

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

Université Ferhat Abbas – Sétif 1-



THÈSE

Présentée à la Faculté des Sciences

Département d'Informatique

Pour l'Obtention du Diplôme de

Doctorat en Sciences

Option : Informatique

par

Moussa SEMCHEDINE

Thème

Contribution à la segmentation d'images médicales par les algorithmes bio-inspirés

Soutenue le : 2 Juillet 2018

devant la commission d'examen :

Dr. SAIDI MOHAMED

Pr. MOUSSAOUI ABDELOUAHAB

Pr. FOUJIL CHERIF

Pr. BABAHENINI MOHAMED CHAOUKI

Dr. LAMICHE CHAABANE

MC-A à l'UFA - Sétif-1

Prof. à l'UFA - Sétif-1

Prof. à l'Univ -Biskra-

Prof. à l'Univ -Biskra-

MC-A à l'Univ -Msila-

Président

Rapporteur

Examineur

Examineur

Examineur

Remerciements

Je tiens à remercier en premier lieu mon directeur de thèse, Professeur Abdelouahab Moussaoui, pour son encadrement, ses conseils et qui a toujours suivi ce travail avec un grand intérêt.

Mes remerciements vont également à Messieurs le président et les membres du jury, d'avoir bien voulu nous faire l'honneur de juger ce travail.

Je souhaiterais exprimer ma gratitude à Docteur Mahieddine Djoudi de l'Université de Poitiers, pour son accueil chaleureux au sein du département d'informatique.

Un grand merci à tous mes enseignants pour le savoir, la connaissance et surtout la générosité durant mes études primaires, secondaires et universitaires.

Je souhaiterais aussi adresser ma gratitude à toute personne avec qui j'ai eu la chance d'enrichir mes connaissances scientifiques.

A ma grand-mère Massida

A mes chers parents

A mes frères et ma sœur

A mon épouse et mes enfants

Table des matières

Introduction générale	1
1 Imagerie Cérébrale par Résonance Magnétique	5
1.1 Introduction	5
1.2 Eléments d'anatomie cérébrale	6
1.2.1 L'encéphale	6
1.2.2 Les principaux tissus cérébraux	7
1.3 Observation du cerveau	8
1.4 Imagerie médicale	9
1.5 Imagerie par Résonance Magnétique (IRM)	9
1.5.1 Principe général	9
1.5.2 Principe physique	10
1.5.3 Paramètres de séquence	10
1.5.4 Les séquences IRM	11
1.5.5 Artefacts des images IRM	12
1.6 Conclusion	13
2 Etat de l'art : Segmentation d'images IRM	15
2.1 Introduction	15
2.2 Définition	16
2.2.1 Définition générale	16
2.2.2 Définition formelle	17
2.3 Objectifs et applications de la segmentation	17
2.4 Spécificités de la segmentation IRM	18
2.5 Méthodes de segmentation	18
2.5.1 Approches contours	19
2.5.2 Approches régions	21
2.5.3 Segmentation par classification	24
2.6 Conclusion	31

3	Algorithmes bio-Inspirés	33
3.1	Introduction	33
3.2	Définition	34
3.3	Le problème d'optimisation	35
3.4	Les méthodes d'optimisation	35
3.4.1	Les méthodes déterministes (exactes)	35
3.4.2	Les méthodes non-déterministes (métaheuristiques)	36
3.5	Les algorithmes bio-inspirés	37
3.5.1	Les automates cellulaires (CA)	37
3.5.2	Recuit simulé (SA)	38
3.5.3	Les algorithmes génétiques (GA)	39
3.5.4	Algorithme de colonies de fourmis (ACO)	41
3.5.5	Algorithme à évolution différentielle (DE)	43
3.5.6	Optimisation par essaim de particules (PSO)	43
3.5.7	Systèmes immunitaires artificiels (AIS)	44
3.5.8	Algorithme de recherche d'harmonie (HS)	45
3.5.9	Algorithme d'essaims de poissons artificiels (AFSA)	45
3.5.10	Optimisation par recherche bactérienne (BFO)	46
3.5.11	Algorithme de colonie d'abeilles (ABC)	46
3.5.12	Algorithme de Recherche gravitationnelle (GSA)	47
3.5.13	Algorithme Firefly (FA)	48
3.5.14	Algorithme de Chauve-souris (BA)	49
3.5.15	Autres algorithmes bio-inspirés	49
3.6	Conclusion	50
4	Algorithme d'optimisation par essaim de particules	51
4.1	Introduction	51
4.2	Inspiration	52
4.2.1	Modèle de Reynolds	52
4.2.2	Métaphore	52
4.2.3	L'intelligence collective	53
4.3	Principe général	53
4.4	Les topologies de voisinage	54
4.4.1	Voisinage géographique	54
4.4.2	Voisinage social	55
4.5	Algorithme PSO	56
4.5.1	Formalisation	56
4.5.2	Paramètres de l'algorithme	57

