient.

Le filtre de Malik et Perona est basé sur l'équation suivante :

$$\begin{cases} \frac{\partial U}{\partial t} = \text{div} (g |\nabla U|) \nabla U \\ U(x, y, t = 0) = U_0(x, y) \end{cases} \quad (2.4)$$

C'est une équation de diffusion qui serait standard s'il n'y avait pas la présence de la fonction g . L'idée véhiculée par cette fonction est que le traitement obtenu par l'équation précédente est conditionnel en chaque point de l'image. Cette fonction est définie de manière à ce que la diffusion soit forte dans les zones à faible gradient et faible dans les zones à fort gradient. La diffusion est conditionnelle[11].

La fonction g peut être définie de la manière suivante :

$$g(U) = \exp - \left(\frac{|\nabla U|^2}{k} \right) \quad (2.5)$$

ou bien :

$$g(U) = \frac{1}{1 + \left(\frac{|\nabla U|^2}{k} \right)} \quad (2.6)$$

Les caractéristiques du filtre linéaire de Malik et Perona sont :

- Le lissage non linéaire de Malik et Perona a l'avantage de constituer des zones plates dans l'image, même quand le contraste est faible.
- La fonction d'arrêt du processus contient le paramètre k , ce paramètre correspond à la hauteur des contours à préserver.
- Ce filtre diminue la sur-segmentation résultante de l'application de méthodes de détection de contours comme la LPE.
- Ce type de filtrage est particulièrement intéressant lorsque les images à analyser sont bruitées. En effet le bruit induit de multiples bassins versants non-significatifs. Un moyen simple de les supprimer est d'effectuer un filtrage passe-bas.

- Ce filtre a l'inconvénient d'atténuer les contours. C'est pourquoi un filtre non-linéaire est utilisé.

Nous visualiserons les résultats sur l'image de la figure (2.1) bruitée avec un bruit gaussien :

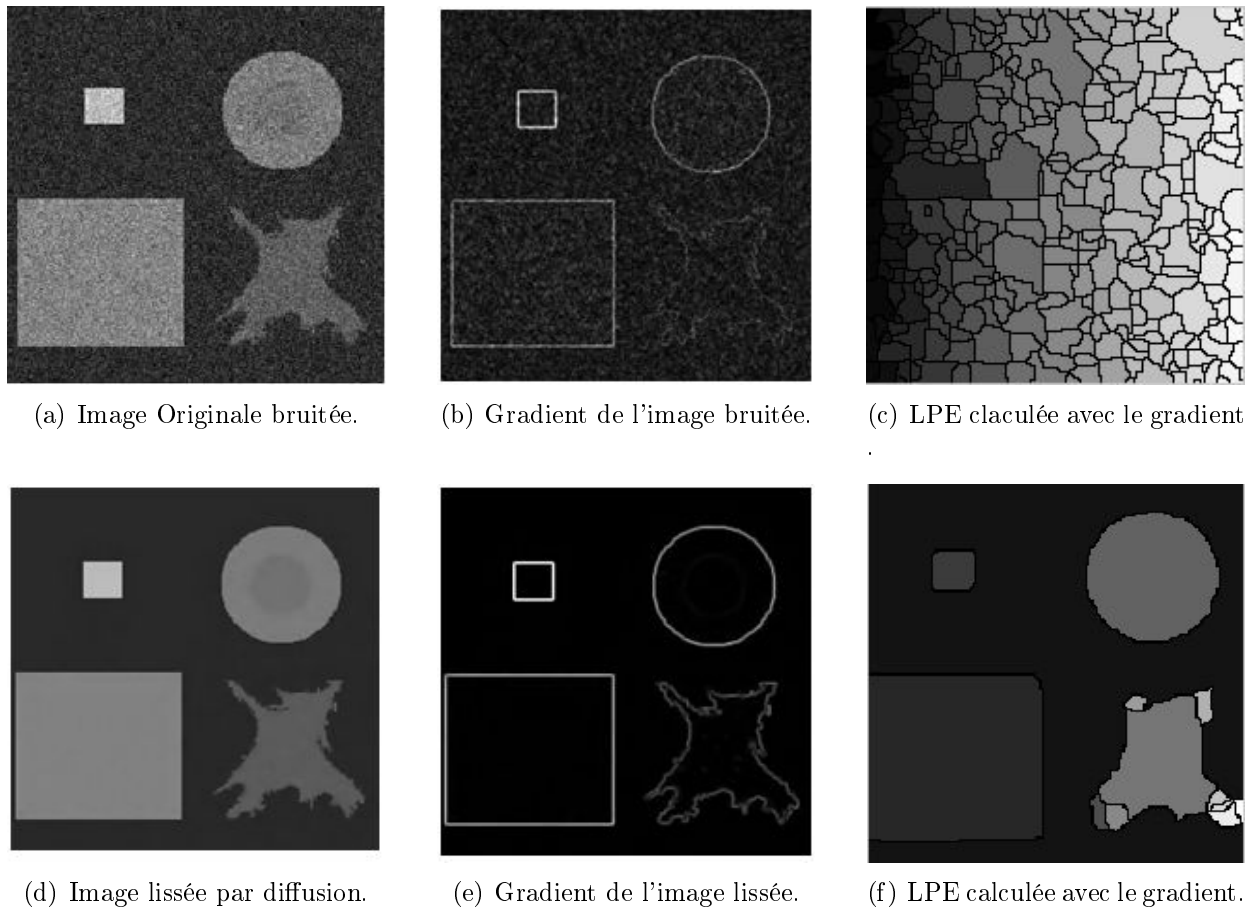


FIG. 2.7 – Résultat de la segmentation par la LPE (Filtre non linéaire).

Concernant l'image filtrée, nous observons que les différents contours ont bien été préservés, à l'exception des contours du disque central R_5 qui ont donc été diffusés (figure(2.7.e)). L'image gradient permet de vérifier que l'amplitude du gradient n'a pas diminué alors que l'effet du bruit a été supprimé. Ce qui peut se vérifier sur la LPE calculée avec le gradient de l'image lissée par diffusion non linéaire (figure (2.7.f)). La LPE demeure malgré tout très sur-segmentée.

2.4 Segmentation par fusion de régions

L'idée consiste à exploiter une partition initiale de l'image constituée de petite régions. Ces régions sont ensuite fusionnées successivement jusqu'à ce que le critère de fusion ne soit plus vérifié.

- Plusieurs règles de regroupement ont été proposées. Certaines de ces règles mettent en jeu :
- Des propriétés statistiques telle que la moyenne ou la variance des niveaux de gris ou couleurs des régions, le contraste maximum des régions, ou d'autres statistiques locales qui expriment l'état de surface des régions . . . etc.
 - Des propriétés géométriques ou morphologiques telle que l'élongation ou la compacité des régions.

A.Gagalowicz et O.Monga [12] suggèrent d'utiliser une succession de prédicats. Lorsque le prédicat courant n'est vrai pour aucune paire de régions on passe au suivant et ainsi de suite. Le seuil du prédicat courant est déterminé de manière à ce que les régions créées par les prédicats précédents vérifient celui-ci. L'utilisation de prédicats emboîtés présentent de nombreux avantages. Tout d'abord, elle permet de limiter le problème du choix du prédicat. En effet, il est difficile de choisir un critère pour fusionner les régions d'une image, ce critère étant bien souvent différent pour tous les couples de régions présents. Le choix d'un seule critère introduit donc des erreurs de segmentation. L'utilisation de plusieurs prédicats emboîtés permet de remédier à ce problème, ou tout au moins de le limiter .

Une difficulté de cette méthode réside néanmoins dans le choix des prédicats et de leurs emboîtements. En effet, quel prédicat favoriser par rapport à quel autre ? D'autre part, cette méthode introduit bien souvent un surcroît en temps de calcul par rapport aux méthodes de fusion classiques.

Parmi les critères d'uniformité, nous pouvons citer :

$$C_{minmax} = \|Max (R_i \cap R_j) - Min (R_i \cap R_j)\| \quad (2.7)$$

$$C_{moy} = \|Moy (R_i) - Moy (R_j)\| \quad (2.8)$$

$$C_{var} = Var (R_i \cup R_j) \quad (2.9)$$

$$C_{fr} = \frac{\sum_{((i,j),(k,l)) \in F(R_i, R_j)} \|I(i, j) - I(k, l)\|}{lf (R_i, R_j)} \quad (2.10)$$

Où pour une région $R_k = \{(x, y) / x = 1, \dots, n, y = 1, \dots, m\}$, les fonctions Max, Min, Moy et Var désignent respectivement le maximum, le minimum, la moyenne et la variance ; elles sont données par :

$$Max (R) = max_{(i,j) \in R} \{I(i, j)\} \quad (2.11)$$

$$Min (R) = min_{(i,j) \in R} \{I(i, j)\} \quad (2.12)$$

$$Moy(R) = \frac{1}{card(R)} \sum_{(i,j) \in R} I(i,j) \quad (2.13)$$

$$Var(R) = \frac{1}{card(R)} \sum_{(i,j) \in R} [I(i,j) - Moy(R)]^2 \quad (2.14)$$

dans l'équation (2.10) , $F(R_i, R_j)$ représente la frontière entre les régions R_i et R_j , elle est définie par l'ensemble des points (p, q) tels que :

$$p \in R_i, \quad q \in R_j \text{ est } d(p, q) = 1 \quad (2.15)$$

avec $d(.,.)$ une distance qui dans le cas d'un système de 4 voisinage est donnée par :

$$d_4(p, q) = |i - k| + |j - l| \text{ o } p = (i, j) \text{ et } q = (k, l) \quad (2.16)$$

$lf(R_i, R_j)$ est la longueur de la frontière et est égale à $car\{F(R_i, R_j)\}$.

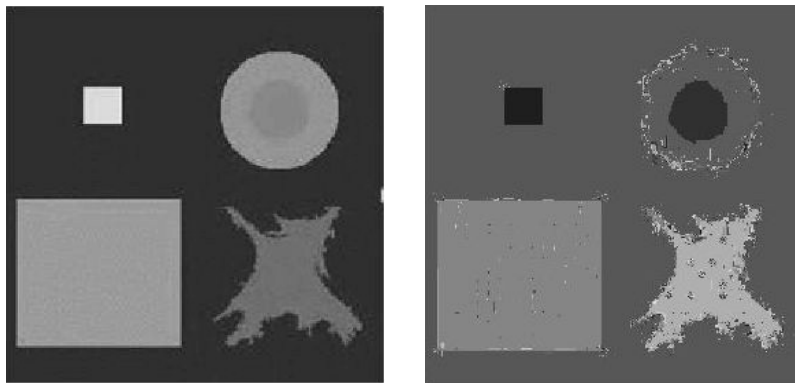
L'opérateur $card$ désigne la cardinalité d'un ensemble.

On peut également définir des critères combinant les critères simples précédents, par exemple :

$$C = aC_{minmax} + bC_{moy} \quad (2.17)$$

$$C = \frac{C_{moy}}{C_{Var}} \quad (2.18)$$

Afin d'illustrer cette méthode, nous utilisons comme critère d'uniformité l'équation (2.13) pour segmenter l'image de synthèse de la figure (2.1) :



(a) Image originale.

(b) Image segmentée en utilisant la moyenne comme critère d'uniformité.

FIG. 2.8 – Résultat de la segmentation par la méthode de fusion de régions .

2.5 Segmentation par division-fusion

La segmentation par division-fusion regroupe les deux types d'approches précédents. Tout d'abord, l'image est divisée en régions homogènes qui respectent des critères globaux, puis les régions adjacentes qui répondent à des critères locaux sont fusionnées.

Nous détaillons une structure de données permettant d'effectuer cette approche : le quadtree.

2.5.1 Tétra-arbre (Quadtree)

Ces méthodes combinent les deux méthodes décrites précédemment, la division de l'image en des petites régions homogènes, puis la fusion des régions connexes et similaires au sens d'un prédicat de regroupement. Horowitz et Pavlidis [13] sont les premiers à avoir proposé une telle approche de segmentations à partir de laquelle de nouvelles variantes sont apparues dans la littérature [14]. Le processus de segmentation utilise la structure pyramidale du Quadtree (figure(2.9)), et peut être décrit comme :

- En premier lieu chaque bloc associé à un noeud du Quadtree est analysé de façon récursive afin de décider s'il doit être divisé en quatre sous-blocs. L'analyse récursive s'arrête lorsque chaque sous-bloc respecte un prédicat d'homogénéité.
- Ensuite, à chaque fois que 4 sous-bloc satisfont un critère d'homogénéité ils sont regroupés à un niveau supérieur du Quadtree. La fusion continue tant qu'il est possible de le faire, c'est à dire tant que le critère d'homogénéité est satisfait. Lorsqu'il n'est plus possible de fusionner, il reste encore une étape pour examiner les blocs adjacents qui n'étaient pas au même niveau dans le Quadtree et les fusionner s'ils satisfont le critère d'homogénéité (figure (2.10)).

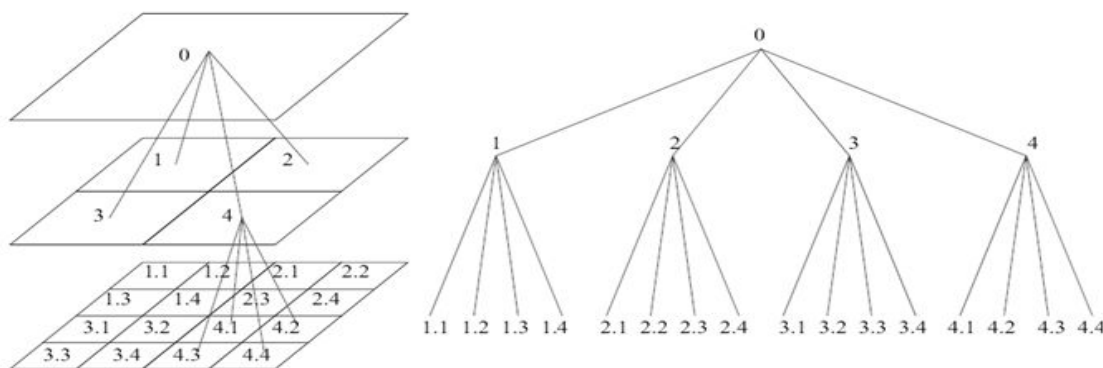


FIG. 2.9 – Structure pyramidale du Tétra-arbre.

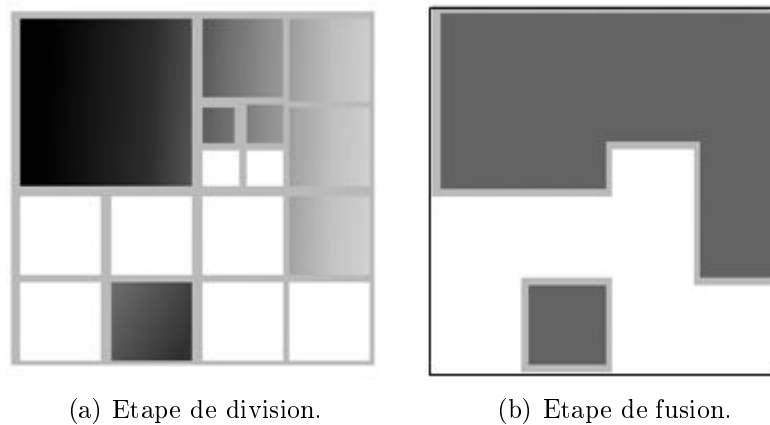


FIG. 2.10 – Segmentation par division-Fusion.

Nous visualiserons les résultats sur l'image de la figure (2.1) en utilisant cette approche :

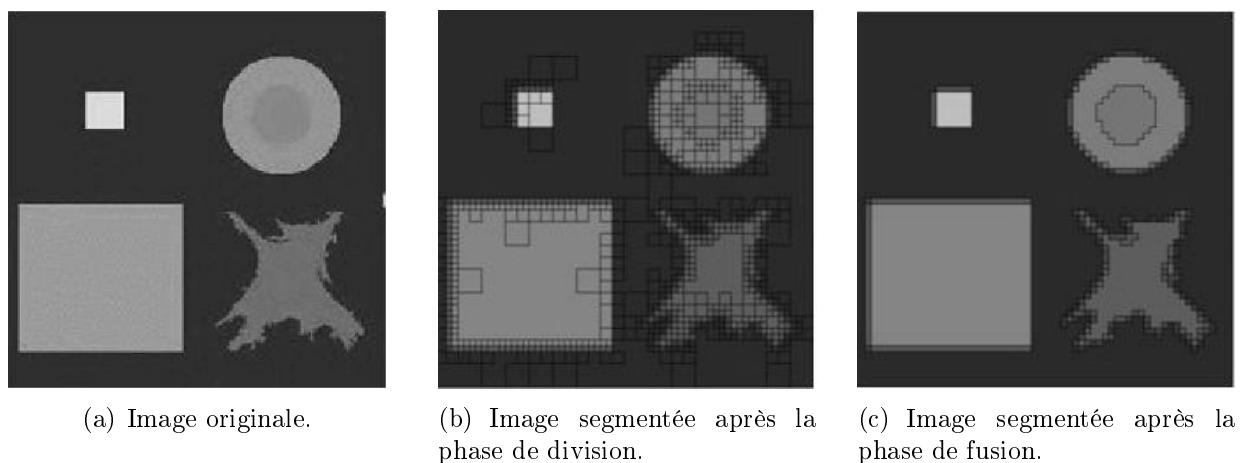


FIG. 2.11 – Résultats de la segmentation par le Tétra-arbre.

La figure (2.11.c) illustre le problème du découpage carré avec le résultat obtenu après la phase de division-fusion de l'image originale de la figure (2.11.a). Un bloc est divisé si la variance de niveaux de gris de ses pixels est supérieure à un certain seuil fixé. L'image de la figure (2.11.b) représente l'image originale de la figure (2.11.a) segmentée après la phase de division. Deux blocs adjacents dans l'image de la figure (2.11.b) fusionnent si le critère d'homogénéité est satisfait.

Le résultat final de segmentation obtenu après fusion des blocs est présenté dans l'image de la figure (2.11.c). Les contours des régions construites sont indiqués en noir sur les images des figures (2.11.b) et (2.11.c).

L'inconvénient majeur du quadtree réside dans la rigidité du découpage carré qu'il impose. Il conduit à un partitionnement global de l'image qui ne respecte pas toujours la forme des régions présentes dans l'image. Par ailleurs, la phase de regroupement des blocs est sensible à l'ordre du parcours du quadtree.

2.6 Conclusion

Dans ce chapitre, nous avons présenté les méthodes de segmentation en régions d'images qui considèrent une région comme un ensemble de pixels connexes respectant des critères d'homogénéité (intensité lumineuse, colorimétrie ...etc.).

Certaines méthodes nécessitent le réglage de nombreux seuils et paramètres, tels que des critères de similarité ou des critères d'arrêt. L'ajustement de ces paramètres est délicat, de telle sorte qu'il est difficile de prévoir si nous aboutirons à une sous-segmentation ou à une sur-segmentation de l'image.

Ces méthodes déterminent des régions dont les contours ne respectent pas toujours la forme des objets présents dans l'image, comme c'est le cas pour les effets de blocs. Par ailleurs, elles peuvent être sensibles aux changements d'échelle ou même à de simples translations des pixels. Enfin, il est important de noter que l'ordre d'agrégation des pixels ou l'ordre de fusion des régions a une influence sur les résultats de la segmentation.

Ces limites ont conduit de nombreux auteurs à privilégier une approche globale de l'image en appliquant des méthodes de classification de données multidimensionnelles. Cette approche fera l'objet du prochain chapitre.

Chapitre 3

Segmentation par classifieurs non supervisés

L

es méthodes de segmentation d'images par classification sont issues du domaine de classification des données multidimensionnelles. Elles définissent une partition de l'image en un ensemble de classes telle que chaque classe rassemble des pixels ayant des vecteurs de caractéristiques (couleurs ou bien souvent réduit au niveau de gris) aussi similaires que possible et que les classes soient aussi distinctes que possible les unes des autres. Généralement, ces méthodes de segmentation ne prennent pas en compte la disposition spatiale des pixels et ne considèrent que le vecteur de caractéristiques utilisé. Elles identifient les classes de pixels présentes dans l'image et affectent à chaque pixel une étiquette indiquant la classe à laquelle il appartient. Ainsi les pixels appartenant à une classe peuvent former plusieurs régions non adjacentes dans l'image mais partageant les mêmes propriétés statistiques. Une segmentation en régions n'est obtenue qu'après analyse de la connexité des pixels dans l'image étiquetée.

Dans ce chapitre, nous nous intéressons dans un premier lieu à la segmentation nette d'images puis nous introduisons les méthodes de classification floue dont le principe repose sur les concepts de la logique floue afin de palier aux limites posées par la classification nette. Des résultats sont présentés en appliquant ces méthodes sur deux images synthétiques.

3.1 Introduction

Les méthodes de classification permettent de regrouper des objets en groupes ou classes d'objets plus homogènes. Les objets regroupés ont des caractéristiques communes, ils sont similaires, mais se distinguent clairement des objets des autres classes. Les éléments classés sont les pixels de l'image qui nous permettent d'avoir des groupes. Généralement, les méthodes de classification conduisent à des résultats plus ou moins différents.

L'éventail de techniques de classification est assez large, il s'avère important de confronter les méthodes existantes afin de trouver celles qui s'adaptent le mieux à notre étude. Afin de déterminer l'appartenance d'un pixel à une classe, nous n'utiliserons que les méthodes de classification non supervisées :

1. Le K-Means.
2. Le Mean-Shift.
3. L'algorithme des C-Moyennes Floues (Fuzzy C-Means) pour et la classification floue.
4. L'algorithme des C-Moyennes Possibilistes (Possibilistic C-Means) qui est une variante modifiée du FCM .
5. L'algorithme des C-Moyennes Possibilistes Floues (Fuzzy Possibilistic C-Means) qui est une combinaison entre le FCM et PCM .

3.2 Généralités sur la classification

3.2.1 Classification

La classification consiste à organiser un ensemble de données multidimensionnelles en un ensemble fini de classes selon un ou plusieurs critère(s) de classification à l'aide d'un classifieur [15]. Il existe de nombreuses méthodes de classification de données multidimensionnelles.

La figure (3.1) illustre le principe de la classification. Les données sont représentées par des points (vecteurs) dans un espace à n dimensions (dans notre exemple $n = 2$). En sortie de la classification, nous obtenons m classes (ou familles) de points selon les critères donnés et le classifieur choisi (dans notre exemple, $m = 3$).

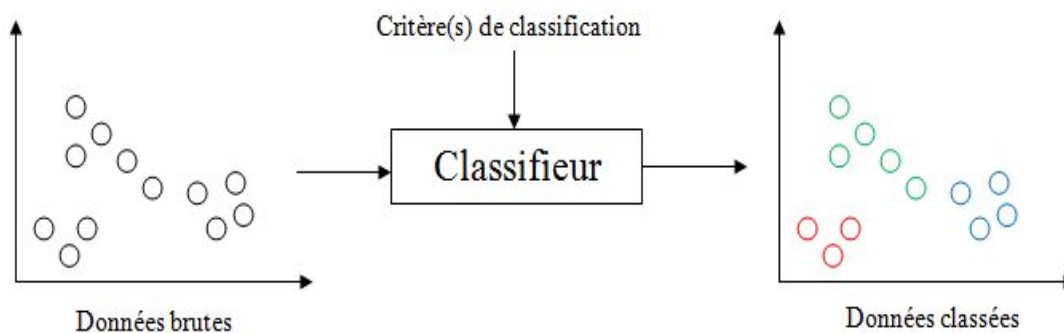


FIG. 3.1 – Principe de la classification.

La création d'une classe part du principe de ressemblance (voisinage) entre objets qui se regroupent et partagent des propriétés communes. Il convient donc de fixer des règles pour déterminer la ressemblance d'un pixel avec une classe (un ensemble de pixels), ou son appartenance à cette classe. Un pixel peut être associé à une classe seulement s'il ressemble aux objets contenus dans cette classe.

Partitionner l'espace spectral en classes équivaut à mesurer les distances entre classes. Ces distances sont issues de mesures statistiques (par exemple la distance Euclidienne ou la distance de Mahalanobis) et sont définies par le type de classe (circulaire, elliptique, conique) auquel le pixel doit être agrégé.