4.6	Avantages et inconvénients de PSO	59
4.7	Conclusion	60
5	FCM rapide pour la segmentation d’images IRM	61
5.1	Introduction	61
5.2	Hybridation d’algorithmes	62
5.2.1	Définition	62
5.2.2	Objectifs de l’hybridation	62
5.2.3	Types d’hybridation	62
5.3	Contribution	64
5.3.1	Motivation	64
5.3.2	Description	64
5.4	Validation et résultats	67
5.5	Discussion	69
5.6	Conclusion	73
6	Contribution à la segmentation d’images par hybridation PSO-FCM	75
6.1	Introduction	75
6.2	Hybridation algorithmes bio-inspirés–FCM	76
6.3	Contribution	80
6.3.1	Motivation	80
6.3.2	Description	81
6.3.3	Algorithme FPSOFCM	84
6.3.4	Choix des paramètres de FPSOFCM	87
6.4	Validation et Résultats	88
6.4.1	Introduction	88
6.4.2	Base d’images	90
6.4.3	Evaluation et étude comparative	90
6.5	Conclusion	121
	Conclusion générale et perspectives	123
	Annexe A Résultats sur des coupes coronales	125
	Annexe B Résultats sur images bruitées	135
	Annexe C Description des images de test	139

Table des figures

1.1	Structures anatomiques de l'encéphale	7
1.2	Les principaux tissus cérébraux	8
1.3	Les trois vues des IRM cérébrales	8
1.4	Schéma fonctionnel de l'IRM	11
1.5	Les séquences IRM	12
1.6	L'effet du volume partiel	13
1.7	L'inhomogénéité RF	13
2.1	Exemple d'une segmentation	16
2.2	Les modèles de contours	19
2.3	Segmentation par approches dérivatives	20
2.4	Principe du contour actif	20
2.5	Segmentation par seuillage simple	22
3.1	Changement de la fonction objectif	34
3.2	Composition d'une population de l'algorithme génétique	39
3.3	Détermination du plus court chemin	42
4.1	Les trois règles de Reynolds	52
4.2	Principe de déplacement d'une particule	54
4.3	Topologies statiques	56
5.1	Hybridation par initialisation	63
5.2	Hybridation par fusion	63
5.3	Hybridation par rétroaction	64

5.4	Construction de la matrice des centres	66
5.5	Comparaison de temps d'exécution , bruit=0%, Rf=0%.	70
5.6	Comparaison de temps d'exécution , bruit=3%, Rf=20%.	71
5.7	Comparaison de temps d'exécution , bruit=5%, Rf=40%.	72
6.1	Schéma général de l'approche proposée	85
6.2	Changement de l'indice XB	89
6.3	Comparaison de l'indice PC / T1 /axiales	94
6.4	Comparaison de l'indice PC / DP /axiales	94
6.5	Comparaison de l'indice MPC / T1 /axiales	95
6.6	Comparaison de l'indice MPC / DP /axiales	95
6.7	Comparaison de l'indice PE / T1 /axiales	96
6.8	Comparaison de l'indice PE / DP /axiales	96
6.9	Comparaison de l'indice XB / T1 /axiales	97
6.10	Comparaison de l'indice XB / DP /axiales	97
6.11	Comparaison de la fonction objectif/ T1 /axiales	99
6.12	Comparaison de la fonction objectif/ DP /axiales	99
6.13	Convergence de FCM et FPSOFCM/ T1 /axiales	100
6.14	Convergence de FCM et FPSOFCM/ DP /axiales	100
6.15	Résultats de segmentation / T1 / axiales / Rf=0%	102
6.16	Résultats de segmentation / T1 / axiales / Rf=20%	103
6.17	Résultats de segmentation / T1 / axiales / Rf=40%	104
6.18	Résultats de segmentation / DP / axiales / Rf=0%	105
6.19	Résultats de segmentation / DP / axiales / Rf=20%	106
6.20	Résultats de segmentation / DP / axiales / Rf=40%	107
6.21	Résultats de segmentation / images réelles.	108
6.22	Séparation des classes segmentées image 1	109
6.23	Séparation des classes segmentées image 2	109
6.24	Séparation des classes segmentées image 3	110
6.25	Comparaison de l'image de référence à l'image segmentée	111
6.26	Taux de recouvrement MG	116

6.27 Taux de recouvrement MB	116
6.28 Erreur de classification MG	116
6.29 Erreur de classification MB	117
6.30 Taux de recouvrement MG	118
6.31 Taux de recouvrement MB	118
6.32 Indice de similarité MG	119
6.33 Indice de similarité MB	119
6.34 Erreur de classification MG	120
6.35 Erreur de classification MB	120
A.1 Comparaison de l'indice PC / T1 / coronales	128
A.2 Comparaison de l'indice PC / DP / coronales	128
A.3 Comparaison de l'indice MPC / T1 /coronales	129
A.4 Comparaison de l'indice MPC / DP /coronales	129
A.5 Comparaison de l'indice PE / T1 /coronales	130
A.6 Comparaison de l'indice PE / DP /coronales	130
A.7 Comparaison de l'indice XB / T1 /coronales	131
A.8 Comparaison de l'indice XB / DP /coronales	131
A.9 Comparaison de la fonction objectif/ T1 /coronales	132
A.10 Comparaison de la fonction objectif/ DP /coronales	132
A.11 Convergence de FCM et FPSOFCM/ T1 /coronales	133
A.12 Convergence de FCM et FPSOFCM/ DP /coronales	133

Liste des tableaux

5.1	Comparaison de temps d'exécution de FCM et MFCM / T1.	68
5.2	Comparaison de temps d'exécution de FCM et MFCM / DP.	68
5.3	Comparaison de temps d'exécution de FCM et MFCM / T2.	68
6.1	Notations.	81
6.2	Changement de l'indice XB	88
6.3	Comparaison des performances de FCM et FPSOFCM / T1 / axiales	92
6.4	Comparaison des performances de FCM et FPSOFCM / DP / axiales	93
6.5	Résultats quantitatifs de segmentation 1	113
6.6	Résultats quantitatifs de segmentation 2	114
6.7	Comparaison des performances de FPSOFCM et FPSO	115
A.1	Comparaison FCM et FPSOFCM / T1 / coronales	126
A.2	Comparaison FCM et FPSOFCM / DP / coronales	127
B.1	Comparaison FCM et FPSOFCM / T1 / salt & pepper	136
B.2	Comparaison FCM et FPSOFCM / DP / salt & pepper	137
C.1	Echantillon d'images N° 1	140
C.2	Echantillon d'images N° 2	140
C.3	Echantillon d'images N° 3	141
C.4	Echantillon d'images N° 4	141
C.5	Echantillon d'images N° 5	142
C.6	Echantillon d'images N° 6	142

Liste des Algorithmes

2.1	Algorithme K-means	26
2.2	Algorithme Fuzzy c-means	28
2.3	Algorithme Possibilistic c-means	31
3.1	Algorithme Récuit simulé	38
3.2	Algorithme Génétique	41
4.1	Algorithme PSO	57
5.1	Modified Fuzzy c-means MFCM	67
6.1	Algorithme proposé FPSOFCM	86

Introduction générale

Explorer l'intérieur du corps humain sans opérer, a toujours intrigué l'homme. Cette action est de nos jours possible grâce aux progrès scientifiques et technologiques dans le domaine médical. L'imagerie médicale a connu des avancées considérables, dont les techniques ne cessent de progresser. Au fil des années, différentes sortes d'imagerie médicale ont été développées. On distingue principalement la radiographie aux rayons X, la tomographie, l'échographie, l'imagerie par résonance magnétique (IRM), la scintigraphie, etc. Parmi l'ensemble de ces techniques, nous nous sommes intéressés dans le cadre de cette thèse à l'imagerie par résonance magnétique (IRM). Cette technique non invasive et indolore, permet l'acquisition d'images de haute résolution sur lesquelles différents contrastes sont possibles. Il s'agit de l'examen de référence pour la localisation précise des différentes structures du cerveau. Il permet en outre l'étude de l'activité cérébrale grâce à l'IRM fonctionnelle (IRMf).

L'une des opérations primordiales pour tout traitement automatique des images est évidemment la segmentation. Elle consiste à partitionner l'image en plusieurs régions, pour la rendre plus simple et plus significative à interpréter. Dans le contexte des IRM cérébrales, la segmentation permet de caractériser les différentes structures du cerveau, à savoir la matière grise, la matière blanche, le liquide céphalorachidien et éventuellement les structures pathologiques. Les principales applications de la segmentation en imagerie médicale sont la morphométrie, la cartographie fonctionnelle et la visualisation.

En médecine, la précision des données manipulées par les praticiens est très importante pour un diagnostic fiable. En termes d'analyse d'images, il est nécessaire que la segmentation soit précise. La segmentation des IRM cérébrales présente des particularités par rapport à d'autres domaines de la segmentation, comme celle des images faciales, ou d'images satellitaires. Ces particularités sont liées d'une part à l'anatomie du cerveau humain qui regroupe des structures enchevêtrées, de formes et de composition variées

et, d'autre part au processus d'acquisition IRM, qui génère des artefacts (ou défauts), rendant difficile la segmentation. Ces artefacts sont principalement dus aux bruits, à l'inhomogénéité de la radio fréquence, et au patient lui-même à cause de ses mouvements pendant l'acquisition de l'image, etc. Les images IRM sont particulièrement caractérisées par l'effet du volume partiel que l'on retrouve lorsqu'un pixel ayant un certain niveau de gris correspond en réalité à un mélange de deux ou plusieurs tissus, cet artefact existe principalement aux frontières entre la matière grise et la matière blanche.

Vu l'importance et les difficultés de la segmentation d'images, un grand nombre de méthodes ont été proposées dans la littérature. On distingue principalement, trois catégories : contour, régions et classification. Les méthodes de segmentation contours et régions, sont simples à mettre en œuvre. Toutefois, l'une comme l'autre, ne sont utilisables que dans des contextes limités et demandent l'intégration d'informations *a priori* sur les zones recherchées. Parmi les approches de segmentation par classification, on distingue les méthodes supervisées pour lesquelles les caractéristiques de chaque classe sont connues *a priori* et les méthodes non supervisées. L'inconvénient des approches supervisées est qu'elles sont trop dépendantes de l'opérateur et nécessitent de créer une base d'apprentissage pour chaque image, ce qui représente une tâche longue et pénible pour les experts. Les méthodes non supervisées appelées aussi méthodes automatiques, visent à regrouper les pixels en classes sans aucune base préalable d'apprentissage et aucune tâche d'étiquetage.