3.2.2 Méthodologie de la classification

La mise en œuvre d'une procédure de classification, ayant pour objectif de classer automatiquement des objets, comporte généralement deux phases fondamentales [16] :

- Une phase d'apprentissage dont le but est de déterminer un espace de représentation des signaux et de rechercher les paramètres discriminants capables de caractériser chaque classe d'objets.
- Une phase de reconnaissance au cours de laquelle on attribue à une classe chacun des différents objets inconnus dans l'espace de représentation déterminé durant l'apprentissage.

3.2.2.1 Phase d'apprentissage

Elle permet de définir des règles de décision à partir d'un ensemble d'objets de référence (base d'apprentissage), dont nous connaissons l'identité à priori (étiquette) et qui sont représentatifs de chaque classe.

3.2.2.2 Phase de classement (ou de reconnaissance)

Elle consiste à attribuer les objets à reconnaître à une des classes définies lors de l'apprentissage. En déterminant l'ensemble des attributs sélectionnés pour caractériser chaque objet à analyser, nous obtenons des observations qui peuvent être assignés aux différentes classes selon leurs positions dans l'espace des attributs par rapport à des critères de décisions.

3.2.3 Différentes classifications

Parmi toutes les techniques de classification, nous distinguons :

3.2.3.1 Classification supervisée

Dans laquelle un expert a fourni le modèle exact des classes à obtenir.

3.2.3.2 Classification non supervisée (clustering)

Regroupant des éléments ayant les mêmes propriétés statistique, géométriques ou linguistiques. Elle utilise un critère de regroupement qui peut être basé sur des distances entre objets (K-Means) ou sur des appartenances floues (FCM, PCM et FPCM). Dans ce type de classification le nombre de classes, inconnu a priori, est déduit directement des données.

nous distinguons ensuite :

3.2.3.3 Classification avec apprentissage

Pour laquelle le classifieur est entraîné à l'aide d'un ensemble de données connues a priori. Cet entraînement a pour but d'adapter les sorties du classifieur en fonction des entrées qui lui sont soumises.

3.2.3.4 Classification sans apprentissage

Le classifieur travaille directement sur les données sans aucune connaissance préalable.

Nous trouvons, enfin, des méthodes de classification :

3.2.3.5 Paramétriques

Pour lesquelles nous supposons que les données à classer suivent un modèle (probabiliste), une certaine fonction, fixe mais inconnue. Nous nous servons de cette fonction pour construire le classifieur.

3.2.3.6 Non paramétriques

Pour lesquelles le classifieur doit se débrouiller seul pour classer les données sans aide extérieure.

Dans le cas non supervisé, les classes sont construites en fonction des données, selon l'algorithme de classification utilisé. Dans notre étude, nous considérons aucune connaissance a priori sur les classes afin de déterminer l'appartenance d'un pixel à une classe, nous n'utiliserons que les méthodes de classification non supervisées.

3.2.4 Dissemblance et distance

Comme mentionné au début de ce chapitre, la classification non supervisée vise à former des groupes d'individus similaires. Cette notion de similarité est un élément essentiel de la classification non supervisée. Il est commun de définir le concept de similarité à l'aide de la notion duale de dissimilarité ; deux individus sont d'autant plus similaires qu'ils sont proches au sens d'une mesure de dissimilarité.

Nous rappelons ci-dessous la définition générale d'une mesure de dissimilarité avant de considérer le cas des métriques et des ultra-métriques qui sont deux types de mesures particulières.

Un indice ou mesure de dissemblance est une fonction d définie par [16] :

$$d : E \times E \rightarrow \mathbb{R}^+ \quad (3.1)$$

$$(x, y) \rightarrow d(x, y)$$

Tel que :

$$d(x, y) = d(y, x) \quad \forall x, y \in E. \text{ (symétrie)}$$

$$d(x, x) = 0 \quad \forall x \in E. \text{ (séparation)}$$

La distance Euclidienne est un indice de dissemblance particulier qui satisfait la propriété d'inégalité triangulaire :

$$d(x, y) \leq d(x, z) + d(z, y) \quad \forall x, y, z \in E. \text{ (inégalité triangulaire)}$$

L'ultra-métrie remplace la condition précédente par la condition suivante :

$$d(x, y) \leq \sup(d(x, z); d(z, y)) \quad (3.2)$$

Une distance entre un individu et un groupe et par la suite une distance entre deux groupes est naturellement inversement liée à l'appartenance de cet individu au groupe. Nous devons donc définir une stratégie de regroupements des éléments, c'est-à-dire se fixer des règles de calcul des distances entre groupes disjoints d'individus, appelées « critères d'agrégation ». Cette distance entre les classes pourra en général se calculer directement à partir des distances des différents éléments impliqués dans le regroupement.

Si x, y, z sont trois objets, et si les objets x et y sont regroupés en un seul élément noté h , nous pouvons définir la distance de ce groupement à z par la plus petite distance des divers éléments de h à z soit :

$$\text{dist}(h, z) = \min(\text{dist}(x, z), \text{dist}(y, z)) \quad (3.3)$$

Distance de Mahalanobis entre un point et un groupe de point dont le centre est C et la matrice de variance-covariance P .

$$d(x, C) = \sqrt{(x - C)^T P^{-1} (x - C)} \quad (3.4)$$

Dans le cas où P^{-1} est diagonale $P = \begin{pmatrix} W_1 & 0 \\ 0 & W_n \end{pmatrix}$:

$$d(x, C) = \sqrt{\sum_i W_i (x_i - C_i)^2} \quad (3.5)$$

C'est la distance de Minkowski pour $P = 2$. Dans l'espace R^n la distance entre le point et le groupe de points s'exprime de façon générale en fonction de leur attributs X_i et X_j est :

$$d_p(X_i, X_j) = \left(\sum_k W_k |x_{ik} - x_{jk}|^p \right)^{\frac{1}{p}} \quad (3.6)$$

Si $p = 1$ c'est la distance de Manhattan et pour $p = 2$, la distance de Mahalanobis.

Si P est la matrice identité on obtient la distance Euclidienne :

$$d(x, C) = \sqrt{\sum (x_i - C_i)^2} \quad (3.7)$$

Ainsi les distances permettent d'évaluer la similarité entre les objets. D'ailleurs, les termes similarité et dissimilarité sont équivalents, respectivement, à ressemblance et dissemblance .

3.3 Segmentation nette d'images par classification de pixels

3.3.1 Segmentation par la méthode des K-Means

3.3.1.1 Introduction

La méthode des K-Means est un outil de classification classique qui permet de répartir un ensemble de données en K classes homogènes. La plupart des images (photos, dessins vectoriels 2D, synthèses 3D, ...) vérifient localement des propriétés d'homogénéité, notamment en termes d'intensité (niveaux de gris ou couleurs). L'algorithme des K-Means permet donc d'apporter une solution à la segmentation d'images.

La méthode des K-Means a été introduite par J. McQueen [17] en 1971 et mise en œuvre sous sa forme actuelle par E. Forgy [18]. De nombreuses variantes se sont succédées depuis afin d'étendre ses capacités de classification (séparations non linéaires) : kernel K-Means [19] (K-Means basée sur des méthodes à noyaux), d'améliorer ses performances : global- K-Means, K-Harmonic Means, et d'automatiser le choix du nombre de clusters : Gaussian-means, X-Means [20].

3.3.1.2 Principe de la méthode des K-Means [21]

L'algorithme des K-Means est un algorithme général de regroupement qui tend à rassembler N points ou primitives dans C groupes dont le nombre est prédéterminé. Étant donné un ensemble de points $X = x_1, x_2, \dots, x_N$. chaque point x_j est caractérisé par n variables et donc représenté dans un espace à n dimensions $x_j \in R^n$.

Dans le cadre de la classification non supervisée, nous cherchons généralement à partitionner l'espace en classes concentrées et isolées les unes des autres. Dans cette optique, l'algorithme des K-Means vise à minimiser la variance intra-classe, qui se traduit par la minimisation de la fonction objective (Energie) suivante :

$$J(X, V) = \frac{1}{2} \sum_{i=1}^C \sum_{j=1}^N \|x_j - v_i\|^2 \quad (3.8)$$

où :

- v_i est le centre du cluster (classe) i .
- $\| \ \|$ représente la norme, ici Euclidienne qui mesure la dissimilarité entre un point et un centre.

L'algorithme se résume en les étapes suivantes (voir **Algorithme 1**) :

Algorithm 1 Algorithme des K-Means

Étape 1 : Choisir aléatoirement les C centres initiaux v_1, v_2, \dots, v_C à partir des points $\{x_1, x_2, \dots, x_N\}$.

Répéter

Étape 2 : Assigner x_j , $j = 1, 2, \dots, N$ au cluster Z_i , $i = 1, 2, \dots, C$

Si

$$\|x_j - v_i\| \leq \|x_j - v_p\|, p = 1, 2, \dots, C \text{ avec } j \neq p$$

Choix aléatoire si conflit

Étape 3 : Calculer les nouveaux centres : $v_i^* = \frac{1}{n_i} \sum_{x_j \in Z_j} x_j$, $i = 1, 2, \dots, C$
 n_i est le nombre d'éléments dans le cluster Z_i

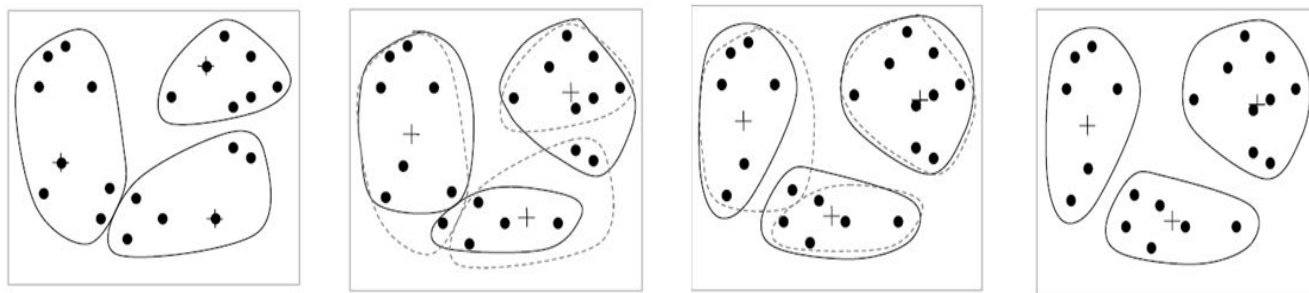
Jusqu'à ce que $v_i^* = v_i, \forall i = 1, 2, \dots, C$.

3.3.1.3 Convergence et initialisation des K-Means

L'algorithme ainsi défini converge vers un minimum local de l'énergie, qui se traduit par une partition de l'espace des données en des classes séparées par des hyper plans. La qualité de la solution ainsi trouvée dépend fortement des noyaux initiaux. De plus la sensibilité de l'algorithme à l'initialisation est d'autant plus grande que la dimensionnalité des données est grande.

3.3.1.4 Choix du nombre K de classes (clusters)

L'algorithme classique des K-Means laisse un paramètre libre ; le nombre de classes, ce qui dans le cas de la segmentation d'images correspond au nombre d'intensités utilisées pour représenter l'image. Généralement le choix de K est fait empiriquement en sélectionnant la valeur de K qui minimise l'équation 3.8.



(a) Choix aléatoire de K objets Centres initiaux et calcul des clusters

(b) Calcul des centres des clusters et mise-à-jour des clusters.

(c) Mise-à-jour des centres des clusters et mise-à-jour des clusters.

(d) Arrêt lorsque les clusters sont stables (critère stable).

FIG. 3.2 – Exemple illustratif du déroulement de l'algorithme des K-Means.

3.3.1.5 Résultats et discussion

Si nous voulons évaluer de manière objective la méthode décrite précédemment, il est préférable d'utiliser des images de synthèse où contours et régions sont parfaitement localisés ainsi que la qualité de la segmentation de référence. Cependant, nous reprocherons à ces images de ne pas représenter toutes les situations possibles d'une utilisation réelle.

Nous avons effectué des tests sur les deux images synthétiques (Image synthétique 1 et 2), choisies de telle sorte qu'elles permettent de bien mettre en exergue les caractéristiques de la méthode :

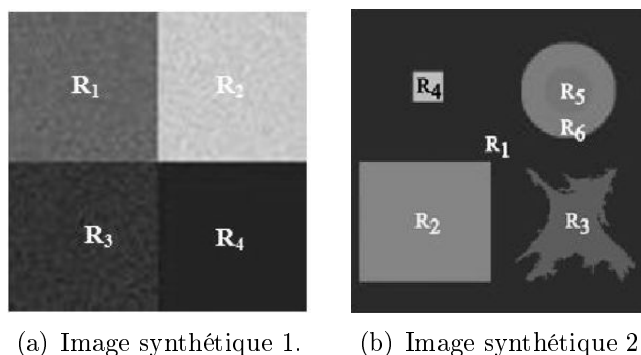


FIG. 3.3 – Images synthétiques utilisées pour les tests (K-Means).

- **Image synthétique 1** : composée de 4 régions de forme carrée bien distinctes.
- **Image synthétique 2** : composée de 6 régions de forme diverses (rectangle , carrée, disque , anneau et une forme irrégulière) bien distinctes.

3.3.1.5.1 Initialisation des centres des classes Afin d'illustrer l'influence de l'initialisation des centres des classes, nous appliquons l'algorithme des K-Means aux deux images synthétiques de la figure (3.3) :

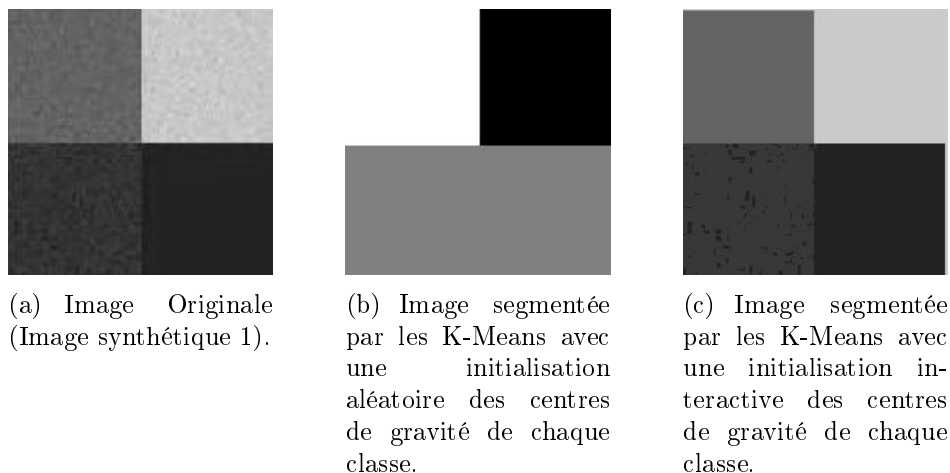


FIG. 3.4 – Influence de l'initialisation des centres des classes (Image synthétique 1) .

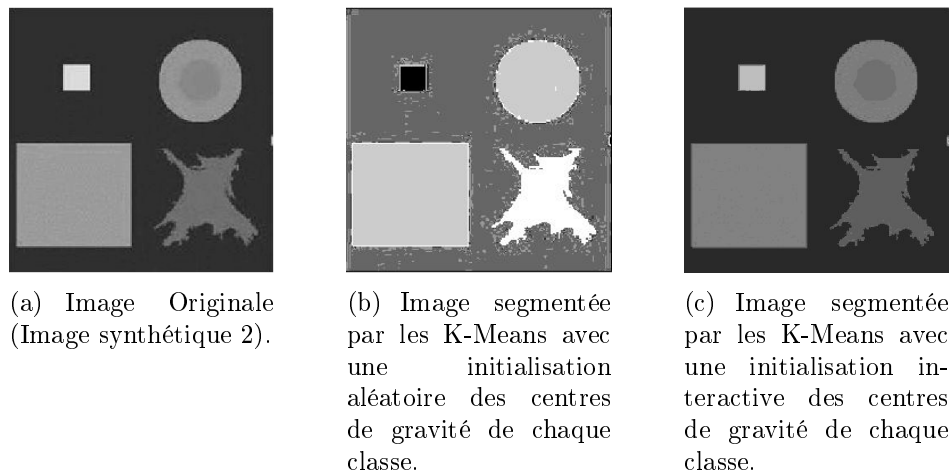


FIG. 3.5 – Influence de l’initialisation des centres des classes (Image synthétique 2).

L’image de la figure (3.4.b) et (3.5.b) illustre les résultats de la segmentation lorsque les centres de gravité des classes sont initialisés de façon aléatoire. L’image segmentée de la figure (3.4.c) et (3.5.c) est obtenue en initialisant interactivement les centres de gravité des classes à proximité des moyennes de niveaux de gris des pixels de chaque région.

Dans l’image de la figure (3.4.b), nous constatons que 3 classes de pixels ont été construites. Par ailleurs, nous remarquons que certaines classes sont constituées de pixels provenant de différentes régions. En effet, les pixels des régions R_4 et R_3 sont affectés à la même classe. Par contre pour l’image de la figure (3.5.b), nous constatons que 5 classes de pixels ont été construites. Aucun pixel n’a été affecté à la sixième classe car la position initiale de son centre de gravité est trop isolée dans l’espace des niveaux de gris. Par ailleurs, nous remarquons que certaines classes sont constituées de pixels provenant de différentes régions. En effet, les pixels des régions R_5 et R_2 sont affectés à la même classe. Il en est de même pour les pixels des régions R_5 et R_6 dont les niveaux de gris sont très proches. Par ailleurs, les pixels appartenant à la région du fond R_1 sont affectés à une classes car le centre de gravité initial a été positionné à proximité du centre de gravité de la classe correspondant au fond de l’image.

Dans l’image de la figure (3.4.c), les centres de gravité des classes sont positionnés interactivement de telle sorte qu’ils soient suffisamment proches des centres réels des classes. Cette fois-ci, les 4 classes de pixels sont bien construites. De même pour l’image de la figure (3.5.c) où nous constatons que les 6 classes de pixels sont bien construites mais quelques erreurs de classification subsistent pour des pixels appartenant aux régions R_5 et R_6 à cause du recouvrement de leurs niveau de gris. Ceci montre que le résultat de la classification des pixels dépend très fortement de l’initialisation des centres.

3.3.1.5.2 Choix du nombre K de classes Pour voir l'influence du nombre K de classes, nous appliquons l'algorithme des K-Means toujours aux deux images synthétiques de la figure (3.3) :

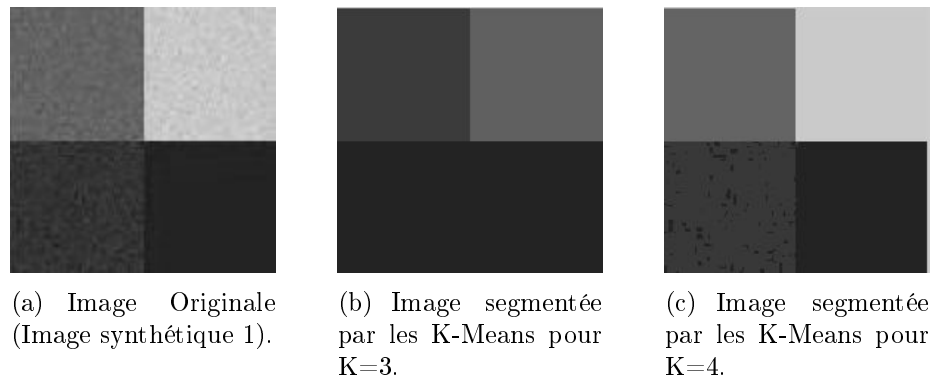


FIG. 3.6 – Résultats de la classification pour $K = 3$ et $K = 4$ (Image synthétique 1).

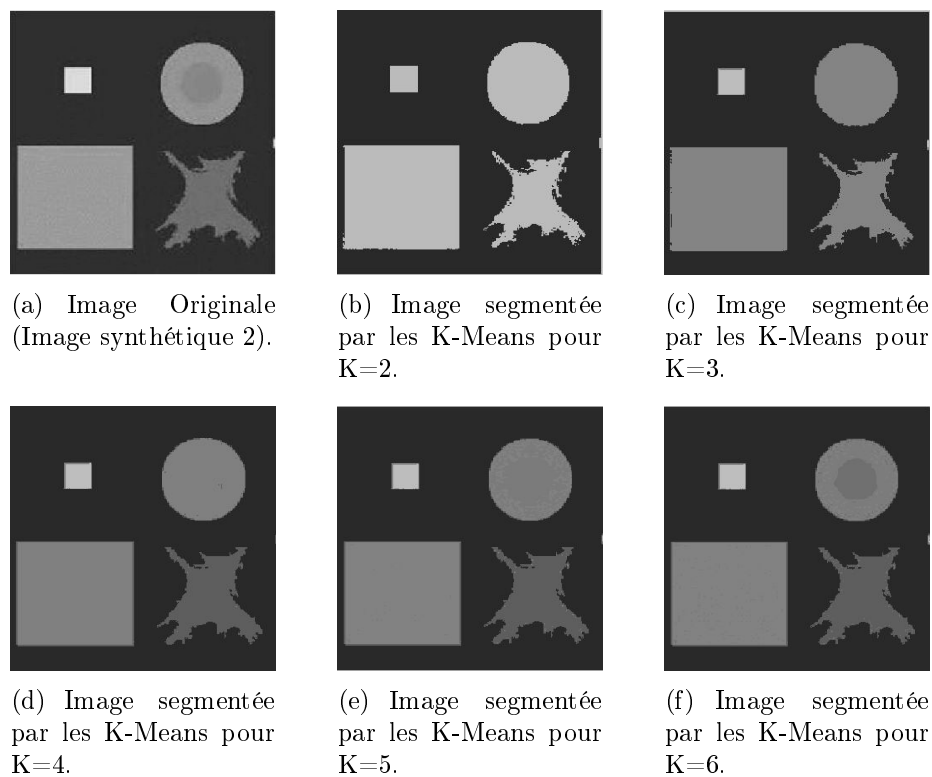


FIG. 3.7 – Résultats de la classification pour $K = 2 \dots 6$ (Image synthétique 2).

Les figures (3.6) et (3.7) illustrent les résultats de la segmentation pour différente valeur de K , nous remarquons par exemple que pour $K = 4$ le nombre de classes de pixels construites est de 4 mais plus ce nombre est grand plus nous nous approchons de l'image originale. Les résultats dépendent de ce paramètre essentiel qui est le nombre de classes, il permet de définir le niveau de finesse de l'analyse.

3.3.1.5.3 Immunité au bruit Pour tester l'immunité du classifieur des K-Means vis-à-vis du bruit, nous avons ajouté aux deux images synthétiques de la figure (3.3) un bruit gaussien de moyenne nulle et de variance relative à la valeur moyenne de l'image :

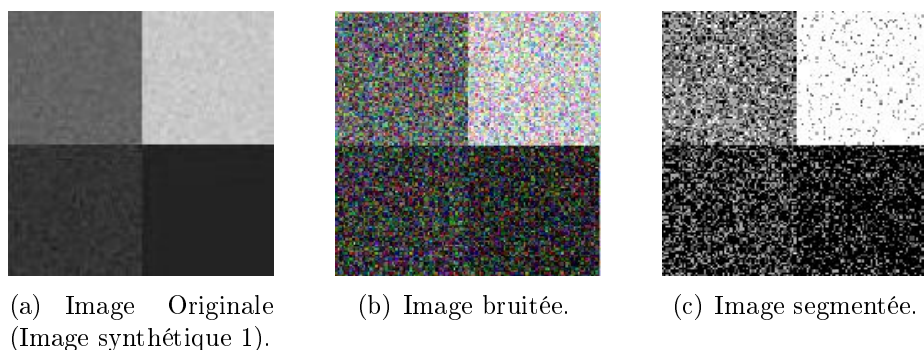


FIG. 3.8 – Résultat de la classification pour un bruit Gaussien ($m = 0, \sigma = 0,05$) (Image synthétique 1).

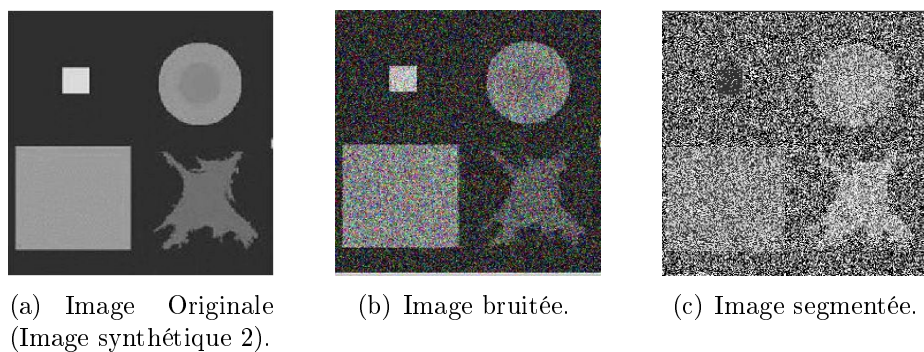


FIG. 3.9 – Résultat de la classification pour un bruit Gaussien ($m = 0, \sigma = 0,05$) (Image synthétique 2).

Les résultats observés sur les deux figures (3.8.c) et (3.9.c) montrent que le classifieur des K-Means est très sensible au bruit car les deux images segmentées sont noyées dans le bruit surtout pour l'image de la figure(3.9.c).

3.3.2 Segmentation par la méthode “Mean Shift”

Le Mean Shift permet dans un premier temps de lisser l'image et d'obtenir ainsi une première partition. La variance et la moyenne sont les critères de regroupement des points de l'image. La méthode ne demande pas cependant de connaissance sur l'image. La densité de probabilité de l'image permet d'effectuer ce groupement. Plusieurs méthodes sont utilisées : les statistiques paramétriques et les statistiques non paramétriques [16].

Dans le cas paramétrique, une connaissance a priori, en faisant une hypothèse sur la loi suivie par la fonction de densité du nuage de points, est utilisée. Dans les cas réels, la fonction considérée a souvent plusieurs modes, ce qui n'est pas le cas pour les lois usuelles. Une somme de lois normales peut permettre de traiter des données multimodales, mais une connaissance sur le nombre de modes est nécessaire.

Pour ne faire aucune hypothèse sur les données, nous devons donc utiliser les statistiques non paramétriques. Le Mean Shift travaille dans une fenêtre dynamique qui évolue dans la direction du gradient de la densité, cela le rend adapté pour les structures locales en préservant les détails de l'image.

3.3.2.1 Estimation de la densité de probabilité

L'estimation non paramétrique de la densité de probabilité, connue sous le nom de technique du noyau de Parzen [22] est la méthode la plus connue.

Soit n points des données $x_i, i = 1, 2, \dots, N$ dans \mathbb{R}^d , l'estimateur de densité multivariable avec le noyau $K(x)$ est une matrice H de taille $d \times d$ définie, symétrique et positive, calculée au point x est donné par :

$$\hat{f}(x) = \frac{1}{N} \sum_{i=1}^N K_H(x - x_i) \quad (3.9)$$

Où :

$$K_H(x) = |H|^{-\frac{1}{2}} K\left(H^{-\frac{1}{2}} x\right) \quad (3.10)$$

Le noyau à d variables $K(x)$ est une fonction bornée sur un support compact \mathbb{R}^d tel que :

$$\int_{\mathbb{R}^d} K(x) dx = 1 \quad (3.11)$$

$$\lim_{\|x\| \rightarrow \infty} \left(\|x\|^d K(x) \right) = 0 \quad (3.12)$$

$$\int_{\mathbb{R}^d} x K(x) dx = 0 \quad (3.13)$$

$$\int_{\mathbb{R}^d} x x^T K(x) dx = C_K I \quad (3.14)$$

Où C_K est une constante. Dans le cas multivariable, le noyau peut être généré à partir d'un noyau univariable $K_1(x)$ de deux manières différentes.

$$K^P(x) = \prod_{i=1}^d K_1(x_i) \quad (3.15)$$

$K^P(x)$ est obtenu à partir du produit des noyaux univariable. Une classe spéciale de noyaux symétriques [23] radiaux peut être utilisée, elle satisfait :

$$K(x) = C_{K,d} k(\|x\|^2) \quad (3.16)$$

Dans ce cas, il suffit de définir la fonction $K(x)$ appelée le « profil » du noyau, seulement pour $x > 0$. La constante de normalisation $C_{K,d}$ qui assure que $\int_{\mathbb{R}^d} K(x) dx = 1$ est supposée supérieur à 0.

La matrice H est choisie proportionnelle à la matrice identité $H = h^2 I$. L'avantage est que seulement un paramètre $h > 0$ doit être fourni ; comme nous pouvons le voir à partir de l'équation (3.10), en employant seulement un paramètre pour H , l'estimation de la densité, équation (3.9) devient l'équation bien connue :

$$\hat{f}(x) = \frac{1}{Nh^d} \sum_{i=1}^N K\left(\frac{x - x_i}{h}\right) \quad (3.17)$$

Le choix de h (appelé paramètre de lissage ou largeur de bande) est crucial. En effet, plus h est élevé, plus la densité de probabilité estimée est lissée et certains modes locaux risquent d'être omis. Inversement, si h est trop petit, la densité de probabilité estimée laisse apparaître des modes inexistants ou des modes associés au bruit dans le cas de données bruitées.

L'équation (3.17) est l'estimateur de densité pour l'ensemble de données $S_I = \{x_i \mid i = 1, 2, \dots, N\}$ dans un espace de dimension d . Cette expression peut être vue comme étant une somme des fonctions de noyau centré sur chacune des observations x_i .

Rappelons que la fonction noyau $K(x)$ satisfait l'équation (3.11) et que sa valeur décroît quand $|x|$ augmente. De là, $g(x) = K(\frac{x-x_i}{h})$ peut être vue comme une « hyperbosse » centrée en x_i dans l'espace à d dimensions.

Voici quelques exemples de fonctions noyaux utilisées couramment :

- Noyau Gaussien (figure (3.10.a)) : $(2\pi)^{-\frac{d}{2}} \exp(-\frac{1}{2} \|x\|^2)$.
- Noyau d'Epanechnikov (figure (3.10.b)) : $K_E(x)$.
- Noyau exponentiel (figure (3.10.c)) : $2^{-d} \exp(-\|x\|)$.

Avec :

$$K_E(x) = \begin{cases} \frac{1}{2} C_d^{-1} (d+2) (1 - \|x\|^2) & \text{pour } \|x\| < 1 \\ 0 & \text{sinon} \end{cases} \quad (3.18)$$

Où C_d est le volume de l'hypersphère unité de dimension d :

$$C_d = \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})}$$

Avec $\Gamma(\cdot)$ est la fonction gamma tel que : $\Gamma(\frac{1}{2}) = \frac{\sqrt{\pi}}{2}$.

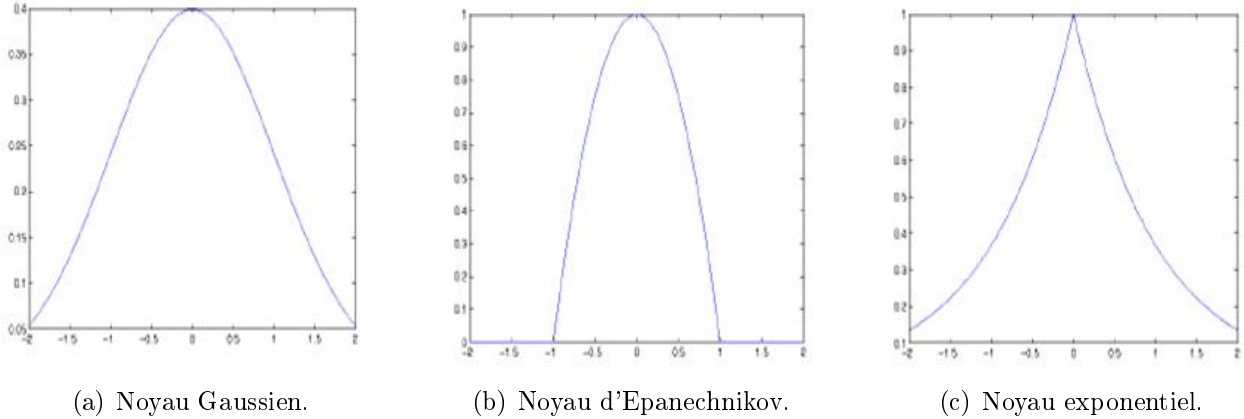


FIG. 3.10 – Exemples de fonctions noyaux utilisées.

Le choix de l'expression du noyau est déterminé par la mesure de la qualité de l'estimation de la reconstruction de \hat{f} de f . Cette qualité peut se mesurer en utilisant l'erreur quadratique moyenne intégrée (*EQMI*)

$$EQMI = E \left[\int_{\mathbb{R}^d} (\hat{f}(x) - f(x))^2 dx \right] \quad (3.19)$$

$$EQMI = \int_{\mathbb{R}^d} E (\hat{f}(x) - f(x))^2 dx = \int_{\mathbb{R}^d} EQM \{ \hat{f}(x) \} dx \quad (3.20)$$

Avec *EQM* l'erreur quadratique moyenne :

$$EQM \{ \hat{f}(x) \} = E (\hat{f}(x) - f(x))^2 \quad (3.21)$$

Le critère *EQMI* (équation 3.20) est minimal dans le cas de l'utilisation du noyau d'Epanechnikov [22]. Ainsi, ce noyau permet une estimation de la densité d'un échantillon x_i en prenant en compte son voisinage compris dans une hypersphère de rayon h .

3.3.2.2 Algorithme de classification du Mean Shift [24]

La détermination des modes d'une densité de probabilité à l'aide des Mean-Shift s'effectue sur une distribution de points X . Le principe de la méthode est de chercher à annuler le gradient de la densité de probabilité :

$$\nabla f(x) = 0 \quad (3.22)$$

La densité de probabilité $f(x)$ étant inconnue, il est impossible de calculer analytiquement le gradient de celle-ci. Le calcul de l'estimation du gradient de la densité de probabilité est remplacé par le calcul du gradient de la densité de probabilité estimée :

$$\hat{\nabla} f(x) \equiv \nabla \hat{f}(x) \quad (3.23)$$

En dérivant l'équation (3.17) nous obtenons :

$$\nabla \hat{f}(x) = \frac{1}{Nh^d} \sum_{i=1}^N \nabla K \left(\frac{x - x_i}{h} \right) \quad (3.24)$$

En utilisant le noyau d'Epanechnikov (équation 3.18) :

$$\nabla \hat{f}(x) = \frac{1}{Nh^d C_d} \frac{d+2}{h^2} \left[\frac{1}{n_x} \sum_{x_i \in S_h(x)} (x - x_i) \right] \quad (3.25)$$

Où $S_h(x)$ est l'hypersphère de rayon h . de volume $h^d C_d$ centrée en x et contenant n_x points de la distribution.

Le dernier terme de l'équation (3.25) est appelé vecteur Mean-Shift :

$$M_h(x) = \frac{1}{n_x} \sum_{x_i \in S_h(x)} (x - x_i) = \left(\frac{1}{n_x} \sum_{x_i \in S_h(x)} x_i \right) - x \quad (3.26)$$

Il est une estimation du gradient de la densité de probabilité estimée : $M_h(x) \equiv \nabla \hat{f}(x)$, pointant dans la direction de la plus grande pente de la densité de probabilité. Le vecteur Mean-Shift correspond à la moyenne des écarts entre le point x et ses voisins dans l'hypersphère.

L'algorithme des Mean-Shift est une procédure itérative. Nous pouvons à partir de chaque point x_i de la distribution, déterminer son mode de convergence. Ainsi, à chaque point de la distribution est associée un mode, celui-ci pouvant être identique ou différent des modes précédemment déterminés. La procédure itérative des Mean Shift (voir **Algorithme 2**) est résumée dans l'algorithme ci-après :

Algorithme 2 Algorithme des Mean-Shift, la procédure est répétée pour chaque $x \in X$

Entrée : x : Vecteur de dimension d appartenant à la distribution X .

h : Rayon de l'hypersphère.

$m \leftarrow x$

Tant que m est différent à chaque itération **Faire**

- Calculer $M_h(m)$ à l'aide de l'équation (3.26)
- $m \leftarrow x + M_h(m)$

Fin tant que

m est le mode associé à x

Chaque mode ainsi déterminé représente une classe conduisant à une carte de segmentation.

3.3.2.3 Résultats et discussion

3.3.2.3.1 Influence du paramètre de lissage (h) Pour illustrer cette méthode de classification, nous avons effectué des tests sur les deux images synthétiques (Image synthétique 1 et 2) de la figure (3.3) pour deux valeurs du paramètre (h) :

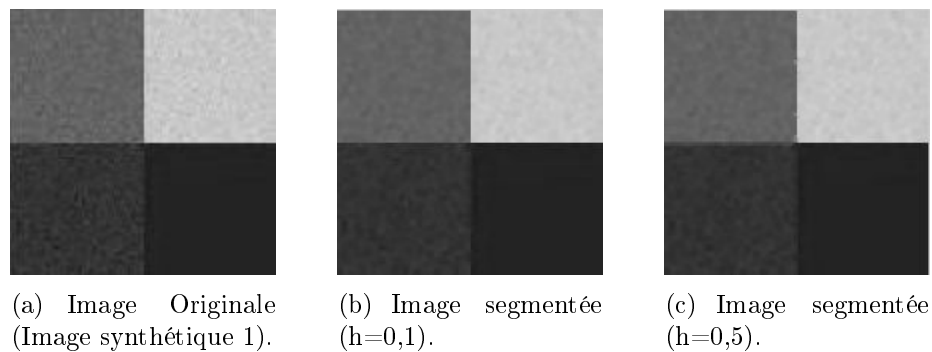


FIG. 3.11 – Résultat de la classification des pixels de l'image synthétique 1 pour $h = 0, 1 \dots 0, 5$

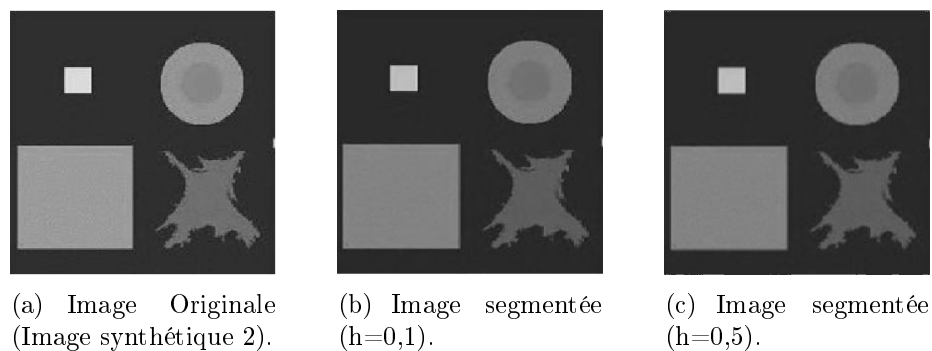


FIG. 3.12 – Résultat de la classification des pixels de l'image synthétique 2 pour $h = 0, 1 \dots 0, 5$

Les figures (3.11) et (3.12) illustrent les résultats de la segmentation pour différentes valeurs de h qui est un paramètre de lissage, plus ce paramètre est élevé plus l'image segmentée est lissée. Un bon choix de ce paramètre est nécessaire pour une meilleure segmentation.

3.3.2.3.2 Immunité au bruit Afin de voir la sensibilité du classifieur « Mean Shift » vis-à-vis du bruit, nous avons ajouté aux deux images synthétiques de la figure (3.3) un bruit gaussien de moyenne nulle et de variance relative à la valeur moyenne de l'image :

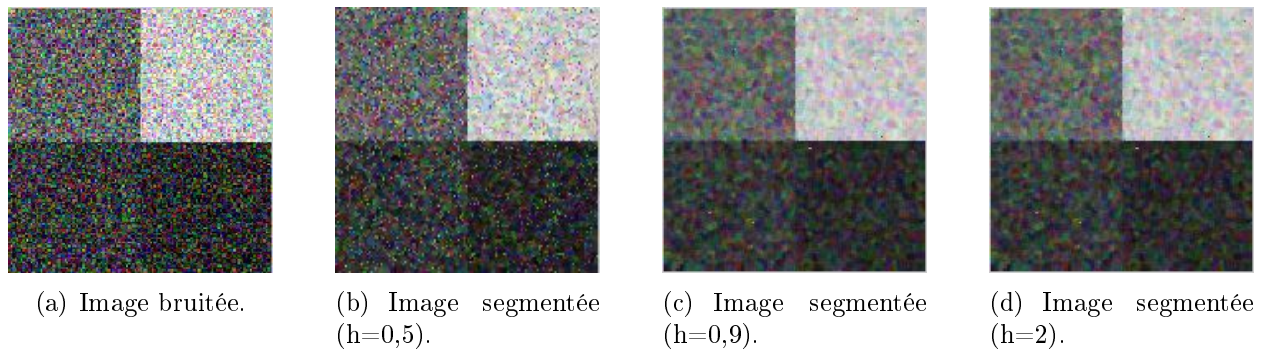


FIG. 3.13 – Résultat de la classification des pixels de l'image synthétique 1 bruitée avec un bruit Gaussien ($m = 0, \sigma = 0,05$) et pour différentes valeurs de h .

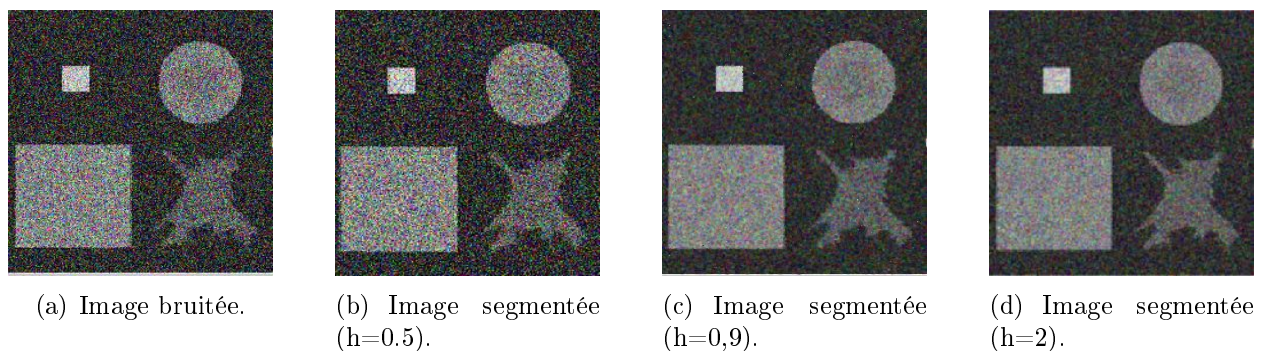


FIG. 3.14 – Résultat de la classification des pixels de l'image synthétique 2 bruitée avec un bruit Gaussien ($m = 0, \sigma = 0,05$) et pour différentes valeurs de h .

La segmentation résultante du classifieur « Mean Shift » est presque correcte pour une certaine valeur du paramètre de lissage (h), l'algorithme offre ce paramètre qui permet de réduire le bruit (figure (3.13.c) et (3.14.c)). Par contre au-delà d'un certain seuil nous remarquons que l'image segmentée est trop lissée (figure (3.13.d) et (3.14.d)). Donc un réglage approprié du paramètre (h) permet d'obtenir de meilleurs résultats.

3.4 Méthodes floues de classification

Les méthodes de segmentation nette considèrent que les régions constituent des sous-ensembles nets de l'ensemble des pixels constituant l'image. La reconstruction des régions s'effectuent en affectant chaque pixel à une et seule région. Cette démarche présente des limites lorsque nous ne disposons pas de connaissance a priori sur l'affectation des pixels.

L'image de la figure (3.15) représente un détail agrandi de l'image originale (Image synthétique 2). Nous pouvons constater qu'il est délicat de décider si le pixel P_1 appartient à la région R_5 ou à la région R_6 . Pour répondre à ce type de situation, les méthodes de segmentation floue supposent que les pixels appartiennent plus ou moins à ces sous-ensembles et considèrent que les pixels peuvent ainsi regrouper en sous-ensembles flous de pixels. Pour cela, ces méthodes sont basées sur les concepts de la logique floue.

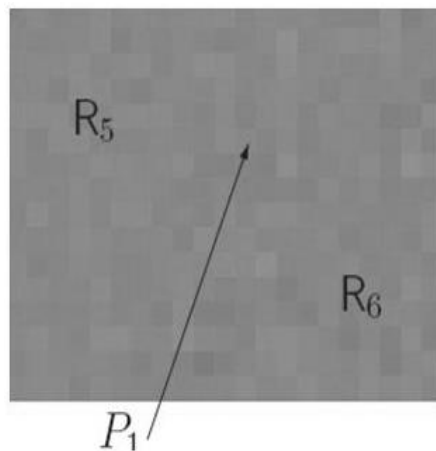


FIG. 3.15 – Détail de l'image originale.

3.4.1 Principaux concepts de la logique floue

Voilà plus de 40 ans que le souci d'automatiser les actions humaines, la mise en équation de concepts flous à été développé mathématiquement grâce aux travaux de Lotfi Zedeh qui a introduit la notion de sous-ensemble flou en 1965 [26]. Dans ce paragraphe, nous allons présenter la définition d'un sous-ensemble flou ainsi que ses principales caractéristiques[27].

3.4.1.1 Définition d'un sous-ensemble flou

Dans un sous-ensemble classique A dont les éléments sont X , sa fonction caractéristique est χ_A , prend la valeur 0 pour les éléments x n'appartenant pas à A et la valeur 1 pour les éléments x appartenant à A , il peut être écrit :

$$\chi_A : X \rightarrow \{0, 1\} \quad (3.27)$$

Pour un sous-ensemble flou A de X , il est défini par une fonction d'appartenance qui associé a chaque élément x de X le degré d'appartenance $\mu_A(x)$ compris entre 0 et 1 tel que :

$$\mu_A : X \rightarrow [0 \ 1] \quad (3.28)$$

3.4.1.2 Caractéristiques d'un sous-ensemble flou

Les principales caractéristiques d'un sous-ensemble flou A de X sont :

3.4.1.2.1 Support de A Le support de sous-ensemble flou A noté $supp(A)$ est la partie X sur laquelle la fonction d'appartenance $\mu_A(x)$ n'est pas nulle :

$$supp(A) = \{x \in X / \mu_A(x) \neq 0\} \quad (3.29)$$

3.4.1.2.2 Hauteur de A La hauteur de sous-ensemble flou A noté $h(A)$ est la plus grande valeur prise par la fonction d'appartenance $\mu_A(x)$:

$$h(A) = sup_{x \in X} \mu_A(x) \quad (3.30)$$

3.4.1.2.3 Noyau de A Le noyau de sous-ensemble flou A noté $noy(A)$ est l'ensemble des éléments X pour laquelle la fonction d'appartenance $\mu_A(x)$ vaut 1 :

$$noy(A) = \{x \in X / \mu_A(x) = 1\} \quad (3.31)$$

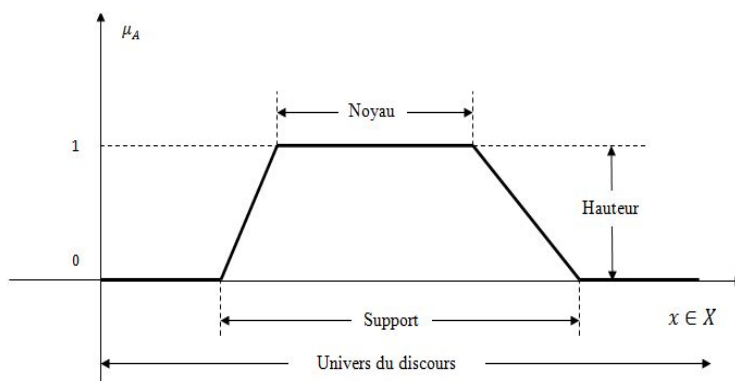


FIG. 3.16 – Support, Noyau, Hauteur d'un sous-ensemble flou.

3.4.1.3 Exemple d'un sous-ensemble flou

L'ensemble de la figure (3.17) contient les nombres réels "proches" de 3. Ainsi, 3 appartient complètement à cet ensemble et il a, par conséquent, un degré 1. Plus nous nous éloignons de 3, plus le degré d'appartenance diminue. Cette propriété est exploitée en segmentation d'images,

et plus précisément en classification où les classes sont représentées par des ensembles flous. Cela est fort utile lorsque les régions ne peuvent pas être définies de manière nette et précise. Leur manipulation en gardant le caractère flou permet de traiter des données imprécises, incertaines et/ou redondantes.