Malgré la diversité des méthodes de segmentation d'images, les recherches ont conclu qu'il n'existe aucune méthode générique. Récemment, une attention particulière est portée sur l'utilisation des algorithmes bio-inspirés pour résoudre les problèmes d'optimisation. La plupart de ces algorithmes sont inspirés de plusieurs processus naturels tels que la biologie, la physique, ou l'éthologie. Cette nouvelle tendance consiste à étudier et imiter les différents phénomènes, fonctions, et modèles des êtres vivants pour développer de nouvelles solutions dans plusieurs domaines d'application (médecine, industrie, réseaux, etc.). En segmentation d'images, les algorithmes bio-inspirés ont rencontré un vif succès grâce à leur simplicité, leur adaptabilité et leur capacité de traiter des données imprécises ou incomplètes pour accéder à un espace plus large de solutions.

L'objectif de cette thèse est de contribuer à améliorer les performances de la segmentation d'images IRM au moyen des algorithmes bio-inspirés. Ces algorithmes

stochastiques utilisent des stratégies et des astuces, afin de fournir des résultats approchés en un temps raisonnable. Ainsi, nous allons étudier l'hybridation des méthodes de segmentation avec les métaheuristiques bio-inspirées. L'intérêt de telle hybridation, est qu'elle exploite la complémentarité d'informations en proposant un système de segmentation plus efficace et plus robuste.

Dans cette thèse, nous focalisons notre étude sur la segmentation automatique des images IRM cérébrales. Parmi les méthodes non supervisées présentées, nous nous sommes intéressés aux approches floues et plus précisément, à l'algorithme Fuzzy *c*-means (FCM). L'avantage principal de cet algorithme, et son adaptation où les données manipulées se chevauchent ou lorsque l'information à traiter est imprécise et incertaine, ce qui est le cas des images IRM cérébrales. Cependant, son inconvénient majeur est lié à l'initialisation aléatoire des centres de classes qui provoque une convergence prématurée, et peut conduire l'algorithme à rester piégé dans des optima locaux, sans atteindre l'optimum global.

Cette thèse contient deux principales contributions. La première contribution est le développement d'une méthode de segmentation rapide, basée sur l'algorithme FCM. Le principe consiste à estimer la configuration initiale des centres de classes à partir des centres obtenus par FCM standard, sur un ensemble d'images IRM possédant des propriétés similaires à l'image à segmenter. L'objectif visé est de minimiser le temps d'exécution. Dans la seconde contribution, et pour pallier à l'inconvénient de la convergence prématurée de l'algorithme FCM, nous avons eu recours à l'algorithme d'optimisation par essaim de particules (PSO), qui présente l'avantage d'être simple à implémenter et bien adapté pour résoudre des problèmes d'optimisation continue. Cette contribution s'articule dans un premier temps sur la reformulation du problème de segmentation en un problème d'optimisation par la définition d'une nouvelle fonction objectif, afin de chercher les meilleurs centres de classes. L'algorithme FCM est ensuite initialisé avec les centres déjà obtenus.

Ce manuscrit est scindé en six chapitres :

Le premier chapitre présente le cadre applicatif du travail. Il rappelle quelques notions sur l'anatomie du cerveau humain et les principes fondamentaux de l'IRM. Il présente également les principaux artefacts qui affectent cette technique d'acquisition.

Le deuxième chapitre est consacré à l'étude bibliographique des différentes approches de segmentation d'images IRM, dont la plus grande partie concerne les approches non

supervisées. Les avantages et les inconvénients de chaque approche sont également présentés.

Dans le troisième chapitre, nous abordons le domaine d'optimisation en mettant l'accent sur les algorithmes bio-inspirés. Nous dressons notamment un état de l'art concernant les différentes métaheuristiques bio-inspirées appliquées au problème de la segmentation d'image.

Le quatrième chapitre est dédié à l'étude détaillée de l'algorithme PSO. Il explique le principe général de PSO et le rôle que jouent ses différents paramètres.

Dans le cinquième chapitre, nous détaillons notre première contribution qui se porte sur l'amélioration du temps d'exécution de FCM. Les évaluations des performances de l'approche proposée seront exposées.

Dans le sixième chapitre, nous exposons un état de l'art sur les différentes hybridations de l'algorithme FCM avec les approches bio-inspirées. Ensuite, nous présentons notre deuxième contribution, qui consiste à l'amélioration de la segmentation floue (FCM) grâce à une variante modifiée de PSO. Puis nous présentons les applications et les résultats de test à travers la segmentation d'images simulées de tissus normaux et des images réelles de tissus pathologiques. Différents critères proposés dans la littérature seront utilisés pour l'évaluation quantitative et qualitative de nos résultats.

Enfin, ce manuscrit s'achève par une conclusion générale, où nous dressons le bilan de notre travail, puis nous proposons des perspectives à considérer pour les futurs travaux.