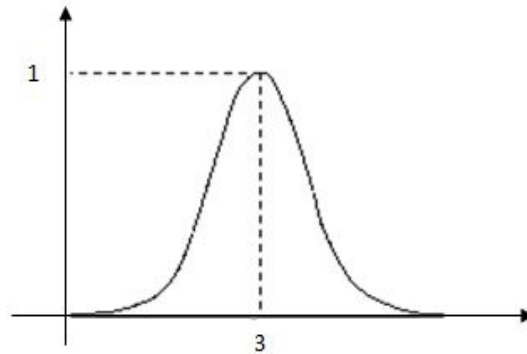


FIG. 3.17 – Exemple d'un sous-ensemble flou.

3.4.2 Segmentation floue d'images par classification de pixels

Le point clé de la segmentation d'images par classification de pixels est la construction des classes de pixels en présence. Celle-ci s'appuie sur les concepts de la logique floue décrits précédemment en considérant les pixels comme des sous-ensembles flous. Les méthodes de construction des classes telles que les C-Means et les Fuzzy C-Means seront présentées dans ce qui suit.

Les algorithmes C-Means (CM) ont été introduits par Duda et Hart en 1973 et ils ont été améliorés par Bezdek en 1981. Ce sont des algorithmes basés sur la minimisation des probabilités qui permettent l'approximation efficace des centres des classes. L'algorithme initial des CM est une extension des Hard C-Means[28] Ainsi, le FCM appelé Fuzzy ISODATA [29] est la version flouifiée de l'algorithme HCM ou Hard ISODATA[30] introduit par Dunn et améliorée par Bezdek plus tard.

3.4.2.1 C-Means (CM)

Soit $E = \{x_1, x_2, \dots, x_k\}$ l'ensemble des échantillons de la base d'apprentissage. x_k est le k-ième vecteur d'attribut x . Les prototypes ou ensembles des centres de classes sont $\{V_1, V_2, \dots, V_c\}$ où c dénote le nombre de classes.

Les algorithmes de type C-Means minimisent tous une fonction objective J (ou somme des distances intra-classes). Cette distance Euclidienne d est calculée entre le vecteur attribut des données et les prototypes de groupes. La fonction objective J est de la forme :

$$J = \sum_{i=1}^c \sum_{k=1}^n \mu_{j,k} d^2(x_k, V_i) \quad (3.32)$$

Où :

- V_i : représente le centre de la classe i .
- $\mu_{j,k} = P(c_j/x_k)$: en général une probabilité d'appartenance de x_k à la classe c_j .

3.4.2.2 Fuzzy C-Means (FCM)

Soit E un ensemble de vecteurs d'attributs, $\{V_1, V_2, \dots, V_c\}$ les centres des différentes classes où c dénote le nombre de classes. Le degré d'appartenance d'un élément x_k à un groupe V_i est noté $\mu_i(x_k)$. Dans cette approche, un vecteur d'attribut donné peut appartenir à plusieurs groupes.

U est la matrice des degrés d'appartenance (appelée également matrice de C-partition floue) de dimension $c \times n$ où c est le nombre de classes et n le nombre d'éléments à classer. L'algorithme de classification non supervisé des FCM est une méthode de partitionnement d'ensemble basé sur l'itération de Picard à travers les conditions nécessaires pour optimiser une somme d'erreurs au carré d'une fonction objective (J_m) qui est construite à base de la somme des distances entre le vecteur et le centre des classes pondérée par les fonctions d'appartenances :

$$J_m(U, V) = \sum_{i=1}^c \sum_{k=1}^n \mu_{ik}^m d^2(x_k, V_i) \quad (3.33)$$

Où : $d = \|x_k - V_i\|$ est la norme Euclidienne et V_i représente le centre de la classe i .

Le paramètre $m > 1$ contrôle le degré de flou. Plus m est grand, plus la partition est floue. En effet, Dunn a tout d'abord étendu J_1 (pour $m = 1$) à J_2 (pour $m = 2$) [31]. Bezdek lui a généralisé J_2 à J_m ($1 < m < \infty$) [32].

$\forall i, j, \mu(x_k) \in \{0, 1\}$ $U = [\mu_{ik}(x_k)]$ c'est-à-dire :

$$U = \begin{pmatrix} \mu_{11} & \mu_{12} & \cdot & \cdot & \mu_{1n} \\ \mu_{21} & \cdot & \cdot & \cdot & \mu_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \mu_{c1} & \cdot & \cdot & \cdot & \mu_{cn} \end{pmatrix}$$

Trois contraintes sont définies pour la matrice des degrés d'appartenance U . La première impose que les valeurs des u_{ik} soient comprises entre 0 et 1 ce qui signifie que tout pixel appartient à la réunion des classes. La deuxième impose que l'appartenance d'un pixel soit répartie sur l'ensemble des classes, sans exclure l'existence de degrés d'appartenance nuls. Enfin, la troisième exprime que la classification ne produit pas de classe vide afin que tous les pixels soient caractérisés. Formellement, ces contraintes s'écrivent comme :

$$0 \leq \mu_{ik} \leq 1 \quad 1 \leq i \leq c, 1 \leq k \leq n \quad (3.34)$$

$$\sum_{i=1}^c \mu_{ik} = 1 \quad , \forall k \in [1, n] \quad (3.35)$$

$$0 \leq \sum_{i=1}^c \mu_{ik} \leq 1, \quad \forall i \in [1, c] \quad (3.36)$$

L'idée de base de la classification FCM est d'assigner à chaque vecteur x_k un degré d'appartenance μ_{ik} , à chaque classe centrée en V_i . L'algorithme minimise une certaine erreur entre les classes en calculant itérativement le degré d'appartenance et les centres de classes en utilisant les relations préalablement définies.

Le FCM est basé sur la mise à jour de la fonction d'appartenance pendant l'itération de l'algorithme donc le FCM fait ainsi évoluer la partition (matrice U) en minimisant la fonction objective J_m (**voir Algorithme 3**).

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}} \quad (3.37)$$

Où μ_{ik} est la fonction de mise à jour des degrés d'appartenance (avec $d_{ik} = \|x_i - c_k\|$).

La fonction de mise à jour des centres est :

$$V_i = \frac{\sum_{k=1}^n \mu_{ik}^m x_k}{\sum_{k=1}^n \mu_{ik}^m} \quad (3.38)$$

Le test d'arrêt au temps $t + 1$ est donné par la différence $|J_m^{t+1} - J_m^t| < \text{seuil}$.

Algorithm 3 Algorithme général d'exécution du FCM

Étape 1 :

- Choisir le nombre de classes : c
- Fixer m , $1 < m < \infty$
- Choisir ε pour le test d'arrêt.
- Initialiser ($t = 0$) la matrice de partition $U^{t=0}$ des fonctions d'appartenance à des valeurs aléatoires ou bien en leur affectant comme valeur $\frac{1}{c}$.
- Choisir la mesure d_{ik} on prend par exemple $d_{ik} = \|x_i - c_k\|$.

Étape 2 : Calculer le vecteur C-Mean V_i des centres en utilisant l'équation (3.37).

Étape 3 : Mettre à jour U^t en utilisant :

$$\mu_{ik}^{t+1} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}} \quad (3.39)$$

Étape 4 : Comparer J_m^{t+1} à J_m^t si $|J_m^{t+1} - J_m^t| < \varepsilon$ alors arrêt sinon $t = t + 1$ et retourner à l'étape 2.

3.4.2.2.1 Analyse de l’algorithme FCM [33] L’algorithme est limité par le choix à priori des paramètres suivants :

- 1) Le nombre de classes qui est fourni à l’avance.
- 2) L’initialisation aléatoire des paramètres.
- 3) Le choix du prototype.
- 4) Le paramètre de contrôle du flou (m).

Ainsi l’algorithme est assez coûteux en place mémoire et les fonctions d’appartenance sont souvent sensibles au bruit.

3.4.2.2.2 Paramètre de contrôle du flou (m) Les valeurs utiles pour ce paramètre sont comprises entre 1,5 et 3 et il n’existe pas de fondement mathématique quand à son existence et à la valeur qui lui est affectée. Le plus souvent $m = 2$ se révèle adéquat. Dans le cas où $m = 1$ nous sommes dans le cas des C-Means classiques.

Cependant ce paramètre joue un rôle de lissage. En effet, plus m est grand, moins les échantillons dont les valeurs d’appartenance sont uniformément faibles affectent l’algorithme. La nature des données, selon qu’elles soient faiblement bruitées ou non, peut donc guider le choix de façon heuristique. Ceci permet dans une certaine mesure, de limiter la sensibilité au bruit qui est un des problèmes majeurs de cette méthode.

3.4.2.2.3 Au niveau de l’initialisation Étant donné que la technique générale utilisée par l’initialisation est aléatoire, un mauvais choix de la partition initiale peut conduire à une solution finale peu pertinente. Le choix du nombre de prototypes est un problème certes ; mais comme nous l’avons vu, il est possible de fixer un nombre de prototypes plus important à l’initialisation puis de faire un regroupement des classes pouvant l’être. Notons que l’algorithme du FCM converge vers un minimum local de la fonction objective.

Aussi, chaque région est caractérisée par un centre, et les degrés d’appartenance sont calculés en faisant intervenir la distance Euclidienne d’où une forme nécessairement hyper-sphérique représentant les classes. Nous sommes donc dans l’incapacité de calculer la structure réelle des données. L’exemple de la figure (3.18) ci-dessous illustre cette situation :

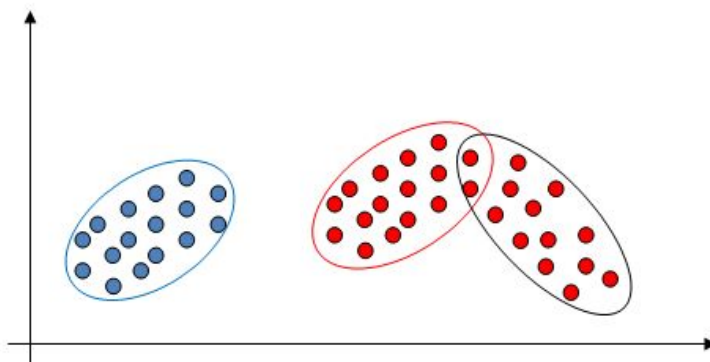


FIG. 3.18 – Exemple illustratif du problème hyper-sphérique représentant les classes.

Dans cet exemple nous disposons de deux groupes d'amas de points de deux couleurs (rouge et bleu), mais l'un des groupes est composé de deux amas de même couleur (rouge). Donc nous sommes dans l'incapacité de dire si nous disposons de 3 ou 2 classes seulement.

Une solution permettant de palier cet inconvénient consiste à considérer que toute forme arbitraire peut être approchée par une fusion de formes sphériques élémentaires figure (3.19) en mettant en place un processus de regroupement des classes obtenues à l'aide de l'algorithme du FCM.

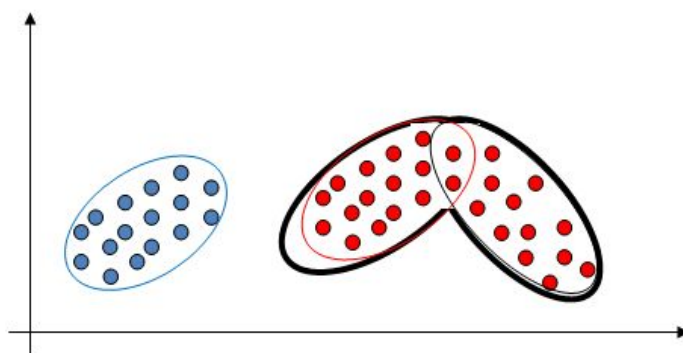


FIG. 3.19 – Fusion de formes sphériques élémentaires.

- **Critère de fusion** : Critère flou de compacité et chevauchement des classes. Intuitivement, deux classes doivent être fusionnées si elles sont proches et si leur réunion constitue une région homogène. Pour ce faire, deux quantités appelées : chevauchement et compacité sont définies (voir annexe 2).

Donc, deux classes sont fusionnées si nous estimons que leur chevauchement est assez élevé par rapport à un seuil dépendant naturellement des classes en question. Moins les classes sont compactes, moins nous sommes exigeant quand au seuil de fusion et on a tendance donc à fusionner et vice versa.

3.4.2.2.4 Forme des fonctions d'appartenance Nous observons que les fonctions d'appartenances obtenues par le FCM n'ont pas la forme de prémisses floues, des points très éloignés des centres de classes n'ont pas des degrés d'appartenance nuls. Ils tendent au contraire vers $\frac{1}{2}$. Cela est dû à la contrainte probabiliste qui impose que la somme des degrés d'appartenance soit égale à 1. (Équation (3.35)).

Il serait souhaitable que deux points équidistants du centre de la classe à laquelle ils appartiennent aient le même degré d'appartenance à cette classe. Ce n'est pas le cas avec le FCM, nous pouvons avoir des points équidistants qui n'ont pas le même degré d'appartenance, cela est dû à l'équation (3.37) qui est fonction de la distance du point à la classe et du point à toutes les autres

classes; de ce fait des singularités se produisent car deux points bien qu'équidistants à une classe ne le sont pas par rapport aux autres classes.

Si nous considérons l'exemple de la figure (3.20) où a est l'unité, donc $\mu_{x_A c_1} = \frac{1}{1+\sqrt{2}}$ et $\mu_{x_B c_1} = \frac{1}{5}$ or nous notons bien visuellement que les deux points x_A et x_B sont équidistants de la classe C_1 donc, que les degrés d'appartenance pour cette classe devraient être égaux. Dans le même ordre d'idée, deux points qui ont une distance au centre très différente peuvent avoir le même degré d'appartenance.

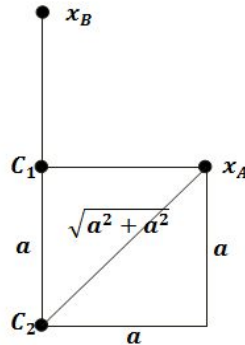


FIG. 3.20 – Mise en évidence du problème classification du FCM, la valeur de la fonction d'appartenance est différente pour x_A et x_B par rapport à C_1 alors que les points sont équidistants du centre de la classe.

3.4.2.2.5 Convergence du FCM[34] La minimisation du critère (J_m) n'est pas simple à cause des minima locaux. D'autre part la forme de ce critère met en échec la théorie de convergence de Bezdek, à cause de l'existence de « point selle » ou « col » figure (3.21) dans plusieurs contre exemples.



FIG. 3.21 – Exemple de point selle.

3.4.2.2.6 Mise en évidence des Cols avec le test d'extremum Considérons une fonction à une dimension $f(x)$, si $f(x)$ a un extremum¹ relatif à x_0 , alors $f'(x) = 0$, soit f n'est pas différentiable en x_0 . Soit le premier test ou le deuxième test de dérivée qui doivent être utilisés pour localiser l'extrema relatif de première sorte.

Une condition nécessaire pour que $f(x)$ ait un minimum à x_0 est :

¹Un extremum est soit un maximum ou un minimum

$$\begin{cases} f'(x_0) = 0 \\ \text{et} \\ f''(x_0) \geq 0 \end{cases} \quad (3.40)$$

Une condition suffisante est $f'(x_0) = 0$ et $f''(x_0) > 0$. Soit $f'(x_0) = 0$, $f''(x_0) = 0, \dots, f^n(x_0) = 0$, mais $f^{n+1}(x_0) \neq 0$. Alors $f(x)$ a un maximum relatif à x_0 si n est impaire et $f^{n+1}(x_0) < 0$ et $f(x)$ a un minimum relatif à x_0 si n est paire et $f^{n+1}(x_0) > 0$.

Il y a un point selle à x_0 si n est impaire. Cette situation est obtenue dans certaines circonstances à cause de la forme de μ_{ik} équation (3.37).

3.4.2.2.7 Résultats et discussion

3.4.2.2.7.1 Mise en évidence du problème de l'initialisation Nous avons procédé à un test sur les deux images de la figure (3.3) pour mettre en évidence l'influence de l'initialisation.

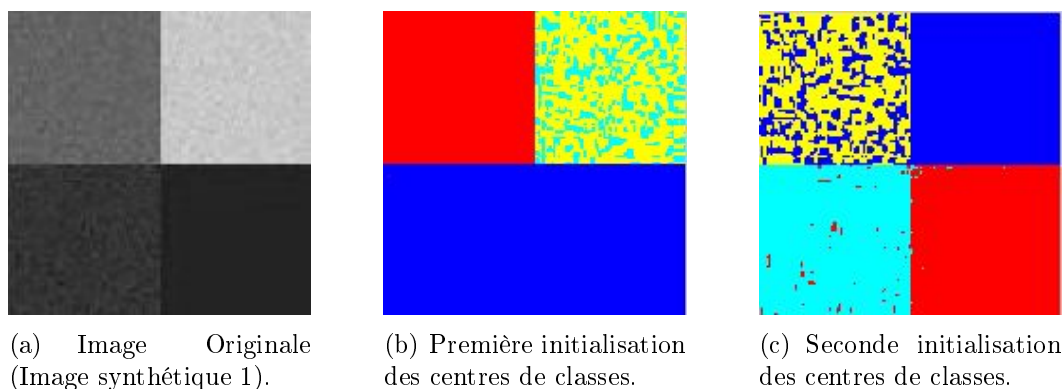


FIG. 3.22 – Mise en évidence du problème de l'initialisation (Image synthétique 1) .

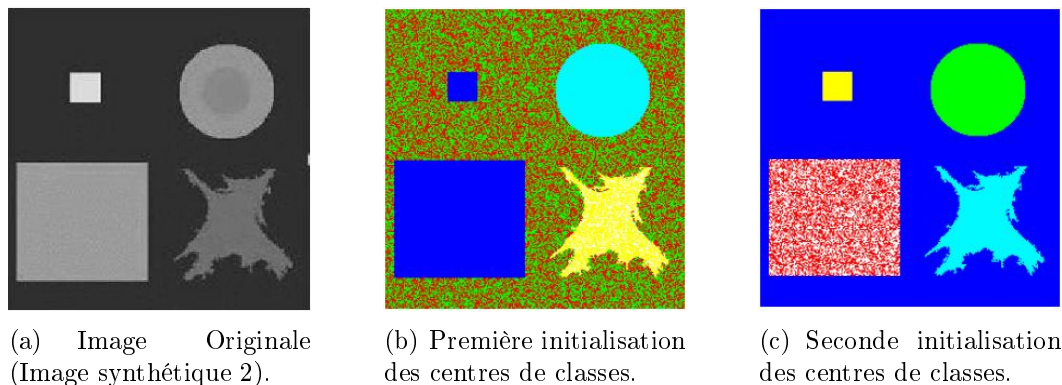


FIG. 3.23 – Mise en évidence du problème de l'initialisation (Image synthétique 2) .

En appliquant directement l’algorithme FCM sur les deux images précédentes, nous obtenons les résultats présentés en figure (3.22) et (3.23). Selon l’initialisation aléatoire des centres et selon l’ordre dans lequel sont pris les pixels, les résultats peuvent varier considérablement, comme le montre les deux images de la figure (3.23.b) et (3.23.c).

Lorsque les FCM ont construit les 6 classes, les pixels sont affectés à la classe dont le degré d’appartenance est le plus élevé. Afin de mieux visualiser les résultats, nous appliquons des fausses couleurs à l’image obtenue après la classification. Les résultats de cette classification sont affichés en affectant un label à chaque pixel sous la forme d’une couleur correspondant à la classes à laquelle il appartient. Ainsi pour la figure (3.23.b), les pixels de la régions R_2 sont assignés à deux classes différentes, tandis que les pixels des régions R_5 et R_6 sont assignés à la même classe.

En appliquant de nouveau l’algorithme des FCM sur l’image de synthèse (2), la nouvelle initialisation des centres des classes donne des résultats différents de ceux obtenu précédemment comme l’illustre l’image de la figure (3.23.c). En effet, les pixels constituant les régions R_5 et R_6 sont de nouveau assignés à la même classe. Pour obtenir une classification des pixels satisfaisante, il faudrait initialiser les centres aux plus proches des centres réels des différentes classes.

3.4.2.2.7.2 Fusion des classes Afin de bien illustrer le problème lié à la forme hypersphérique représentant les classes et l’incapacité de calculer la structure réelle des données, nous avons introduit des modifications sur les deux images de la figure (3.3) de sorte à avoir deux images couleurs (où toutes les régions présentent sur chacune des images sont colorées).

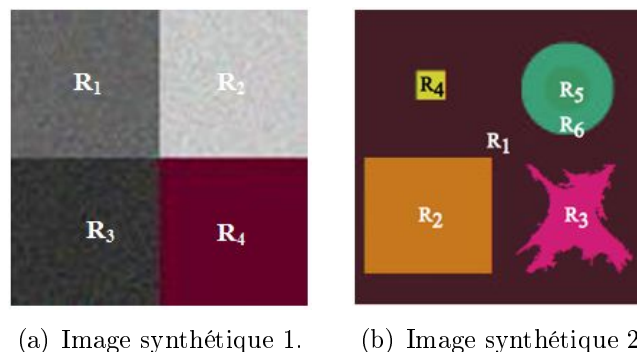
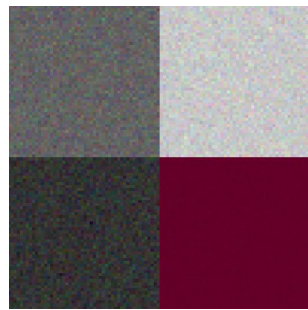
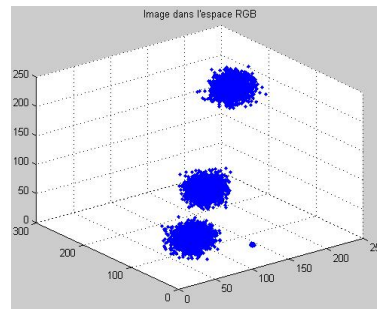


FIG. 3.24 – Images synthétiques modifiées.

Nous avons procédé par la suite à une fusion de formes sphériques élémentaires en mettant en place un processus de regroupement des classes obtenues l’aide de l’algorithme du FCM appliqué aux deux images de la figure (3.24) :



(a) Image Originale (Image synthétique 1).

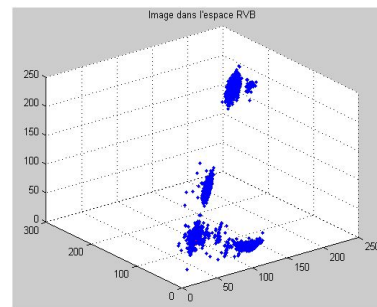


(b) Représentation RVB de l'image .

FIG. 3.25 – Représentation des classes dans l'espace RVB (Image synthétique 1) .

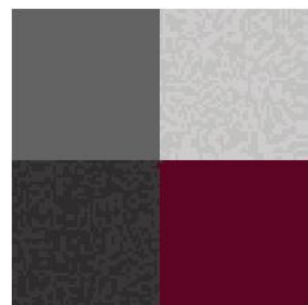


(a) Image Originale (Image synthétique 1).

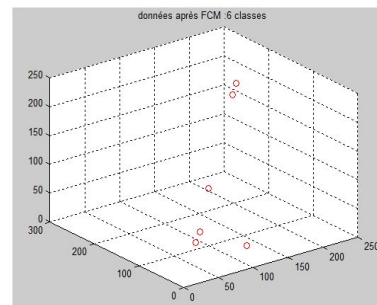


(b) Représentation RVB de l'image.

FIG. 3.26 – Représentation des classes dans l'espace RVB en initialisant $K=8$ classes .



(a) Segmentation initiale par FCM.



(b) Centres des classes initiales (FCM) $K = 8$.

FIG. 3.27 – Résultat de la segmentation initiale par les FCM .

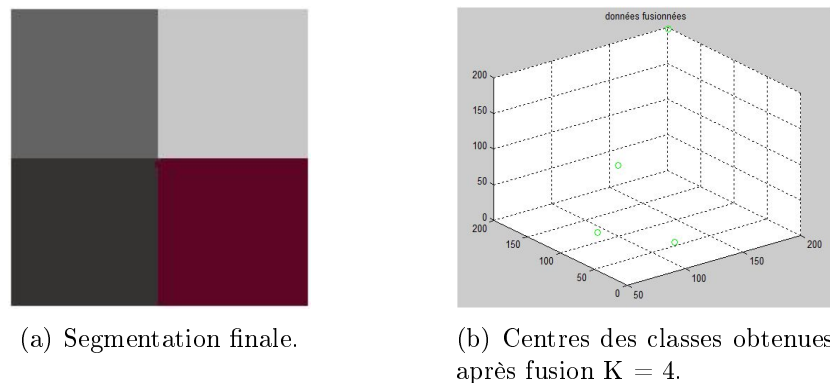


FIG. 3.28 – Résultat de la segmentation finale après fusion ($f_u = 0,05$) .