Chapitre 1

Imagerie Cérébrale par Résonance Magnétique

Sommaire

1.1	Introduction	5
1.2	Eléments d'anatomie cérébrale	6
1.2.1	L'encéphale	6
1.2.2	Les principaux tissus cérébraux	7
1.3	Observation du cerveau	8
1.4	Imagerie médicale	9
1.5	Imagerie par Résonance Magnétique (IRM)	9
1.5.1	Principe général	9
1.5.2	Principe physique	10
1.5.3	Paramètres de séquence	10
1.5.4	Les séquences IRM	11
1.5.5	Artefacts des images IRM	12
1.6	Conclusion	13

1.1 Introduction

Dans la médecine moderne, l'imagerie médicale a connu d'importants progrès faisant appel à de nombreuses technologies avancées. L'exploitation des informations pertinentes sur le cerveau humain comporte de nombreuses applications cliniques utiles, qui aident

le praticien dans le processus de diagnostic de nombreuses maladies neurologiques (Alzheimer, Parkinson, épilepsie, etc.) ou psychiatriques (dépression, schizophrénie, autisme, etc.).

Ce chapitre a pour objectif de présenter le cadre applicatif que nous avons envisagé. Dans un premier temps nous présentons quelques notions élémentaires d'anatomie cérébrale, afin de préciser les termes qui seront utilisés le long de ce manuscrit. Dans un second temps, nous introduisons brièvement les principes physiques de l'imagerie par résonance magnétique (IRM), en illustrant les différentes structures cérébrales qui doivent être identifiées sur les images, tout en insistant plus particulièrement sur les artefacts qui affectent cette technique d'acquisition.

1.2 Eléments d'anatomie cérébrale

1.2.1 L'encéphale

C'est la partie supérieure du système nerveux central, constitué du tronc cérébral, du cervelet et du cerveau. Il assure le contrôle de l'ensemble de l'organisme [1]. L'encéphale est contenu à l'intérieur de la boîte crânienne (Figure 1.1) ¹.

- **Le cerveau** : c'est la partie la plus élevée, la plus volumineuse ² et la plus complexe de l'encéphale. Il se situe au-dessus du tronc cérébral et du cervelet. Le cerveau est partagé en deux hémisphères droit et gauche réunis par le cerveau moyen, ou diencephale, et par le corps calleux. Le cerveau est constitué, comme le reste du système nerveux central, de matière grise (corps des neurones et des synapses) et de matière blanche (fibres myélinisées). Le cerveau baigne dans un liquide jouant le rôle d'un amortisseur appelé liquide céphalo-rachidien (LCR).
- **Le cervelet** : partie de l'encéphale située à la base du crâne, en arrière du tronc cérébral, et responsable de la coordination de l'activité musculaire nécessaire à l'équilibre et aux mouvements.
- **Le tronc cérébral** : il forme la transition entre le cerveau et la moelle épinière. Il est responsable du contrôle de la respiration, de la fréquence cardiaque et de la pression sanguine. Le tronc cérébral est également impliqué dans les réflexes de vomissements et le contrôle de la douleur.

1. www.fondationmiro.org/le-gitc.html [Online ; 2017]

2. Poids moyen de 1 500 g chez un homme adulte

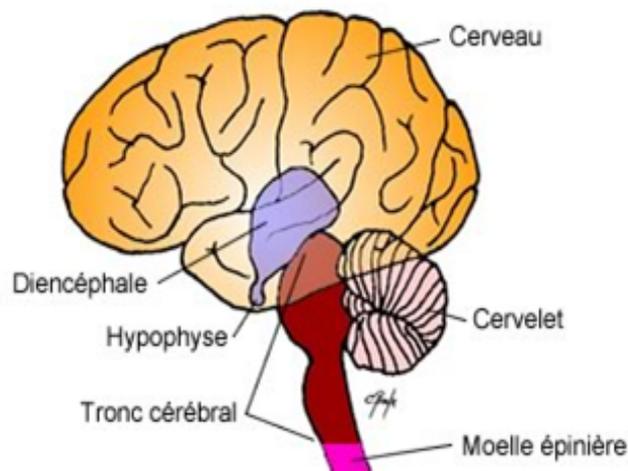


Figure 1.1: Structures anatomiques de l'encéphale.

1.2.2 Les principaux tissus cérébraux

- **Le liquide céphalo-rachidien (LCR)** : le liquide céphalo-rachidien ou cérébro-spinal (en anglais cerebro-spinal fluid) est une substance qui entoure le cerveau et remplit les ventricules cérébraux. Il permet de protéger le cerveau des chocs en jouant un rôle d'amortisseur, et assure des fonctions importantes dans le transport des hormones entre les différentes régions du cerveau.
- **La matière grise (MG)** : la matière grise constitue la partie des tissus du système nerveux central concentrant les corps cellulaires des neurones et des cellules gliales. Elle apparaît de couleur plus foncée que le reste des tissus (Figure 1.2) et elle est présente principalement au niveau du cortex cérébral, du cortex cérébelleux, dans les noyaux gris centraux, dans les noyaux du tronc cérébral et à l'intérieur de la moelle épinière. La matière grise constitue le cœur du traitement de l'information nerveuse : réception des messages, élaboration des réponses et l'analyse complexe des informations [1].
- **La matière blanche (MB)** : la partie interne du cerveau qui contient des fibres nerveuses myélinisées, elle est de couleur blanchâtre et brillante (couleur de la myéline). Le rôle de la substance blanche est de garantir la bonne circulation des informations entre les différentes parties du système nerveux.

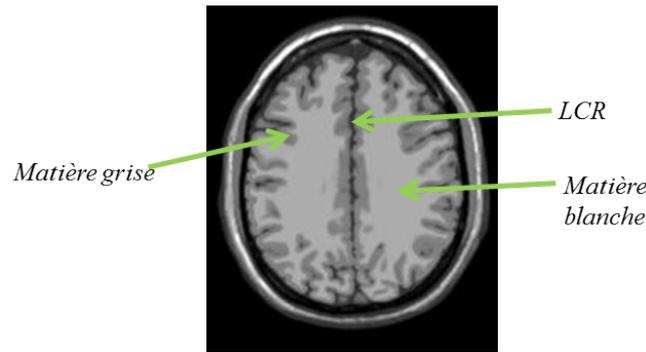


Figure 1.2: Les principaux tissus cérébraux sur une IRM.

1.3 Observation du cerveau

La visualisation du cerveau par des coupes en deux dimensions peut être effectuée selon plusieurs angles de vue. Ainsi, on distingue trois axes anatomiques pour réaliser les différentes coupes (Figure 1.3)³.

1. **Coupes axiales** : ces coupes correspondent à un plan horizontal. En IRM, elles correspondent à un plan perpendiculaire à l'axe du champ magnétique principal.
2. **Coupes sagittales** : ces coupes sont effectuées dans des plans parallèles au plan inter hémisphérique. Il s'agit des vues latérales du cerveau.
3. **Coupes coronales** : ce sont des coupes verticales aux coupes axiales et sagittales. Il s'agit des vues frontales du cerveau.

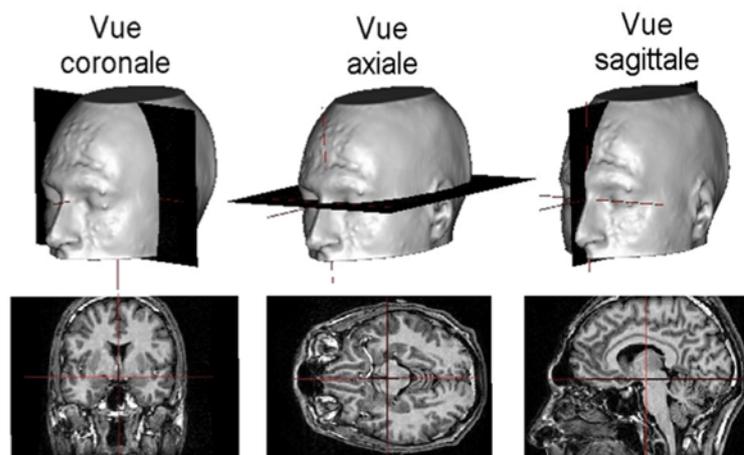


Figure 1.3: Les trois vues des IRM cérébrales.

3. www.cerveau-en-action.blogspot.com [Online ; 2017]

1.4 Imagerie médicale

L'imagerie médicale est certainement l'une des branches de la médecine qui a connu des progrès considérables depuis l'évolution de l'informatique. Elle est définie comme un ensemble de techniques consistant à visualiser en images les différentes régions ou différents organes du corps humain.

Il existe plusieurs techniques d'imagerie médicale qui sont utilisées en fonction des zones à étudier. On peut citer, de façon non exhaustive la radiologie en rayons X pour visualiser les os, les poumons, etc.; l'échographie, qui est une technique d'exploration des organes de l'abdomen du corps basée sur les ultrasons; le scanner⁴ qui balaie la région à explorer et reconstitue des images en tranches et qui permet de réaliser la représentation d'organes en 3D afin de déterminer précisément la localisation des lésions et enfin l'imagerie par résonance magnétique (IRM).

1.5 Imagerie par Résonance Magnétique (IRM)

L'imagerie par résonance magnétique (IRM) est une technique d'imagerie non invasive qui produit des images anatomiques 2D ou 3D détaillées. Elle est basée sur le principe de résonance magnétique nucléaire (RMN, terme créé en 1930 par Isidor Isaac Rabi⁵). La première image IRM fut publiée en 1973 par Paul Christian Lauterbur [2]. L'IRM est souvent utilisée pour la détection, le diagnostic et la surveillance de l'évolution des maladies présentes dans les tissus mous.

L'IRM cérébrale est généralement classée en deux catégories :

1. **IRM anatomique** : également appelée IRM structurelle, permet d'obtenir des informations sur la structure, la forme et le contenu des tissus cérébraux.
2. **IRM fonctionnelle (IRMf)** : l'imagerie fonctionnelle ne s'intéresse guère à l'anatomie cérébrale, mais à l'étude de l'activité cérébrale. L'IRMf est le plus souvent liée au domaine de la recherche des sciences cognitives et neurosciences.