Signalons d’abord un fait très important : l’espace RVB n’est pas très représentatif de la perception des couleurs par l’œil humain .

En effet, la figure (3.25.a) présente l’image synthétique 1 et la figure (3.25.b) sa représentation dans l’espace des couleurs RVB. Contrairement à l’image originale, où nous percevons 4 types de couleurs différentes, l’image dans l’espace des couleurs ne contient que 3 classes significatives.

Cela dit, nous avons modifié encore une fois l’image synthétique 1 pour avoir une représentation appropriée dans l’espace des couleurs (figure (3.26.a)), ce qui permet de bien illustrer le processus de fusion.

Les figures (3.27.a) et (3.28.a) représentent respectivement la segmentation initiale de l’image de la figure (3.26), et celle obtenue après l’étape de fusion pour $f_u = 0,05$. La fusion, en diminuant le nombre de régions, permet d’obtenir une bonne segmentation en quatre régions homogènes. Elle se visualise mieux dans l’espace des couleurs (figures (3.27.b) et (3.28.b)). En effet, les classes très proches, représentées dans les figures par leurs centres, ont été fusionnées.

3.4.2.2.7.3 Influence du paramètre de fusion (f_u) Pour voir l'influence du paramètre de fusion (f_u), nous avons effectués des tests sur l'image synthétique 2 car elle contient plus de régions que l'image synthétique 1 :

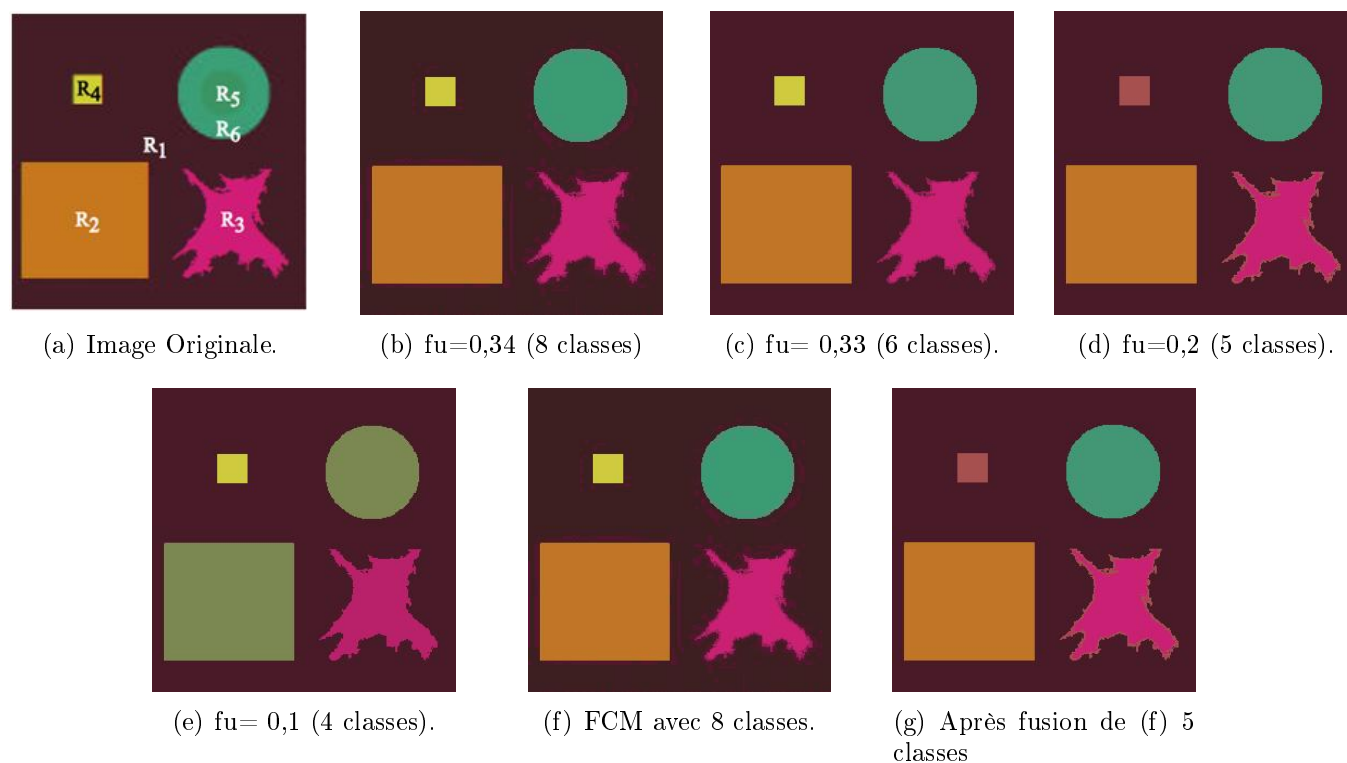


FIG. 3.29 – Résultats de la segmentation pour différente valeur du paramètre de fusion (f_u).

Le premier constat qui peut être fait est que le nombre de régions de la segmentation finale croît avec f_u . Le nombre de régions passe de 4 pour $f_u=0,1$, à 5 pour $f_u=0,2$, 6 pour $f_u=0,33$ et 8 pour $f_u=0,34$. Cela est tout à fait normal, car un seuil f_u petit tolère plus de fusion et par conséquent, le nombre de classes initiales issues du FCM diminue.

Par ailleurs, le choix de f_u peut être très sensible dans certains cas comme le montrent les deux figures (3.29.b) et (3.29.c) où une variation de l'ordre de 0,01 fait passer le nombre de régions de 6 à 8.

Nous pouvons remarquer l'effet de la fusion sur les figures (3.29.f) et (3.29.g), où la fusion a permis d'éliminer le flou tout autour du cercle vert (région R_6) et la forme en grenat (région R_3), donnant ainsi des classes plus précises et plus distinctes.

3.4.2.2.7.4 Sensibilité du FCM au bruit Afin de voir la sensibilité du FCM vis-à-vis du bruit, nous avons ajouté à l'image synthétique 2 de la figure (3.24) un bruit gaussien de moyenne nulle et de variance relativement faible puis de variance relative à la valeur moyenne de l'image :

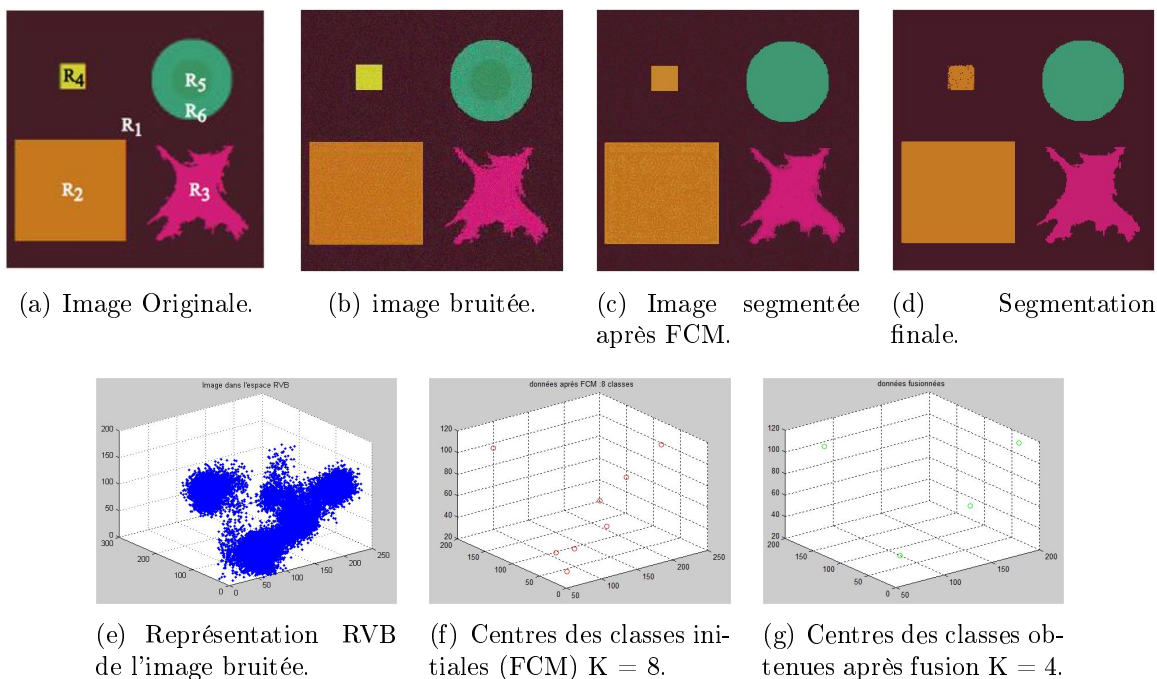


FIG. 3.30 – Résultats de la segmentation pour un bruit Gaussien ($m = 0, \sigma = 0,001$) et ($fu = 0,34$)

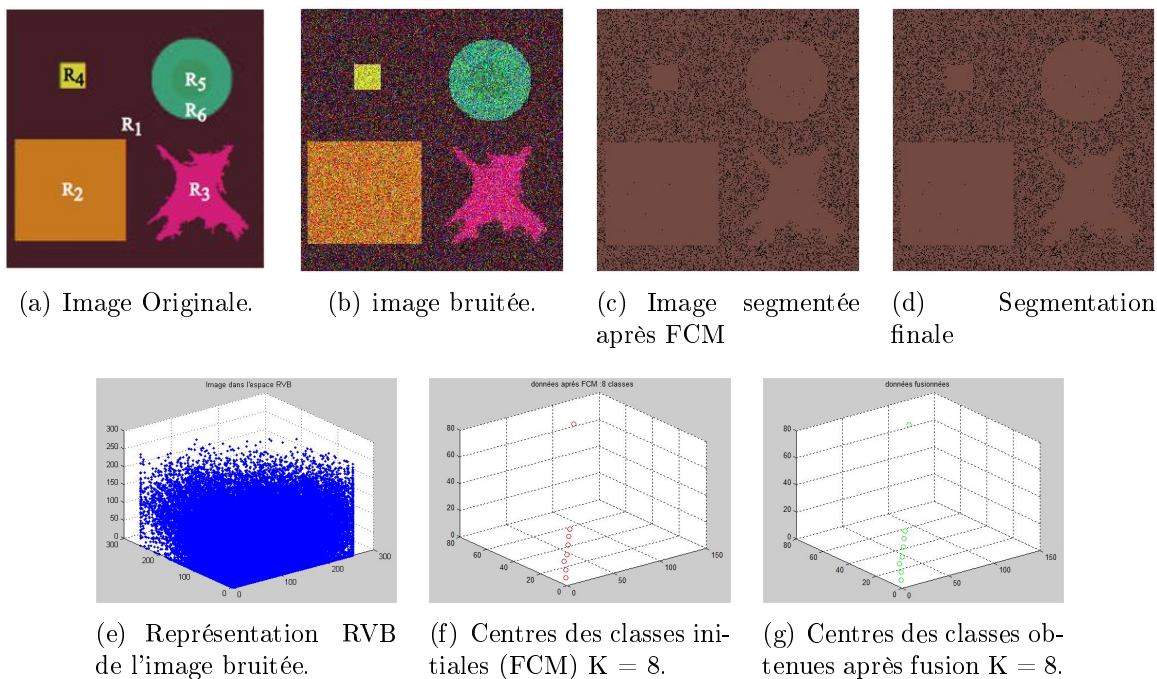


FIG. 3.31 – Résultats de la segmentation pour un bruit Gaussien ($m = 0, \sigma = 0,05$) et ($fu = 0,34$).

D'après les résultats obtenus, nous constatons que pour une image peu bruitée (figure (3.30.b)) le résultat de la segmentation est acceptable (figure (3.30.d)) par rapport à l'image fortement bruitée (figure (3.31.b)), où la segmentation finale est médiocre.

3.4.2.2.8 Avantages et inconvénients de l'algorithme (FCM) Le principal avantage de l'algorithme provient de l'introduction des degrés d'appartenance μ_{ik} . Grâce à eux le processus d'optimisation itératif est rendu beaucoup plus robuste notamment en permettant de prendre en compte les recouvrements entre les classes. Il permet ainsi d'obtenir des partitions plus pertinentes et plus proches de la réalité. En outre ces degrés permettent de prendre des décisions nuancées pour l'assignation d'une forme à une classe ce qui s'avère très intéressant pour toute forme de classification.

Parmi les autres avantages de l'algorithme, sa complexité algorithmique est relativement réduite par rapport à d'autres algorithmes de classification non supervisé, cela le rend plus facilement exploitable pour traiter des problèmes de taille importante (avec beaucoup de données).

Malgré tout, l'algorithme possède aussi quelques inconvénients. Nous pouvons citer par exemple le problème de la sensibilité à l'initialisation (différentes initialisations peuvent aboutir à différentes partitions), le manque de flexibilité sur la forme des classes qu'ils peut détecter. Dans les FCM la contrainte imposée aux degrés d'appartenance des individus fait que les classes sont établies les unes par rapport aux autres. La conséquence directe est la forme particulière des fonctions d'appartenance; elles sont définies les unes par rapport aux autres et traduisent la notion de "partage" des individus entre les classes.

Une autre conséquence de cette notion de partage est que l'algorithme reste assez sensible au bruit.

Afin de remédier à ces différents problèmes, il faut modifier le mode de fonctionnement de l'algorithme. L'algorithme des C moyennes possibilistes correspond à une telle évolution.

3.4.2.3 C-Moyennes Possibilistes (Possibilistic C-Means)

3.4.2.3.1 Introduction La classification possibiliste, a été récemment proposée par Krishnapuram et Keller [35], elle appartient à la famille des classifications non supervisées. Elle se différencie de l'approche des C-Moyennes floues, introduite précédemment, par une définition plus souple du concept de partition floue ainsi que en caractérisant les classes, non plus de manière relative les unes par rapport aux autres, mais au contraire de façon absolue [36].

3.4.2.3.2 Principe général des PCM La définition d'une partition floue, donnée en équation (3.35), impose que la somme des degrés d'appartenance d'un élément à l'ensemble des classes soit égale à 1, cette contrainte forte engendre un certain nombre de problèmes si nous voulons extraire les propriétés intrinsèques de chaque classe. Pour remédier à ces problèmes, Krishnapuram et Keller ont proposé de relâcher cette contrainte forte qui est à l'origine de ces comportements indésirables.

La partition floue utilisée dans le cadre de la classification possibiliste est alors définie de la manière suivante :

$$\forall i, k, \quad \mu_{ik} \in [0, 1] \quad (3.41)$$

$$\sum_{i=1}^c \mu_{ik} \leq 1 \quad , \forall k \in [1, n] \quad (3.42)$$

$$0 \leq \sum_{i=1}^c \mu_{ik} \leq 1, \quad \forall i \in [1, c] \quad (3.43)$$

La fonction objective à minimiser est redéfinie comme :

$$J_m(U, V) = \sum_{i=1}^c \sum_{k=1}^n \mu_{ik}^m d^2(x_k, V_i) + \sum_{i=1}^c \eta_i \sum_{k=1}^n (1 - \mu_{ik})^m \quad (3.44)$$

où η_i est un paramètre positif qui sera explicité par la suite.

Cette nouvelle fonction objective correspond à celle définie en (3.43) pour l'algorithme des FCM, à laquelle a été ajoutée un second terme contraignant les degrés d'appartenance à être les plus grands possibles.

L'ajout de cette contrainte compense la définition plus souple de la partition floue et évite ainsi les solutions triviales où tous les degrés d'appartenance sont égaux à 0.

La minimisation de la nouvelle fonction objective engendre alors une modification de la condition nécessaire utilisée pour la mise à jour des degrés d'appartenance qui sont maintenant calculés de la manière suivante [37] :

$$\mu_{ik} = \frac{1}{1 + \left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{1}{m-1}}} \quad (3.45)$$

– Paramètre η_i

Le paramètre positif η_i définit l'importance du second terme ajouté à la fonction objective (3.44) pour le processus itératif de minimisation. Il caractérise en fait la distance à laquelle le degré d'appartenance d'un élément à une classe est égal à 0.5. Ce paramètre peut être défini de plusieurs manières dans la mise en oeuvre de l'algorithme itératif de minimisation. Il est souvent estimé à l'initialisation de l'algorithme et éventuellement ré-estimé lors d'une seconde passe. Par exemple, il peut être choisi comme étant proportionnel à la distance moyenne intra classe [38] :

$$\eta_i = K \frac{\sum_{k=1}^n \mu_{ik}^m d^2(x_k, V_i)}{\sum_{k=1}^n \mu_{ik}^m} \quad (3.46)$$

Où K est le plus souvent pris égal à 1.

L'algorithme des PCM :

Algorithm 4 Algorithme général d'exécution du PCM**Étape 1 :**

- Choisir le nombre de classes : c
- Fixer m , $1 < m < \infty$
- Choisir ε pour le test d'arrêt.
- Initialiser ($t = 0$) la matrice de partition $\mu_{ik}^{t=0}$ des fonctions d'appartenance avec l'algorithme des FCM.
- Évaluer le paramètre η_i à partir de l'équation (3.46)

Étape 2 : Mise à jour du vecteur C-Mean v_i des centres en utilisant l'équation (3.38).

Étape 3 : Mettre à jour μ_{ik}^t en utilisant l'équation (3.45).

Étape 4 : Calculer $E^t = \max_{i,k} |\mu_{ik}^t - \mu_{ik}^{t-1}|$ si $E^t \leq \varepsilon$ alors arrêt sinon $t = t + 1$ et retourner à l'étape 2

3.4.2.3.3 Résultats et discussion Afin de classer les pixels des deux images de la figure (2.1), l'algorithme des PCM a été appliqué sur ces deux images en fixant les paramètres de l'algorithme :

- $c = 4$ classes pour l'image synthétique 1 et $c = 6$ classes pour l'image synthétique 2 .
- $m = 2$.
- $\varepsilon = 10^{-3}$.
- Bruit Gaussien ($m = 0, \sigma = 0,05$).

Les résultats de la figure (3.32) et (3.33) décrivent les régions obtenues :

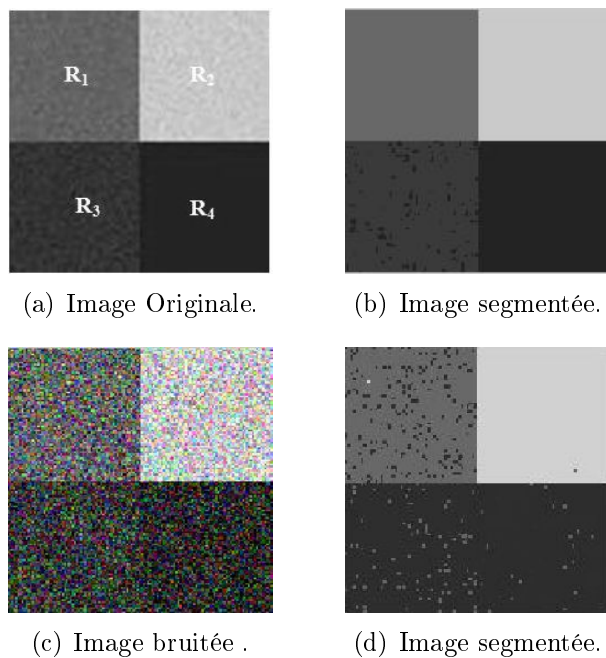


FIG. 3.32 – Résultats de la segmentation par l'algorithme des PCM (Image synthétique 1) .

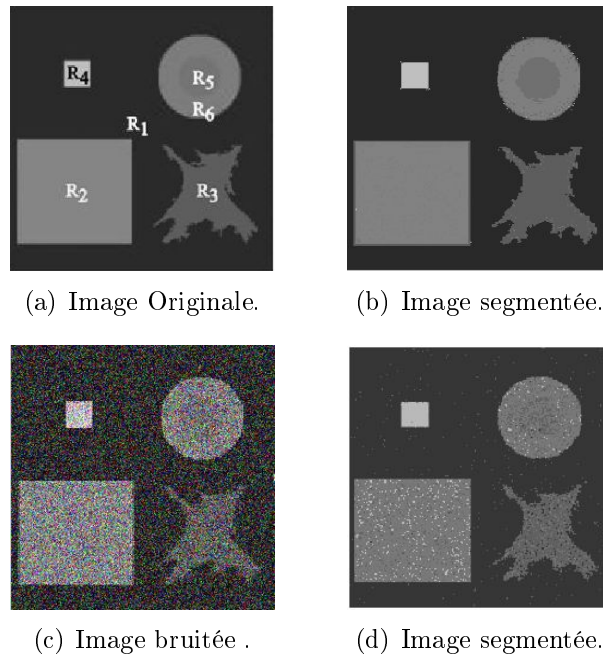


FIG. 3.33 – Résultats de la segmentation par l’algorithme des PCM (Image synthétique 2) .

Dans l’image de la figure (3.32.b), nous constatons que 4 classes de pixels ont été construites. Par ailleurs, nous remarquons que certaines classes contiennent des pixels provenant de différentes régions dans le cas où nous appliquons l’algorithme pour l’image bruitée (figure (3.32.d)).

Pour l’image de la figure (3.33.b), toutes les classes ont été détectées même la région R_5 qui est très proche de la région R_6 à été construite. Par contre pour l’image de la figure (3.33.d) représentant le résultat de la classification des pixels de l’image originale bruitée, nous constatons que le résultat est meilleur comparé à celui obtenu en appliquant l’algorithme des FCM (1 classe seulement est détectée).

Donc l’algorithme des PCM fourni des résultats assez satisfaisants comparé aux autres algorithmes présentés précédemment.

3.4.2.4 C-Moyennes Possibilistes Flous (Fuzzy Possibilistic C-Means (FPCM))

Pour remédier aux problèmes rencontrés par les FCM et les PCM, un nouveau Fuzzy Possibilistic C-mean (FPCM) algorithme a été proposé dans [39], en combinant ces deux algorithmes. La valeur de la fonction d’appartenance de la partition floue est importante afin de pouvoir attribuer une étiquette pour le vecteur d’entrée à classer. Mais en même temps, il est très utile d’utiliser la typicité (possibilité) pour déplacer les centres autour des vecteurs d’entrée, en évitant les effets indésirables dus à la présence de valeurs aberrantes .

La fonction objective qui devrait être minimisée est :

$$J_m(U, V, T) = \sum_{i=1}^c \sum_{k=1}^n (\mu_{ik}^m + t_{ik}^\lambda) d^2(x_k, V_i) \quad (3.47)$$

avec les contraintes :

$$\forall i, k, \mu_{ik} \in [0, 1] \quad (3.48)$$

$$\sum_{i=1}^c \mu_{ik} = 1, \quad \forall k \in [1, c] \quad (3.49)$$

$$\sum_{k=1}^n t_{ik} = 1, \quad \forall i \in [1, n] \quad (3.50)$$

avec : t_{ik} : typicité et nous définissons $T = [t_{ik}]$ la matrice de typicité.

La minimisation de la nouvelle fonction objective engendre alors :

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}} \quad (3.51)$$

et pour la typicité :

$$t_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{\lambda-1}}} \quad (3.52)$$

avec $\lambda \in [3, 5]$

La fonction de mise à jour des centres est :

$$V_i = \frac{\sum_{k=1}^n (\mu_{ik}^m + t_{ik}^\lambda) x_k}{\sum_{k=1}^n (\mu_{ik}^m + t_{ik}^\lambda)} \quad (3.53)$$

L'algorithme des FPCM est :

Algorithm 5 Algorithme général d'exécution du FPCM

Étape 1 :

- Choisir le nombre de classes : c
- Fixer m , $1 < m < \infty$
- Fixer λ , $\lambda \in [3, 5]$
- Choisir ε pour le test d'arrêt.
- Initialiser ($t = 0$) la matrice de partition $\mu_{ik}^{t=0}$ des fonctions d'appartenance avec l'algorithme des FCM.

Étape 2 : Mise à jour du vecteur C-Mean v_i des centres en utilisant l'équation (3.53).

Étape 3 : Mettre à jour μ_{ik} et t_{ik} en utilisant respectivement l'équation (3.51 et 3.52).

Étape 4 : Calculer $E^t = \max_{i,k} |\mu_{ik}^t - \mu_{ik}^{t-1}|$ si $E^t \leq \varepsilon$ alors arrêt sinon $t = t + 1$ et retourner à l'étape 2.

3.4.2.4.1 Résultats et discussion

Les paramètres de l'algorithme sont :

- $c = 4$ classes pour l'image synthétique 1 et $c = 6$ classes pour l'image synthétique 2 .
- $m = 2$. • $\lambda = 4$. • $\varepsilon = 10^{-3}$. • Bruit Gaussien ($m = 0, \sigma = 0,05$)

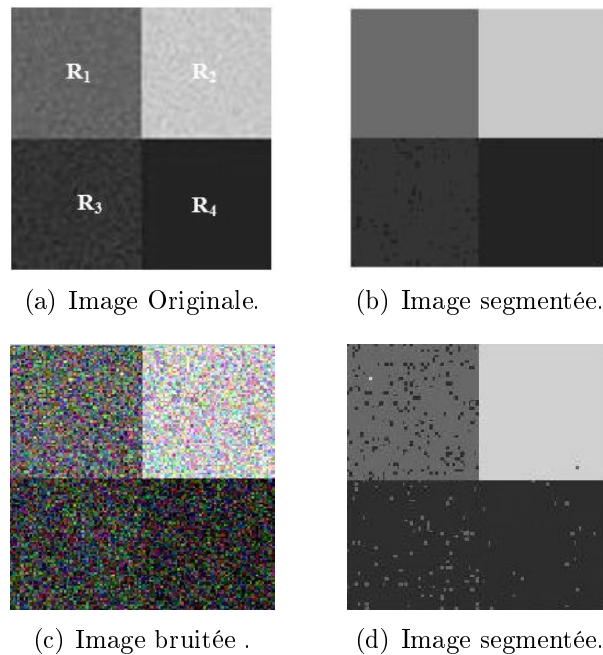


FIG. 3.34 – Résultats de la segmentation par l'algorithme des FPCM (Image synthétique 1) .

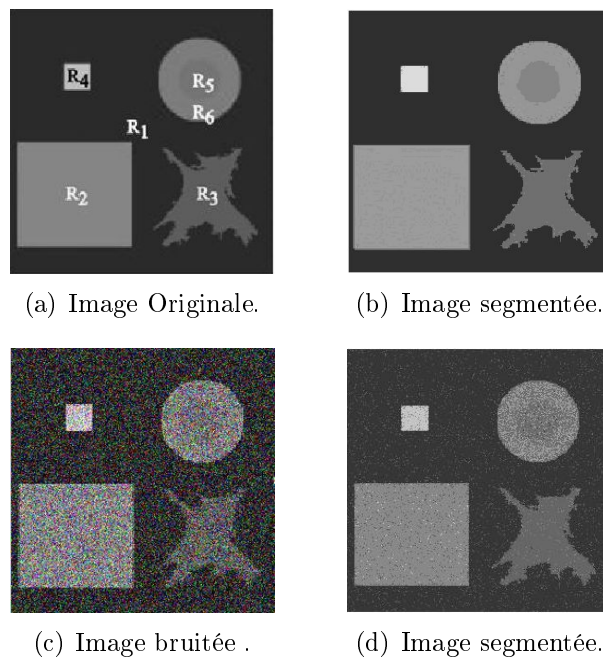


FIG. 3.35 – Résultats de la segmentation par l'algorithme des FPCM (Image synthétique 2) .

Dans l'image de la figure (3.34.b), nous constatons que 4 classes de pixels ont été bien construites. Par ailleurs, nous remarquons que certaines classes contiennent des pixels provenant de différentes régions dans le cas où nous appliquons l'algorithme pour l'image bruitée (figure (3.34.d)).

Pour l'image de la figure (3.35.b), toutes les classes ont été détectés même la région R_5 qui est très proche de la région R_6 à été construite. Par contre pour l'image de la figure (3.35.d) représentant le résultat de la classification des pixels de l'image originale bruitée, nous constatons que le résultat est meilleur comparé à celui obtenu en appliquant l'algorithme des FCM et PCM .

Donc l'algorithme des FPCM fourni des résultats assez satisfaisants comparé aux autres algorithmes présentés précédemment .

3.5 Conclusion

Dans ce chapitre, nous avons présenté un certain nombre de techniques de segmentation d'images. Nous nous sommes intéressés aux méthodes de segmentation d'images effectuant une classification des pixels par analyse de la distribution des niveaux de gris. Ces méthodes de classification considèrent qu'une région est constituée par un ensemble de pixels connexes qui appartiennent à une même classe.

La plupart de ces méthodes analysent globalement l'image et nécessitent soit de connaître au préalable le nombre de classes, soit de régler finement des paramètres afin de construire les classes de pixels constituant l'image à analyser.

L'approche par classification est efficace lorsqu'elle définit une partition de l'image en un ensemble de classes telle que chaque classe rassemble des pixels ayant des vecteurs de caractéristiques (souvent réduit au niveau de gris) aussi similaires que possible et que les classes soient aussi distinctes que possible les unes des autres. C'est généralement le cas lorsque les images sont constituées d'un faible nombre de régions ayant des niveaux de gris bien différenciées. Cependant, pour certaines images, le nombre de régions peut être élevé et les niveaux de gris des régions peuvent être proches les unes des autres. Les nuages de points se chevauchent, ce qui peut conduire à une classification incorrecte.

Généralement, ces méthodes de segmentation ne prennent pas en compte la disposition spatiale des pixels et ne considèrent que le vecteur de caractéristiques utilisé. En effet, les propriétés spatiales ne sont pas prises en compte dans le processus de segmentation et n'interviennent que dans un deuxième temps afin de construire les régions constituées de pixels connexes appartenant à la même classe. Ainsi, lors de la classification, le contexte spatial est perdu, et des nuages de points dans l'espace des niveaux de gris peuvent ne pas correspondre à des régions significatives dans l'image analysée. De plus , il est parfois délicat de décider qu'un pixel appartient ou non à une classe donnée. C'est pourquoi les méthodes de classification floue considèrent que chaque pixels appartient plus au moins aux différentes classes en présence. Les degrés d'appartenance de chaque pixel à chacune des classes sont évalués à partir de la distance niveaux de gris dans l'espace.

Comme nous l'avons présenté, l'algorithme des Fuzzy C-Means et les C-Moyennes Possibilistes (et Fuzzy C-Moyennes Possibilistes qui est une combinaison entre les FCM et PCM) restent les

algorithmes de références des méthodes de classification floue de pixels. Ces algorithmes sont bien adaptés à l'identification de classes et sont d'autant plus performants que nous connaissons le nombre de classes a priori.

Chapitre 4

Application à l'imagerie par résonance magnétique (IRM)

L

a segmentation des tissus du cerveau à partir de données médicales est une problématique importante pour de nombreuses applications, notamment liées au diagnostic et au suivi de maladies neurodégénératives. Elle peut aussi être utilisée comme étape préliminaire pour d'autres processus d'analyse d'images complexes. La segmentation manuelle de ces tissus, bien que théoriquement réalisable, est toutefois, d'un point de vue pratique, n'est pas envisageable en des temps raisonnables. Il est donc fondamental de pouvoir disposer de techniques automatiques, capables de fournir des résultats similaires à ceux d'un expert, donc intégrant les connaissances a priori similaires à celles sur lesquelles un tel expert se baserait.

Dans ce chapitre, nous allons appliquer et comparer les approches de classification décrites dans le chapitre précédent sur une base de données "Brainweb" constituée d'images cérébrales IRM en vue d'une classification des différentes classes correspondant aux différents tissus et structures.

4.1 Introduction

L'imagerie par résonance magnétique (IRM) constitue une des avancées les plus significatives en médecine ; elle est issue des travaux sur les propriétés magnétiques des noyaux des atomes menés par F. Bloch et E. Purcell [40] en 1946. Aujourd'hui, les images IRM qui exploitent les propriétés magnétiques des éléments constitutifs de la matière et se fonde sur les phénomènes physiques de résonance et de relaxation (**Voir annexe 3**), sont pour les médecins une source d'informations précieuse et conséquente, permettant l'analyse et le suivi d'une tumeur ou d'une pathologie. En effet, ce type d'imagerie fournit, de façon non invasive, des représentations contrastées et de haute résolution de l'anatomie cérébrale, permettant ainsi son observation *in vivo*¹. Cependant, l'étude systématique et « manuelle » de clichés est longue et fastidieuse, et souffre de la variabilité d'interprétation des spécialistes. Ainsi, de nombreuses recherches ont été menées afin d'automatiser l'analyse des images ; la segmentation en forme le point de départ.

4.1.1 Segmentation des tissus cérébraux (MB, MG, LCR)

La segmentation des tissus du cerveau à partir de données médicales est une problématique importante pour de nombreuses applications, notamment liées au diagnostic et au suivi de maladies neurodégénératives. Elle peut aussi être utilisée comme étape préliminaire pour d'autres processus d'analyse d'images complexes.

Dans ce qui suit, nous allons introduire les notions de base concernant l'anatomie cérébrale afin de comprendre les structures cérébrales que nous souhaitons segmenter en différentes classes (**Matière Blanche**, **Matière Grise** et **Liquide Céphalo-Rachidien**) en appliquant les approches de classification décrites dans le chapitre précédent.

4.1.1.1 Anatomie cérébrale [40]

Le système nerveux central est constitué de la moelle épinière logée dans le canal vertébral, et de l'encéphale.

4.1.1.1.1 Encéphale L'encéphale est constitué du tronc cérébral, du cervelet et du cerveau. L'ensemble flotte dans un coussin protecteur de liquide céphalo-rachidien (**LCR**). Le tronc cérébral est situé dans la fosse postérieure du crâne, sous le cerveau. Il représente un carrefour majeur du système nerveux central entre le cerveau, en haut, le cervelet, en arrière, et la moelle épinière, en bas. Le cervelet (figure 4.1) est le centre nerveux chargé de la coordination des mouvements. C'est la deuxième plus grosse structure de l'encéphale après le cortex. Il est issu de la partie dorsale du métencéphale et est relié au reste de l'encéphale par les pédoncules cérébelleux. Le cerveau (figure 4.1) est constitué de deux hémisphères séparés par un profond sillon médian. Les deux hémisphères sont reliés par des faisceaux de fibres appelés commissures.

¹C'est une expression latine qualifiant des recherches ou des examens pratiqués sur un organisme vivant.

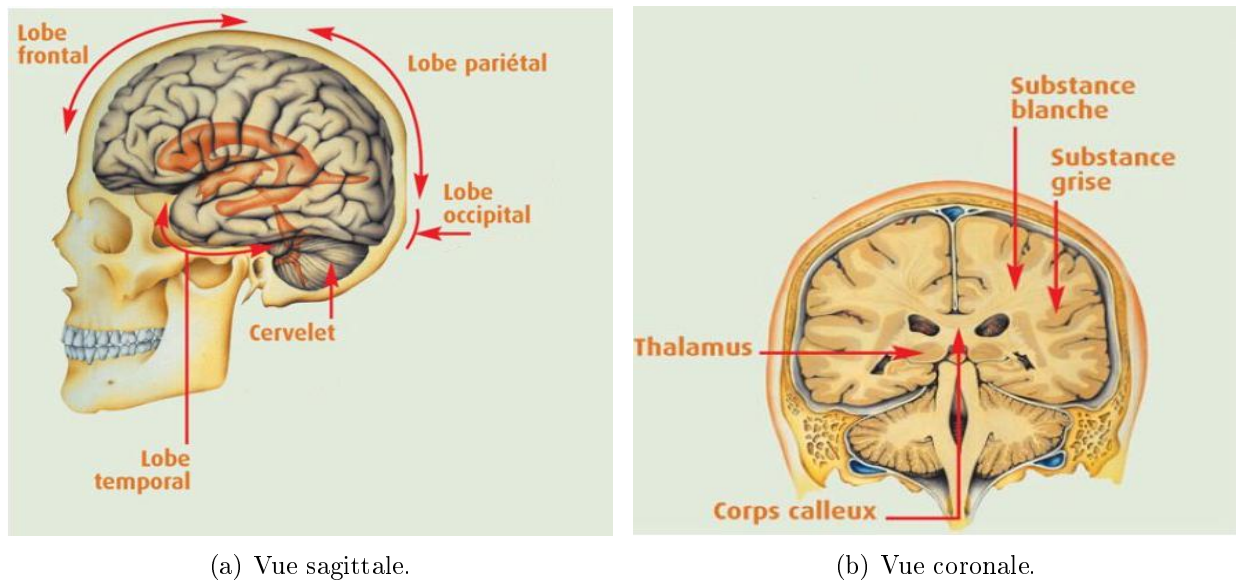


FIG. 4.1 – Cerveau humain

4.1.1.1.2 Tissus du cerveau Lorsque nous observons cerveau, il est principalement constitué d'une substance blanchâtre, d'où le nom de matière blanche, et à la surface du cerveau se trouve une fine pellicule de substance grisâtre : le cortex cérébral, ou matière grise. Cette différence de couleur provient du fait que le cortex contient principalement les corps cellulaires des neurones, alors que la substance blanche est constituée d'un agglomérat d'axones qui relient les différentes aires corticales les unes aux autres. Les axones sont responsables de la transmission du flux nerveux.

4.1.1.1.2.1 Matière grise (MG) La matière grise (figure 4.1) est répartie en deux types de structures : le cortex et les noyaux. D'un point de vue topologique, le cortex peut être représenté par une sphère dont l'épaisseur varie de 4 à 5 mm et recouvre une surface totale de plus de $2m^2$. Afin de pouvoir loger dans la boîte crânienne le cortex est replié formant des arêtes (circonvolutions) et des vallées (fissures). Le cortex cérébral de l'homme contient environ 80% des corps cellulaires de tout le système nerveux. La forme et la position des plis corticaux sont très variables d'un individu à l'autre. Les noyaux gris sont des regroupements de substance grise, situés à l'intérieur de l'encéphale. Ils sont constitués du noyau caudé, du putamen, du globus pallidus, du corps de Luys (noyau sous thalamique) et de la substance noire.

4.1.1.1.2.2 Matière blanche (MB) La matière blanche (figure 4.1) est comprise entre le cortex, les noyaux gris centraux et les ventricules. Elle est composée d'axones qui établissent les connexions entre les corps cellulaires du cortex et d'autres parties du cerveau.

4.1.1.1.2.3 Liquide céphalo-rachidien (LCR) Le liquide céphalo-rachidien est une substance transparente constituée de 99% d'eau. Il forme un coussin protecteur pour le système nerveux central (SNC) et le nourrit. Le LCR est contenu dans le SNC, dans deux zones anatomiques bien distinctes :

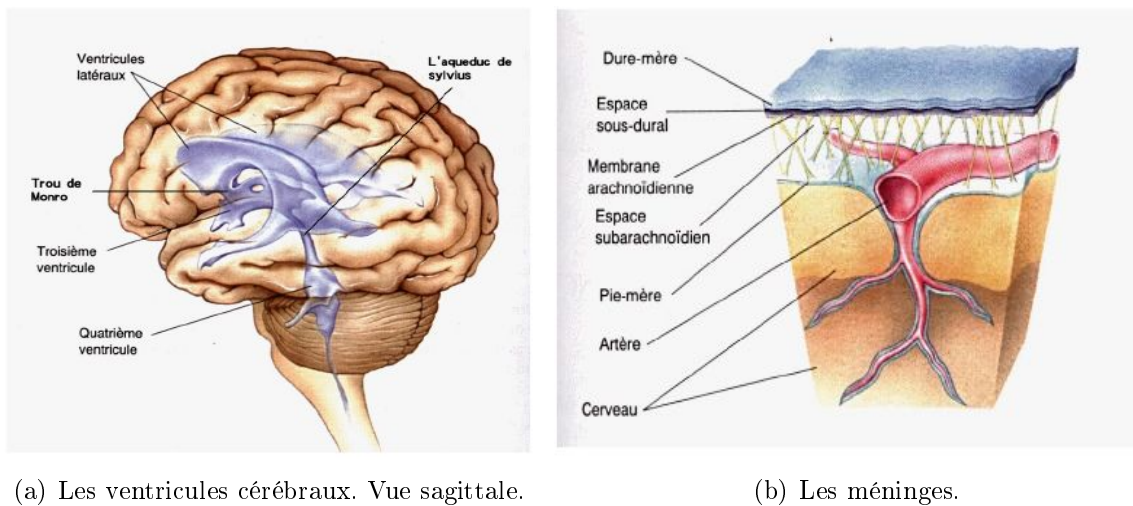


FIG. 4.2 – Localisation du LCR.

- Le système interne constitué par les ventricules cérébraux (figure 4.2.a), qui sont au nombre de quatre : les ventricules latéraux, un dans chaque hémisphère cérébral, le troisième ventricule entre les deux thalamus (figure 4.1), et le quatrième ventricule en arrière du tronc cérébral. Les ventricules latéraux communiquent avec le troisième ventricule via le trou de Monro (figure 4.2.a), et le quatrième ventricule avec le troisième ventricule par l'intermédiaire de l'aqueduc de Sylvius (figure 4.2.a). Ce sont les plexus choroïdes contenus dans les ventricules cérébraux qui sécrètent le liquide céphalo-rachidien.
- Le système externe appelé également système de résorption : il s'agit des cavités situées en dessous de l'arachnoïde (figure 4.2.b) (cette cavité est plus précisément située entre la pie-mère et l'arachnoïde).

4.1.1.2 Résultats et discussion

4.1.1.2.1 Présentation de la base de données utilisée (Brainweb) Pour appliquer et comparer les approches de classification décrites dans le chapitre précédent, nous nous basons sur la base de données Brainweb² de l'Institut Neurologique de l'Université McGill de Montréal. Cette base a été choisie dans la mesure où elle est très fréquemment utilisée et permet donc de pouvoir fournir un point de comparaison plus aisé. Le site Web de Brainweb permet de simuler des IRM cérébrales avec différents niveaux de bruit et d'inhomogénéités. Ces simulations sont générées à partir d'une vérité terrain qui correspond à la classification d'une image cérébrale (de résolution millimétrique) en différentes classes correspondant aux différents tissus et structures.

4.1.1.2.2 Résultats Pour nos évaluations, nous avons ainsi constitué différents jeux de données, faisant varier les niveaux de bruit. Un jeu particulier est constitué des pondérations en T_1 pour chaque niveau de bruit (3%, 5% et 9%). Pour bien comparer les résultats obtenus avec les

²<http://www.bic.mni.mcgill.ca/cgi>

méthodes de classification décrites précédemment (K-Means , FCM ,PCM et FPCM), nous jugeons la qualité de la segmentation obtenue par rapport à un estimateur souvent utilisé dans la littérature [40] : Le MCR (Miss Classification Ratio) donné par :

$$MCR = \frac{\text{nombre de pixels non classes}}{\text{nombre total de pixels}} \quad (4.1)$$

Les algorithmes ont été implémenté sous Matlab 7.4 sur un PC P4 2.8GHz avec 1GO de RAM, les temps d'exécutions des différents algorithmes ont été aussi évalués.

4.1.1.2.2.1 Choix des paramètres des algorithmes utilisés Pour segmenter les tissus cérébraux dans les images IRM, il faut définir les différents paramètres gouvernant ces algorithmes, à savoir les valeurs de m et le nombre de classes C , et enfin les vecteurs forme représentant les pixels des images.

– **Détermination du nombre de classes**

Notre objectif consiste à segmenter l'encéphale, ce qui nous pousse à fixer à 3 le nombre de classe à identifier ($C = 3$) correspondant aux trois tissus cérébraux présents dans l'encéphale (MB, MG, LCR) .

– **Initialisation de l'algorithme**

Les résultats des algorithmes vont évidemment dépendre de l'étape d'initialisation. Ainsi, l'initialisation, si elle doit exister, ne doit pas être aussi précise que dans d'autres algorithmes comme FCM . Tout algorithme flou ou non flou peut donc être utilisé, et FCM constitue une manière adéquate d'initialiser les données, puisqu'il donne accès à une première estimation de U et de V .

– **Choix des vecteurs forme**

Le choix des vecteurs forme est fondamental puisque leurs pertinences va permettre de discriminer les pixels entre eux. Ce choix est défini suivant le type de modalité. Le vecteur forme x_j d'un pixel j est alors formé des niveaux de gris de ce pixel .

4.1.1.2.2.2 Image à 0% de bruit

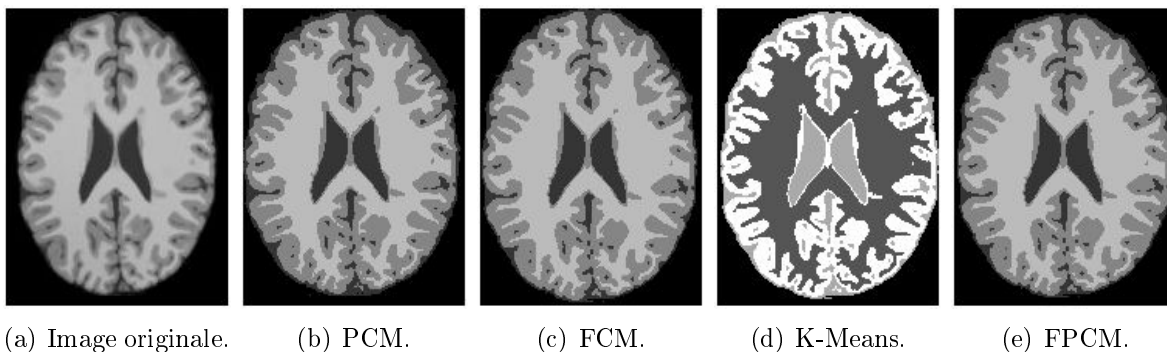
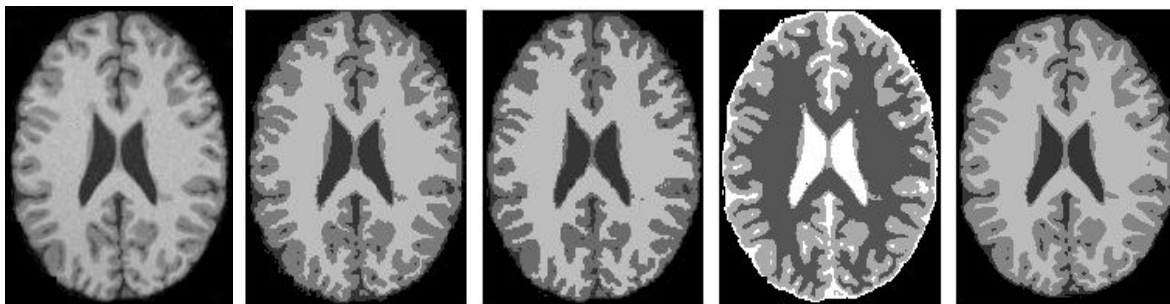


FIG. 4.3 – Résultats obtenus sur une coupe axiale pondérée en T_1 à 0% de bruit .

Méthode	MCR(%)	Temps d'exécution (s)
PCM	4,17	0.219234
FCM	5,28	0.155152
K-Means	6,08	2.332648
FPCM	3,23	0.35641

TAB. 4.1 – Misclassification Ratio (MCR) pour l'IRM à 0% de bruit

4.1.1.2.2.3 Image à 3% de bruit



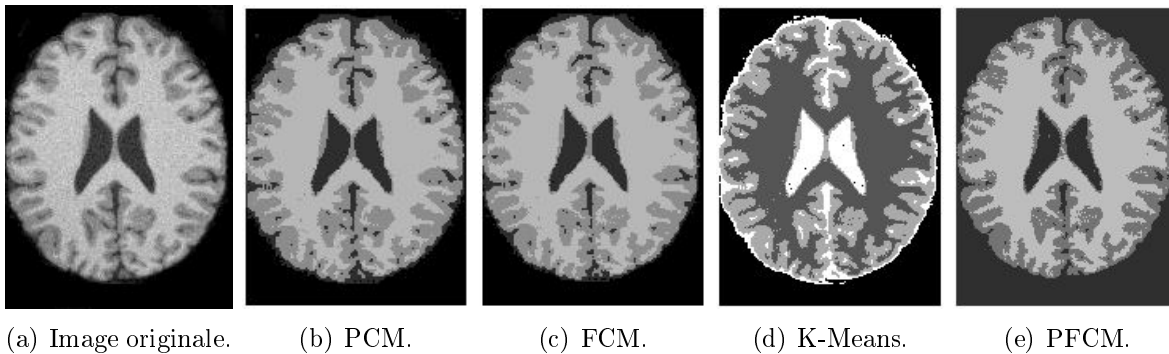
(a) Image originale. (b) PCM. (c) FCM. (d) K-Means. (e) PFCM.