1.5.1 Principe général

L'IRM est une modalité d'imagerie très performante, parfaitement non invasive, elle est basée sur le principe de la résonance magnétique des protons d'hydrogène au sein d'un

4. Aussi appelé tomodynamométrie

5. Prix Nobel de physique en 1944

champ électromagnétique. Ce dernier est créé par un aimant très puissant en forme de tunnel au sein duquel le patient est allongé. Les ondes électromagnétiques provoquent une aimantation des noyaux d'hydrogène présents dans les tissus. Le signal émis par les protons, lors de leur retour à leur état initial, est à l'origine de la création des images de grande précision anatomique. Les images obtenues sont traitées par des ordinateurs qui permettent la reconstruction des différentes coupes. L'examen IRM peut être procédé sans ou avec l'injection d'un produit de contraste dans la veine du patient afin d'observer plus précisément l'organe ciblé.

1.5.2 Principe physique

Quand un corps humain est placé dans un champ magnétique puissant B_0 , les noyaux d'hydrogène libres s'alignent dans la direction de B_0 . Le moment magnétique d'un proton va tourner très rapidement autour d'un axe en décrivant un cône de révolution.

- **Phénomène de la résonance** : c'est le transfert d'énergie entre deux systèmes oscillant à la même fréquence. Le proton d'hydrogène est basculé d'un état d'énergie E_1 à un état E_2 par l'application d'une onde radiofréquence RF.
- **Relaxation et temps de relaxation** : la relaxation correspond au retour à l'état d'équilibre de départ de l'aimantation tissulaire après l'arrêt de l'émission de la radiofréquence RF. Elle s'accompagne d'une émission d'énergie sous la forme d'ondes RF qui constituent le signal enregistré en IRM. Plus le signal émis du corps est intense, plus le point correspondant de l'image finale est blanc et inversement. Le temps de relaxation c'est la période de retour à l'équilibre des atomes d'hydrogène après leur résonance (excitation) par la radiofréquence RF. Pour chaque tissu, on distingue 2 temps de relaxation : la relaxation longitudinale (T1) et la relaxation transversale (T2).

1.5.3 Paramètres de séquence

Les paramètres de séquence sont les paramètres que l'opérateur fixe sur la console pour générer la séquence IRM correspondante.

- **Temps d'écho** : le temps d'écho (TE) fait référence au temps entre l'application de l'impulsion d'excitation radiofréquence RF et le sommet du signal induit dans la bobine. Il est mesuré en millisecondes.
- **Temps de répétition** : le temps de répétition (TR) est le temps de l'application

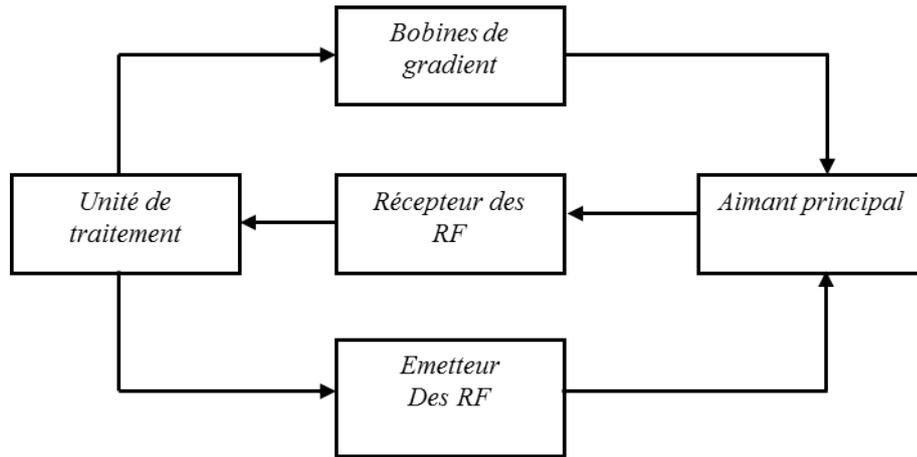


Figure 1.4: Schéma fonctionnel de l'IRM.

d'une impulsion d'excitation à l'application de la prochaine impulsion. Il détermine la quantité d'aimantation longitudinale qui se rétablit entre chaque impulsion. Il est mesuré en millisecondes. Le TR, comme le TE, est un facteur de contraste.

- **Angle de basculement** : l'angle de basculement α est un paramètre par lequel l'axe du proton d'hydrogène passe de son axe longitudinal (axe magnétique statique B_0) à son axe transversal par excitation à l'aide d'impulsions radiofréquences (RF). L'angle de basculement est mesuré en degrés ($^\circ$). Plus α est grand, plus les images en T1 sont contrastées.

1.5.4 Les séquences IRM

En changeant les valeurs des différents paramètres de séquence (TE, TR, α), une grande diversité de volumes variant par le contraste peuvent être générés.

- **Image pondérée en densité de protons (DP)** : pour un TR long et un TE court, on obtient une séquence pondérée en densité de protons DP. Les tissus sont ordonnés par niveaux de gris croissants en MB, MG et LCR.
- **Image pondérée en T2** : pour un TR long et un TE long, on obtient une image pondérée en T2. Les tissus sont ordonnés par niveaux de gris croissants en MB, MG et LCR.
- **Image pondérée en T1** : pour un TR court et un TE court, l'image obtenue est pondérée en T1. Les tissus sont ordonnés par niveaux de gris croissants en LCR, MG et MB.