FIG. 4.4 – Résultats obtenus sur une coupe axiale pondérée en T_1 à 3% de bruit .

Méthode	MCR(%)	Temps d'exécution (s)
PCM	5,43	0.275004
FCM	6,22	0.226401
K-Means	8,12	3.278916
FPCM	4,56	0.426232

TAB. 4.2 – Misclassification Ratio (MCR) pour l'IRM à 3% de bruit .

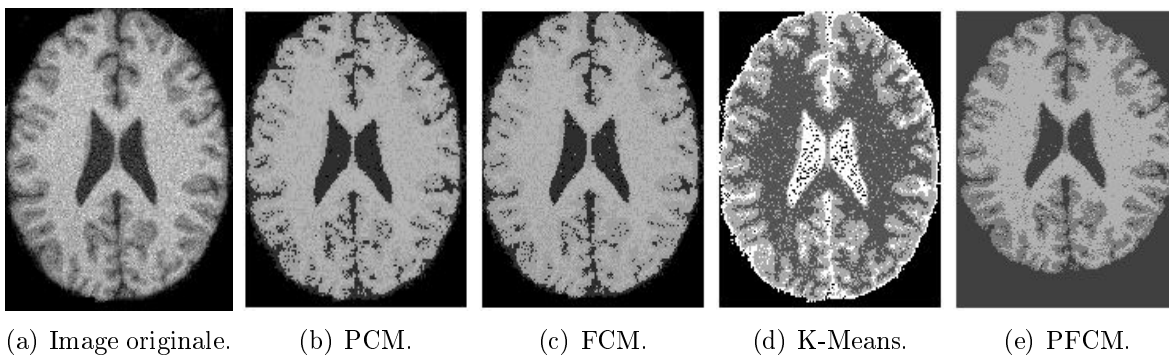
4.1.1.2.2.4 Image à 5% de bruit

FIG. 4.5 – Résultats obtenus sur une coupe axiale pondérée en T_1 à 5% de bruit .

Méthode	MCR(%)	Temps d'exécution (s)
PCM	6,85	0.267822
FCM	7,92	0.336419
K-Means	10,23	3.288359
FPCM	5,02	0.43279

TAB. 4.3 – Misclassification Ratio (MCR) pour l'IRM à 5% de bruit

4.1.1.2.2.5 Image à 9% de bruit

FIG. 4.6 – Résultats obtenus sur une coupe axiale pondérée en T_1 à 9% de bruit .

Méthode	MCR(%)	Temps d'exécution (s)
PCM	15,94	0.252168
FCM	18,26	0.232224
K-Means	26,32	3.275690
FPCM	08,48	0.446237

TAB. 4.4 – Misclassification Ratio (MCR) pour l'IRM à 9% de bruit .

Les figures (4.3,4.4,4.5 et 4.6) illustrent les résultats obtenus après segmentation des différents tissus cérébraux d'images pondérées en T_1 sur des coupes axiales .

Pour la figure (4.3) où l'image originale ne contient aucun bruit , nous constatons que la qualité des segmentations obtenues pour les différents algorithmes est satisfaisante. Dans ces images (b, c, d, e) la distinction entre les 3 classes du cerveau (MB, MG, LCR) est remarquable. Par contre le temps d'exécution est assez considérable pour la méthode des K-Means et très faible pour les deux méthodes PCM, FCM et FPCM qui s'exécutent plus rapidement par rapport au "K-Means" (voir tableau 4.1).

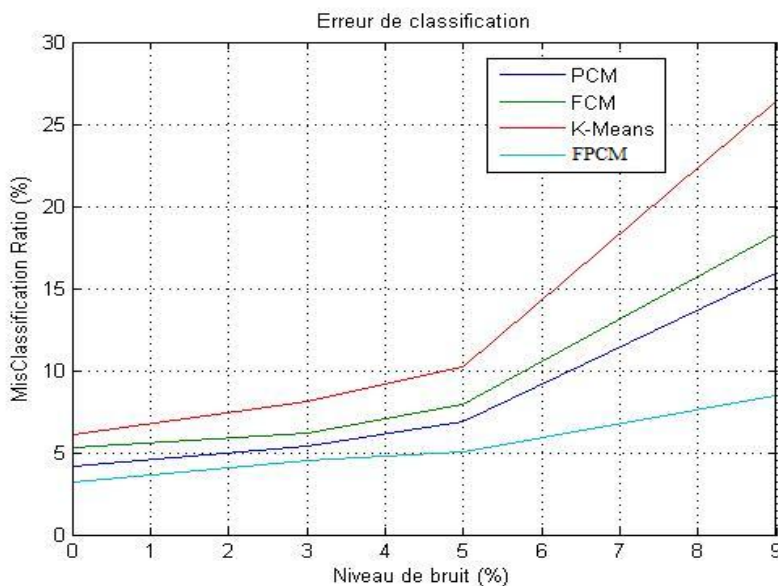


FIG. 4.7 – Comparaison du taux d'erreur de classification (MCR) des algorithmes.

L'observation des seuls résultats quantitatifs (tableaux 4.2, 4.3, 4.4) mettent en évidence certains points notables. Tout d'abord la qualité des segmentations obtenues pour les différents tissus est relativement bonne dans le cas des images présentant très peu de bruit (figure 4.4 et 4.5) surtout pour la méthode PFCM, PCM et FCM, mais se détériore progressivement à mesure que la quantité de bruit augmente (figure 4.6). Ensuite le temps d'exécution qui est plus important pour la méthode K-Means (soit plus de 14 fois plus que pour les méthodes PCM et FCM).

4.1.1.2.3 Amélioration du FPCM , PCM et FCM Afin d'améliorer la sensibilité des deux algorithmes FPCM, PCM et FCM au bruit, nous utilisons l'algorithme du Mean Shift comme chaîne de prétraitement en le combinant avec les algorithmes FPCM, PCM et FCM

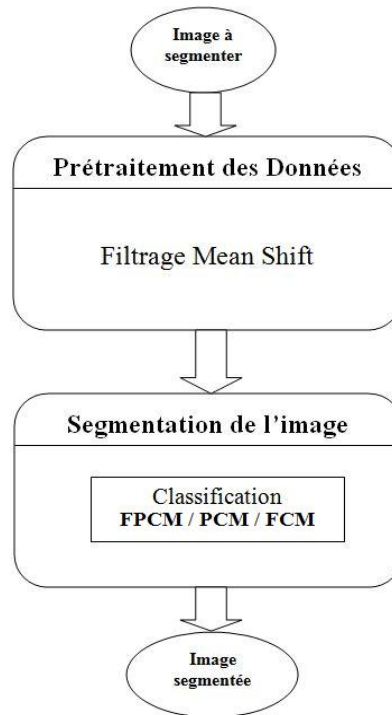


FIG. 4.8 – Chaîne de prétraitement de l'image.

Nous reprenons le cas de l'image IRM à 9% de bruit et nous procédons à un prétraitement de l'image avec l'algorithme du Mean Shift. Ce dernier permet dans un premier temps de lisser l'image et d'obtenir ainsi une première partition, puis une classification est effectuée avec le FPCM, PCM ou FCM (figure 4.9).

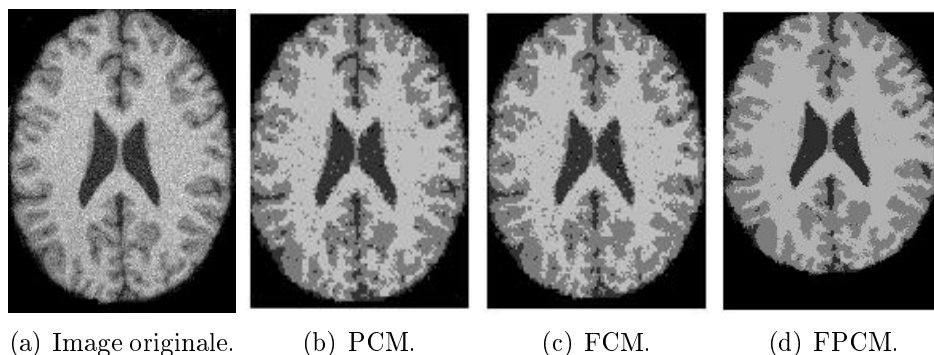


FIG. 4.9 – Résultats obtenus sur une coupe axiale pondérée en T_1 à 9% de bruit en utilisant le prétraitement .

4.1.1.2.4 Extraction des classes obtenues Après avoir obtenu une classification de l'image IRM à 9% de bruit avec l'algorithme FPCM , PCM ou FCM, nous procedons à une extraction des classes correspondant aux tissus afin de les comparer avec une vérité terrain (segmentation de référence des différents tissus) :

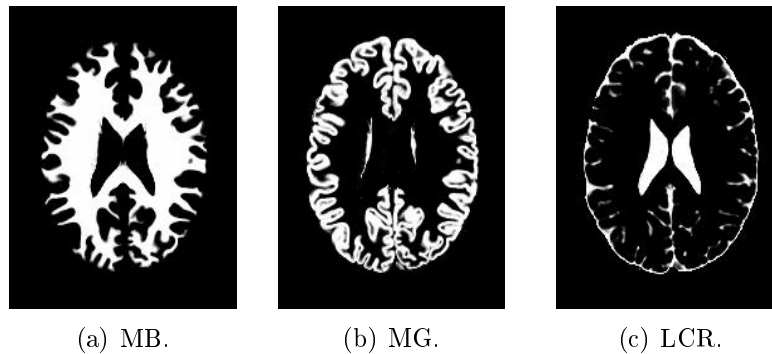


FIG. 4.10 – Classification de référence (MB, MG, LCR) de la base de données Brainweb.

Les figures (4.11 , 4.12 et 4.13) illustrent les résultats obtenus après extraction des différentes classes des tissus cérébraux d'images pondérées en T_1 sur des coupes axiales IRM à 9% de bruit .

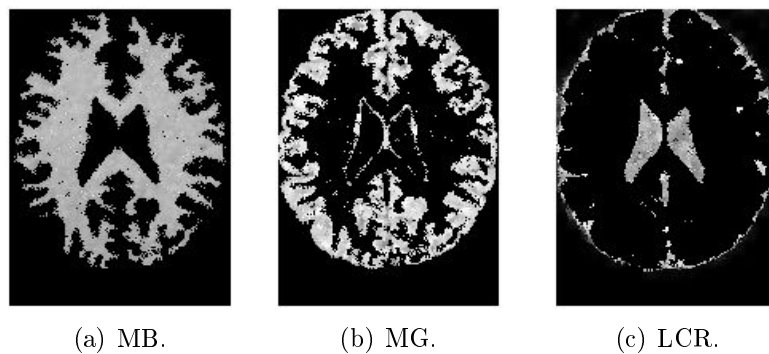


FIG. 4.11 – Classes obtenues (MB, MG, LCR) avec le FCM.

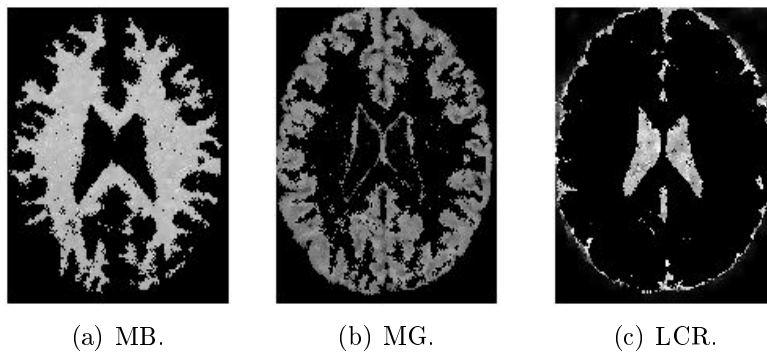


FIG. 4.12 – Classes obtenues (MB, MG, LCR) avec le PCM.

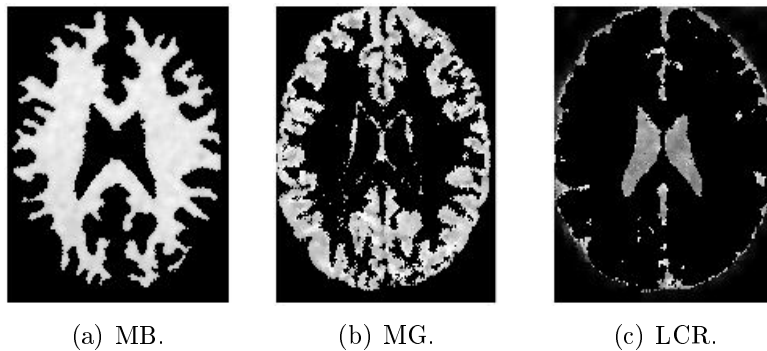


FIG. 4.13 – Classes obtenues (MB, MG, LCR) avec le FPCM.

	FCM	PCM	FPCM
MB	10,02	09,23	04,65
MG	13,20	12,56	09,32
LCR	22,12	21,43	15,04

TAB. 4.5 – Comparaison des taux d'erreur obtenus par différents algorithmes.

Les taux d'erreurs de classification obtenus par la méthode de classification coopérative (FPCM) sont inférieurs à ceux calculés avec les résultats de la méthode FCM, PCM pour les trois classes de tissus. D'après le tableau (4.5) Le FPCM donne des résultats satisfaisants en terme de classification des trois classes de tissus malgré que l'image IRM est à 9% de bruit .

Conclusions et perspectives

Dans ce mémoire, nous avons abordé le problème de segmentations d'images.

Dans un premier temps, nous effectuons une synthèse des différentes méthodes de segmentation qui aboutissent à une segmentation de l'image en régions homogènes telle que une région est composée de l'ensemble des pixels connexes possédant les mêmes propriétés au sens d'un prédicat d'homogénéité donné. Certaines de ces méthodes nécessitent le réglage de nombreux seuils et paramètres, tels que des critères de similarité ou des critères d'arrêt. L'ajustement de ces paramètres est délicat, de telle sorte qu'il est difficile de prévoir si nous aboutirons à une sous-segmentation ou à une sur-segmentation de l'image.

De plus, ces méthodes déterminent des régions dont les contours ne respectent pas toujours la forme des objets présents dans l'image, comme c'est le cas pour les effets de blocs. Par ailleurs, elles peuvent être sensibles aux changements d'échelle ou même à de simples translations des pixels. Enfin, il est important de noter que l'ordre d'agrégation des pixels ou l'ordre de fusion des régions a une influence sur les résultats de la segmentation.

Les limites observées sur les méthodes précédentes, nous conduit à privilégier une approche globale de l'image en appliquant des méthodes de classification de données multidimensionnelles. Ces méthodes considèrent une région comme un ensemble de pixels connexes appartenant à une même classe. Elles supposent donc que les pixels qui appartiennent à une même région possèdent des niveaux de gris similaires et forment un nuage de points bien identifiable. La classification non supervisée consiste à retrouver ces nuages de points qui correspondent aux classes de pixels présentes dans l'image. Nous devons tout d'abord préciser si l'algorithme doit être supervisé ou non. L'emploi d'un algorithme supervisé nécessite, une base d'apprentissage pour chaque classe. Cela constitue un inconvénient de ce type de méthodes. De plus, la littérature rapporte que les méthodes supervisées en segmentation d'images peuvent être très dépendantes de la base d'apprentissage. Pour toutes ces raisons, nous avons opté pour des méthodes de classification non supervisée lorsqu'aucune connaissance a priori n'est disponible.

Les résultats obtenus sur des images de synthèse par les méthodes de segmentation nette d'images par classification de pixels tel que K-Means et Mean Shift, montrent les limites de ces deux classifieurs car ils considèrent que les régions constituent des sous-ensembles nets de l'ensemble des pixels constituant l'image. La reconstruction des régions s'effectuent en affectant chaque pixel à une et seule région. Cette démarche présente des limites lorsque nous ne disposons pas de connaissance a priori sur l'affectation des pixels. En plus, en terme de performance, l'algorithme des K-Means ne garantit pas un optimum global. La qualité de la solution dépend grandement des ensembles initiaux et peut, en pratique, être bien en deçà de l'optimum global.

Parmi les méthodes qui supposent que les pixels appartiennent plus ou moins à des sous-ensembles nets et considèrent que les pixels peuvent ainsi regroupés en sous-ensembles flous de pixels, nous retrouvons les méthodes de classification dite floues. Celle-ci s'appuie sur les concepts de la logique floue en considérant les classes de pixels comme des sous-ensembles flous. Nous constatons d'une part que ces méthodes de segmentation ne prennent pas en compte la disposition spatiale des pixels et ne considèrent que le vecteur de caractéristiques utilisé. Ainsi, lors de la classification, le contexte spatial est perdu, et les nuages de points dans l'espace des niveaux de gris peuvent ne pas correspondre à des régions significatives dans l'image analysée comme le montrent les résultats obtenus sur les deux images synthétiques. Nous constatons d'autre part que ces méthodes ne tiennent généralement pas compte de la non équiprobabilité des classes, alors que les régions présentes dans une image sont généralement de tailles différentes. Enfin notons aussi le problème de la sensibilité à l'initialisation (différentes initialisations peuvent aboutir à différentes partitions).

Les résultats obtenus sur des images IRM en vue d'une segmentation des tissus du cerveau en 3 classes à savoir (MB, MG, LCR), montrent que l'algorithme des Fuzzy C-Means et les C-Moyennes Possibilistes mais aussi les C-Moyennes Possibilistes Flous restent les algorithmes de référence des méthodes de classification floue de pixels. Ces algorithmes sont bien adaptés à l'identification de classes et ils sont d'autant plus performants que nous connaissons le nombre de classes a priori. En plus de la simplicité algorithmique de ces deux algorithmes et leur rapidité d'exécution.

Bien que le problème de la segmentation d'images reste un problème qui est encore loin d'être résolu et la segmentation, telle qu'elle est définie n'est évidemment pas unique. Le choix d'une méthode est lié à plusieurs facteurs et l'ajustement des paramètres qui gouverne le fonctionnement de l'algorithme mérite une attention particulière.

Enfin, il nous paraît intéressant d'envisager l'intégration de la disposition spatiale des pixels ainsi que l'utilisation combinée de plusieurs algorithmes de classification travaillant en coopération permettant ainsi de pallier aux problèmes rencontrés par l'utilisation d'un seul algorithme. Des stratégies de coopération sont à envisager afin d'aboutir à une convergence plus rapide des algorithmes précédents pour une segmentation beaucoup plus performante.

Annexe 1

– Algorithme de la LPE :

Il y a plusieurs manières d'implémenter le principe de la LPE . Certains algorithmes ne simulent pas le principe de l'immersion mais trouvent directement les lignes de partages des eaux. Nous avons donc choisi d'implémenter un algorithme d'immersion qui est celui de Vincent-Soille :

Algorithm 6 Algorithme "Watershed" de Vincent-Soille

```
1 : procedure Watershed-by-Immersion
2 : Input : digital grey scale image  $G = (D; E; im)$ .
3 : Output : labelled watershed image  $lab$  on  $D$ .
4 : #dene init -1 ( initial value of lab image )
5 : #dene mask -2 ( initial value at each level )
6 : #dene wshed 0 ( label of the watershed pixels )
7 : #dene fictitious (-1;-1) ( ctitious pixel 62 D)
8 :  $curlab \leftarrow 0$  ( curlab is the current label )
9 :  $fifo\_nit(queue)$ 
10 : for all  $p \in D$  do
11 :  $lab[p] \leftarrow init$ ;  $dist[p] \leftarrow 0$  ( dist is a work image of distances )
12 : end for
13 : SORT pixels in increasing order of grey values (minimum  $h_{min}$ , maximum  $h_{max}$ )
14 :
15 : ( Start Flooding )
16 : for  $h_{min}$  to  $h_{max}$  do
17 : for all  $p \in D$  with  $im[p] = h$  do ( mask all pixels at level h )
18 : ( these are directly accessible because of the sorting step )
19 :  $lab[p] \leftarrow mask$ 
20 : if  $p$  has a neighbour  $q$  with ( $lab[q] > 0$  or  $lab[q] = wshed$ ) then
21 : ( Initialize queue with neighbours at level h of current basins or watersheds )
22 :    $dist[p] \leftarrow 1$ ;  $fifo\_add(p; queue)$ 
23 :   end if
24 : end for
25 :  $curdist \leftarrow 1$ ;  $fifo\_add(fictitious; queue)$ 
```

Algorithm 7 Algorithmme "Watershed"de Vincent-Soille (Suite).

```

26 : loop ( extend basins )
27 :  $p \leftarrow \text{fifo\_remove}(\text{queue})$ 
28 : if  $p = \text{fictitious}$  then
29 : if  $\text{fifo\_empty}(\text{queue})$  then
30 : break
31 : else
32 :  $\text{fifo\_add}(\text{fictitious}; \text{queue}); \text{curdist} \leftarrow \text{curdist} + 1;$ 
33 :  $p \leftarrow \text{fifo\_remove}(\text{queue})$ 
34 : end if
35 : end if
36 : for all  $q \in N_G(p)$  do ( labelling p by inspecting neighbours )
37 : if  $\text{dist}[q] < \text{curdist}$  and  $(\text{lab}[q] > 0 \text{ or } \text{lab}[q] = \text{wshed})$  then
38 : ( q belongs to an existing basin or to watersheds )
39 : if  $\text{lab}[q] > 0$  then
40 : if  $\text{lab}[p] = \text{mask}$  or  $\text{lab}[p] = \text{wshed}$  then
41 :  $\text{lab}[p] \leftarrow \text{lab}[q]$ 
42 : else if  $\text{lab}[p] \neq \text{lab}[q]$  then
43 :  $\text{lab}[p] \leftarrow \text{wshed}$ 
44 : end if
45 : else if  $\text{lab}[p] = \text{mask}$  then
46 :  $\text{lab}[p] \leftarrow \text{wshed}$ 
47 : end if
48 : else if  $\text{lab}[q] = \text{mask}$  and  $\text{dist}[q] = 0$  then ( q is plateau pixel )
49 :  $\text{dist}[q] \leftarrow \text{curdist} + 1; \text{fifo\_add}(q; \text{queue})$ 
50 : end if
51 : end for
52 : end loop
53 : ( detect and process new minima at level h )
54 : for all  $p \in D$  with  $\text{im}[p] = h$  do
55 :  $\text{dist}[p] \leftarrow 0$  ( reset distance to zero )
56 : if  $\text{lab}[p] = \text{mask}$  then ( p is inside a new minimum )
57 :  $\text{curlab} \leftarrow \text{curlab} + 1;$  ( create new label )
58 :  $\text{fifo\_add}(p; \text{queue}); \text{lab}[p] \leftarrow \text{curlab}$ 
59 : while not  $\text{fifo\_empty}(\text{queue})$  do
60 :  $q \leftarrow \text{fifo\_remove}(\text{queue})$ 
61 : for all  $r \in N_G(q)$  do ( inspect neighbours of q )
62 : if  $\text{lab}[r] = \text{mask}$  then
63 :  $\text{fifo\_add}(r; \text{queue}); \text{lab}[r] \leftarrow \text{curlab}$ 
64 : end if
65 : end for
66 : end while
67 : end if
68 : end for
69 : end for
70 : ( End Flooding )

```

Toutefois, cet algorithme de segmentation d'images, qui sont à niveaux de gris dans notre cas, nécessite la mise en évidence et le repérage des contours des objets. Cette détermination des minima et des maxima utilise les variations du contraste de l'image, quantifiées par le calcul du gradient. De nombreuses méthodes de calcul du gradient existent. Nous pouvons notamment, par les outils morphologiques, calculer un gradient de l'image, c'est le gradient morphologique. En fait, le gradient morphologique pourrait être défini comme étant la différence entre le dilaté de l'image originale I par l'élément structurant symétrique B et l'érodé de I par B .

Annexe 2

Critère de fusion : critère flou de compacité et chevauchement des classes.

Deux régions doivent être fusionnées si elles sont proches et si leur réunion constitue une région homogène. Pour ce faire, deux quantités appelées : chevauchement et compacité sont définies.

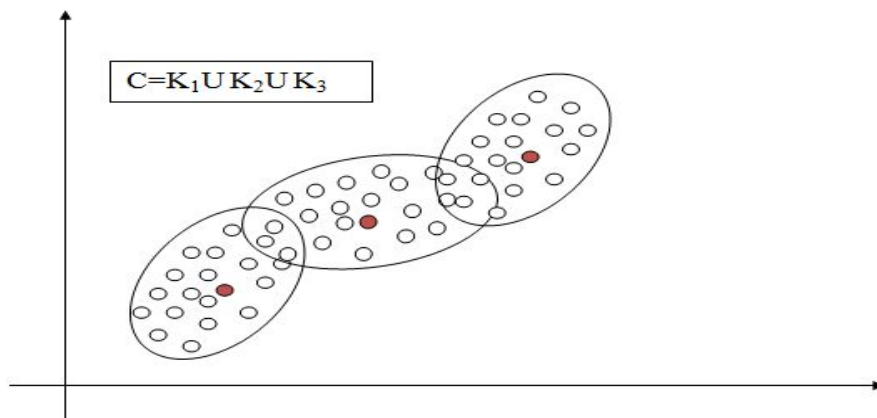


FIG. 4.14 – Fusion de classes.

– **Cardinal, réunion et intersection d'ensembles flous :**

Soit A et B deux ensembles flous de fonctions d'appartenance f_A et f_B respectivement. Les notions de cardinal, réunion et intersection, peuvent être étendues aux ensembles flous en utilisant les fonctions d'appartenance par :

1. **Cardinal :**

$$Car(A) = \sum_x f_A(x)$$

2. **Intersection :**

$$f_{(A \cap B)}(x) = \text{Min}(f_A(x), f_B(x))$$

3. **Réunion :**

$$f_{(A \cup B)}(x) = \text{Max}(f_A(x), f_B(x))$$

– **Chevauchement de deux classes :**

$$O_{ij}(C_1, C_2) = \frac{\text{Card}(C_1 \cap C_2)}{\text{Card}(C_1 \cup C_2)}$$

$$0 \leq O_{ij} \leq 1$$

Si $O_{ij} = 0$ classes disjointes. Si $O_{ij} = 1$ classes confondues.

– **Compacité d'une classe :** $C = K_1 \cup K_2 \cup \dots \cup K_p$

La compacité d'une classe est définie comme moyenne des chevauchement entre ses différentes composantes élémentaires.

$$CP(C) = \frac{1}{p} \sum_i \text{Min}_{j \neq i} (O_{ij})$$

$$0 \leq CP \leq 1$$

Si $CP = 0$ classe non compacte. Si $CP = 1$ classe compacte.

– **La compacité moyenne entre deux clusters :**

La compacité moyenne de deux classes C_1 et C_2 est la moyenne des compacités individuelles, pondérées par les cardinaux.

$$CP_{12} = \frac{(\text{Card}(C_1).CP_1 + \text{Card}(C_2).CP_2)}{\text{Card}(C_1) + \text{Card}(C_2)}$$

– **Règle de Fusion :** Soit deux classes C_i et C_j , $\alpha \in [0, 1]$ un seuil à choisir.

Si : $O_{ij} > \alpha.CP_{ij}$ Alors : Fusionner C_i et C_j .

C'est à dire :

" Deux classes sont fusionnées si leur chevauchement est assez élevé par rapport à un seuil dépendant naturellement des classes en question. Moins les classes sont compactes, moins nous sommes exigeant quand au seuil de fusion et on a tendance donc à fusionner et vice versa".

Annexe 3

L'Imagerie par Résonance Magnétique

Principes physiques

La technique de l'IRM exploite les propriétés magnétiques des éléments constitutifs de la matière et se fonde sur les phénomènes physiques de résonance et de relaxation.

Moment magnétique des noyaux [40]

L'existence du moment magnétique des noyaux a été montrée simultanément par Bloch et Purcell. Le corps humain étant constitué en moyenne de 70% d'eau, on s'intéresse en pratique à la molécule d'eau et en particulier au noyau d'hydrogène (proton).

Le noyau d'hydrogène se comporte comme une charge en rotation autour de son axe : c'est le mouvement de spin (figure 4.15). Les protons peuvent alors être assimilés à des dipôles magnétiques. En l'absence de tout champ magnétique, ceux-ci vont s'orienter dans l'espace de façon aléatoire. Ce mouvement confère au noyau un moment cinétique qui dépend de sa masse et un moment magnétique qui dépend de sa charge.

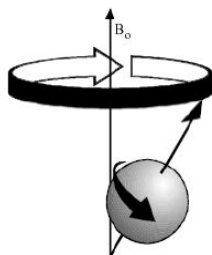


FIG. 4.15 – Mouvement de spin

Dans un champ magnétique B_0 , les protons s'orientent alors par rapport à B_0 et décrivent autour de ce champ un mouvement de précession, de vitesse angulaire constante. L'ensemble des protons s'orientant dans le sens de B_0 forme alors une sous-population de spins $+\frac{1}{2}$; l'ensemble s'orientant dans le sens inverse forme la sous-population de spins $-\frac{1}{2}$. La sous-population de spins

$+\frac{1}{2}$ étant la plus importante, il existe une aimantation résultante M_0 proportionnelle au nombre de protons présents. M_0 ne possède qu'une composante longitudinale, dans le sens de B_0 (figure 4.16).

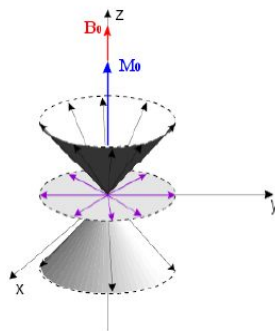


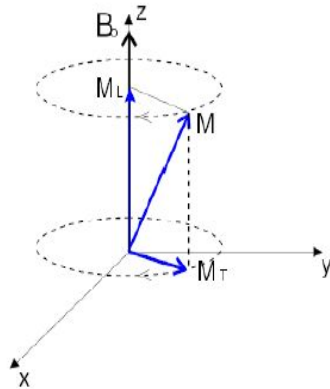
FIG. 4.16 – Mouvement de précession d'un ensemble de moments magnétiques

Phénomènes de résonance et de relaxation

Les phénomènes de résonance et de relaxation sont liés au principe énergétique de la matière. Lorsque des protons, placés dans un champ magnétique, reçoivent un apport d'énergie sous la forme d'ondes radio fréquences de pulsation égale à leur fréquence de résonance et émises par un champ magnétique B_1 , ils passent d'un niveau bas d'énergie à un niveau haut d'énergie; cette transition correspond au phénomène de résonance magnétique. L'orientation du champ magnétique résultant change et passe à un nouvel état d'équilibre M tant que B_1 dure. On décompose alors M en un moment magnétique longitudinal M_L et en un moment magnétique transversal M_T . Le retour à l'équilibre des protons, la relaxation, lors de la disparition du champ magnétique B_1 , s'accompagne d'un mouvement en spirale des protons autour du champ magnétique B_0 . Bloch a montré que l'évolution de ce mouvement est liée au temps de relaxation longitudinal T_1 et au temps de relaxation transversal T_2 (figure 4.17). Les valeurs de ces derniers dépendent des tissus biologiques rencontrés (tableau 4.6).

	T_1	T_2
Liquide Céphalo-Rachidien (LCR)	2500 ms	2000 ms
Matière Grise (MG)	900 ms	90 ms
Matière Blanche (MB)	750 ms	80 ms
Graisse (G)	300 ms	40 ms

TAB. 4.6 – Ordre de grandeur des temps de relaxation à 1,5 Tesla.

FIG. 4.17 – Décomposition du moment magnétique M .

Formation des images IRM

Pour constituer une image IRM, on soumet le patient à un champ magnétique B_0 et à une onde radio fréquence B_1 de fréquence égale à la fréquence de précession des protons. Sous l'effet de cette onde, les protons basculent d'un angle, généralement de 90 ou de 180 par rapport à M_0 . À l'arrêt de l'impulsion, le retour à l'équilibre s'établit : il y a réduction rapide du moment transversal M_T et repousse du moment longitudinal M_L . Pendant ce retour à l'équilibre, le proton continue à précesser et induit un champ magnétique FID (Free Induction Decay). C'est ce signal qui est recueilli par les bobines de réception puis transformé en signal électrique (figure 4.18).

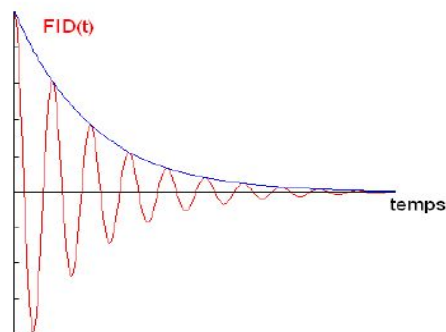


FIG. 4.18 – Signal FID recueilli par les bobines réceptrices.

L'image numérique IRM représente ainsi la répartition spatiale d'une grandeur physique : chaque élément d'une coupe correspond à la représentation de l'aimantation nucléaire d'un volume de l'espace exploré. Ces éléments sont habituellement appelés pixels quand on considère

individuellement une coupe et voxels quand on considère le volume constitué de l'empilement des coupes.

La problématique liée au codage de l'image est alors de savoir différencier les signaux issus des différentes zones examinées. Ce codage est réalisé grâce au couplage de trois gradients différents :

- Le gradient de coupe sélective sélectionne le plan de coupe, celui-ci pouvant être quelconque dans l'espace.
- Le gradient de codage de phase sélectionne les lignes dans le plan de coupe sélectionné.
- Le gradient de fréquence sélectionne les colonnes dans le plan de coupe sélectionné.

L'orientation des coupes, qui dépend en particulier du gradient de coupe sélective, varie en fonction de la pathologie recherchée. On retrouve cependant trois orientations principales : axiale, coronale et sagittale (figure 4.19).

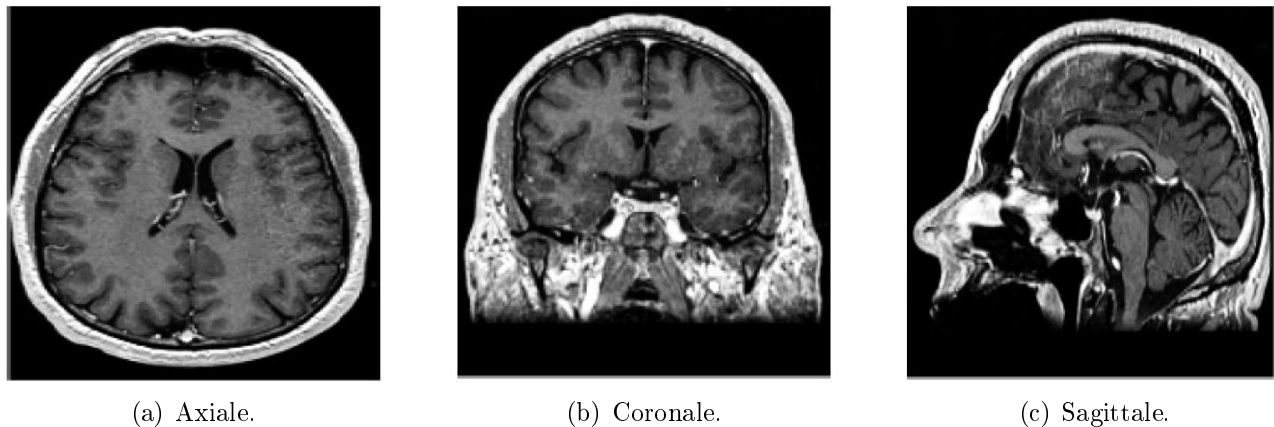


FIG. 4.19 – Plans sur une acquisition en T_1 .

Différentes pondérations

Nous pouvons identifier les paramètres qui influencent le contraste de l'image IRM en deux grandes classes [40] :

- La première est constituée de paramètres intrinsèques liés directement aux tissus observés. Il s'agit de la densité en protons, des temps de relaxation T_1 et T_2 , de la présence d'un agent de contraste ou encore de la vitesse des fluides circulant.
- La seconde est constituée de paramètres liés à l'appareil lui-même (en particulier l'intensité et la constance du champ magnétique B_0) et à la séquence d'acquisition. Cette dernière dépend essentiellement du temps de répétition T_R séparant deux impulsions de l'onde radio fréquence B_1 et du temps d'écho T_E séparant l'impulsion de la lecture du signal.

Parmi ces paramètres, seuls les paramètres d'acquisition sont modulables et conduisent à l'obtention d'images dites « pondérées en T_1 », « pondérées en T_2 » ou encore « pondérées en ρ » (figure 4.20).

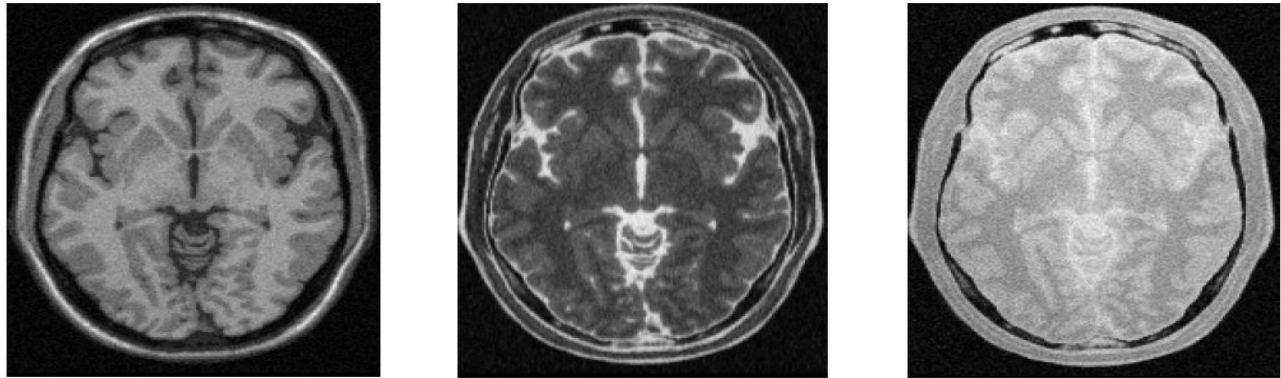
(a) Pondération en T_1 .(b) Pondération en T_2 .(c) Pondération en Rho .

FIG. 4.20 – Une même coupe obtenue sous différents paramètres d'acquisition.

Image pondérée en T_1

Pour des T_R de l'ordre de 600 ms, le contraste entre les tissus dépend essentiellement de leur vitesse d'aimantation, donc de T_1 . Pour des T_E d'environ 20 ms, les différences de décroissance du signal entre les tissus n'ont pas le temps de s'exprimer, rendant le contraste indépendant de T_2 . Ainsi, on obtient une image pondérée en T_1 , où les tissus sont ordonnés par niveaux de gris croissants en liquide céphalo-rachidien, matière grise puis matière blanche.

Image pondérée en T_2

Pour des T_R de l'ordre de 2 s et des T_E d'environ 90 ms, la décroissance du signal domine la différence de densité protonique entre tissus. Le signal est alors suffisant pour réaliser une image dite pondérée en T_2 , où les tissus sont ordonnés par niveaux de gris croissants en matière blanche, matière grise puis liquide céphalo-rachidien.

Image pondérée en densité de protons ρ

Pour un T_R de l'ordre de 2 s et un T_E court d'environ 20 ms, la différence de densité protonique entre la matière grise et la matière blanche s'exprime. On obtient une séquence qui reflète la localisation et la concentration des noyaux d'hydrogène des différentes structures. Les tissus sont ordonnés par niveaux de gris croissants en matière blanche, matière grise puis liquide céphalo-rachidien.

Bibliographie

- [1] S.Chabrier, "Contribution à l'évaluation de performances en segmentation d'images", Thèse de Doctorat, Université d'Orléans,2005.
- [2] J. Lecoeur, C. Barillot,"Segmentation d'images cérébrales : Etat de l'art", Rapport de recherche n°6306 , Juillet 2007.
- [3] L. Lifshitz, S. Pizer,"A multiresolution hierarchical approach to image segmentation based on intensity extrema", IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(6),pp.529–540, 1990.
- [4] J. Hadamard, "Le Problème de Cauchy et les Equations aux Dérivées Partielles Linaires Hyperboliques", Herman et Cie, 1932.
- [5] S.Zucker,"Region growing : childhood and adolescence, Computer Graphics and Image Processing", pp.382–399, 1976.
- [6] H. D. Cheng, X. H. Jiang, Y. Sun et J. Wang,"Color image segmentation : advances and projects", Pattern Recognition, 34, pp.2259–2281, 2001.
- [7] A. Trémeau et N. Borel,"A region growing and merging algorithm to color segmentation", Pattern Recognition, 30 , pp.1191–1203, 1997.
- [8] L. Shaffarenko et M. Petrou,"Automatic watershed segmentation of randomly textured color images". IEEE Transactions On Image Processing, 6(11), pp.1530–1543, November 1997.
- [9] L.Vincent, P.Soille, "Watershed in digital spaces, an efficient algorithm based on immersion simulation", IEEE Transactions on Pattern Analysis and Machine Intelligence, June 91.
- [10] F.Meyer, "Skeletons in digital spaces. Image analysis and mathematical morphology, theoretical advances", Serra, Academic press, 1988.
- [11] M.Perona, J.Malik, "Scale-space and edge detection using anisotropic diffusion", IEEE Transactions on Pattern Analysis and Machine Intelligence, 12, pp.629-639, July 1990.
- [12] A.Gagalowicz, O.Monga, "A New approach for image segmentation", 8 th IEEE Int Pattern Recognition, pp.265-267, Paris France , December 1986.
- [13] S. L. Horowitz, S. Pavlidis, "Picture segmentation by a directed split and merge procedure". 2nd Int Joint Conf on Pattern Recognition, 1974.
- [14] K. C. Strasters, J. J. Gerbrands, "Three-dimensional image segmentation using a split, merge and group approach", Pattern Recognition Letters, 1991.
- [15] J. Landré,"Analyse multirésolution pour la recherche et l'indexation d'images par le contenu dans les bases de données images -Application à la base d'images paléontologique trans'nyfipal", Décembre 2005.

-
- [16] J.C. Atine, "Méthodes d'apprentissage flou : Application à la segmentation d'images biologiques", Rapport n° 05600 LAAS, 2005.
- [17] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability", Berkeley, University of California Press, 1971.
- [18] E. Forgy, "Cluster analysis of multivariate data : Efficiency vs. interpretability of classifications", *Biometrics*, pp.768, 1965.
- [19] R. Zhang, I. Alexander Rudnicky, "A large scale clustering scheme for kernel k-means", *ICPR* (4), pp.289-292, 2002.
- [20] D. Pelleg, A. Moore, "Means : Extending -means with efficient estimation of the number of clusters", *Proc. 17th International Conf on Machine Learning*, pp.727-734, San Francisco, 2000.
- [21] V. S. Moertini, "Introduction to five data clustering algorithms", University of Parahyangan, Bandung, Août 2002.
- [22] R. Duda, P. Hart and D. Stork, "Pattern Classification", Wiley-Interscience, deuxième édition, 2001.
- [23] B. Silverman, "Density Estimation for Statistics and Data Analysis", Chapman & Hall, 1986.
- [24] D. Comaniciu, P. Meer, "Mean Shift : A robust approach toward feature space analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.
- [25] D. Comaniciu, P. Meer, "Mean Shift analysis and application", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and *Proc of ICCV 99*, Kerkyra, Greece, 1999.
- [26] L.A Zadeh, "Fuzzy sets, Information and Control", 8, 1965.
- [27] B. Meunier, "La logique floue et ses applications", Addison Wesley, 1995.
- [28] W.T. Tucker, J.C. Bezdek, "Counterexample to the convergence theorem for fuzzy ISODATA clustering algorithms", *Analysis of Fuzzy Information Conference*, 3, Edition Boca Raton, 1987.
- [29] G. Looney, "A fuzzy clustering and fuzzy merging algorithm Carl", Technical report, CS-UNR-101, 1999.
- [30] J.P. Stitt, R.L. Tutwiler and A.S. Lewis, "Fuzzy C-Means image segmentation of side scan sonar images", *Proceeding of 2001 IASTED, International Symposium on Signal and Image Processing*, vol. 1, pp. 27-32, 2001.
- [31] J.C. Bezdek, J. Hathaway, M.J. Sabin, and W. T. Tucker, "Convergence Theory for Fuzzy C-Means : Counterexamples and Repairs", *IEEE Transactions*, 17, pp. 873-877, September/October 1987.
- [32] R.O. Duda, P.E Hart, "Pattern classification and scene analysis", John Willey and Sons, New York, 1973.
- [33] P. Lambert, H. Greu, "A quick and coarse color image segmentation", *IEEE*, *ICIP* 2003.
- [34] J. C. Bezdek, "A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2, pp.1-8, January 1980.

-
- [35] R. Krishnapuram and J.M Keller, "A possibilistic approach to clustering". IEEE Transactions on Fuzzy Systems, 1(2), pp.98-110, 1993.
 - [36] M.J. Lesot, B.Bouchon-Meunier,"Extraction de concepts descriptifs avec exceptions par classification non supervisée hybride.", Laboratoire d'Informatique de Paris VI, Université Pierre et Marie Curie.
 - [37] R .Krishna, "The Possibilistic C-Means Algorithm : Insights and Recommendations", IEEE transactions on Fuzzy Systems, 1996.
 - [38] S. Moussa , T.Lyazid et M.Abdelouaheb,"Système Coopératif de Classification Floue Possibiliste avec Rejet d'Ambiguïté : Application à la segmentation d'images IRM ", 4th International Conference on Computer Integrated Manufacturing CIP'2007.
 - [39] Nikhil R. Pal, Kuhu Pal, James M. Keller, and James C. Bezdek, "A possibilistic fuzzy c-means clustering algorithm", IEEE Transactions on Fuzzy Systems, 13(4),pp.517–530,1997.
 - [40] S. Miri,"Segmentation des structures cérébrales en IRM : intégration de contraintes topologiques", Rapport de stage, Université Louis Pasteur Strasbourg, 2007.

Résumé :

La segmentation d'image est une opération de traitement d'images de bas niveau qui consiste à localiser dans une image les régions (ensembles de pixels) appartenant à une même structure. Cette opération est à la base de nombreuses applications tant en vision industrielle, qu'en imagerie médicale. L'approche par classification fournit une partition de l'image en regroupant des pixels ayant des niveaux de gris similaires dans une même classe de pixels connaît un très grand succès. L'objectif de ce mémoire est d'étudier différentes méthodes de classification non supervisées et de les confronter afin de les appliquer sur des images cérébrales IRM en vue d'une segmentation et d'une classification des différentes régions correspondant aux différents tissus et structures. Nous nous sommes intéressés à une famille d'algorithmes : FCM qui modélise l'incertitude et l'imprécision qui a été beaucoup utilisé pour la segmentation des images cérébrales, quelle que soit la modalité et le type d'acquisition, PCM qui génère un degré d'appartenance absolu reflétant de manière exacte la réalité de distribution des pixels et le PFCM dont le but étant de tirer partie des avantages de chacun.

Abstract :

Image segmentation is a low-level processing operation. This operation is the basis for many applications in both industrial vision, medical imaging that. The approach by classification provided a partition of the image gathering pixels with similar levels of gray in the same class of pixels has a great success. The purpose of this memorandum is to explore different unsupervised classification methods and confront them in order to implement them on brain MRI images for segmentation and classification of different regions corresponding to the various tissues and structures. We are interested in a family of algorithms: FCM that models the uncertainty and inaccuracy which has been widely used for segmentation of brain images, whatever the modality and the type of acquisition, PCM, which generates a degree belonging absolute accurately reflecting the reality of distribution of pixels and PFCM whose goal is to capitalize on the advantages of each.

الخلاصة :

التقسيم أو التجزيء ويهتم بتقسيم الصورة إلى مكوناتها الجزئية أو إلى عناصر هي معالجة الصور على مستوى منخفض ويعتبر من أهم العمليات على الصورة. هذه العملية هي الأساس لكثير من التطبيقات الصناعية في كل من الرؤية، التصوير الطبي... إلخ. والغرض من هذه المذكرة هو استكشاف الطرق المختلفة للتصنيف (classification) غير الخاضعة لإشراف ومواجهتها من أجل تنفيذها على صور المخ (MRI) وتصنيف مختلف المناطق المقابلة لمختلف الهياكل و الأنسجة. ونحن مهتمون في أسرة مكونة من الخوارزميات : FCM نماذج من عدم الثقة (منطوق ضبابي) التي كانت تستخدم على نطاق واسع لتقسيم الصور في المخ ، PCM ، الذي يولد درجة الانتماء المطلق بدقة تعكس واقع توزيع نقاط الشاشة. FPCM والتي تهدف إلى الاستفادة من مزايا كل منها