Chaque séquence IRM contient des informations spécifiques, donc, le choix entre les différentes séquences dépend de la structure à étudier.

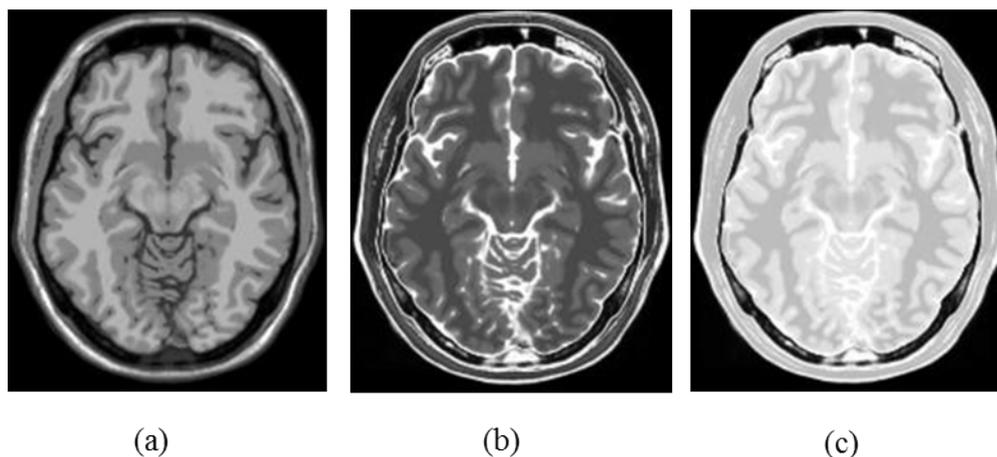


Figure 1.5: Les séquences IRM : (a) image pondérée en T1, (b) image pondérée en T2, (c) image pondérée en DP.

1.5.5 Artefacts des images IRM

Les sources d'artefacts (erreurs) en IRM sont nombreuses et de nature variée. Elles sont différentes de celles observées dans d'autres domaines du traitement d'images. Parfois, l'image IRM contient des structures qui ne sont pas réellement présentes. Ces artefacts peuvent imiter les pathologies, conduisant à des diagnostics incorrects. On distingue essentiellement trois types d'artefacts [3]. Les artefacts physiologiques qui sont causés par le mouvement du patient, y compris la respiration, le rythme cardiaque et le flux sanguin. Les artefacts qui peuvent découler de la physique inhérente de l'IRM, comme la présence de métal ou de changement chimique. Enfin, ceux liés aux matériels et logiciels impliqués dans la construction d'images IRM. On présente dans la suite de cette section, les principaux artefacts :

- **Le bruit** : le bruit en IRM a des origines variées, liées en grande partie aux propriétés physiques, au contexte d'acquisition et au patient lui-même (mouvement, agitation, etc.). L'objectif recherché par tous les constructeurs d'appareillage d'imagerie médicale, est focalisé sur un point principal [4] : améliorer le rapport signal/bruit tout en conservant une bonne résolution.
- **Effet du volume partiel** : l'effet du volume partiel se produit pendant le processus de numérisation du signal dans le cas où la résolution spatiale (voxel⁶) dépasse la taille des objets présents dans l'image ou lorsque les structures cérébrales se chevauchent . Ainsi, le niveau de gris de ce voxel sera un mélange des niveaux de

6. Voxel est un pixel en 3D

gris de chaque objet (Figure 1.6).

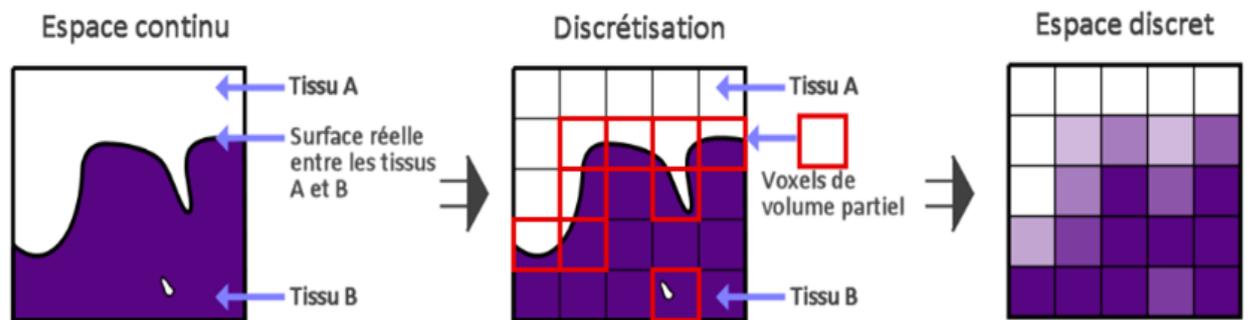


Figure 1.6: L'effet du volume partiel en IRM [4].

En IRM cérébrale, cet artefact se manifeste principalement aux frontières entre les substances cérébrales (MG, MB, LCR, graisse, os). L'effet du volume partiel est atténué par l'utilisation d'une taille de voxel plus petite.

- **Inhomogénéité de la radio fréquence (RF)** : la variation de l'intensité d'un même tissu à travers l'image peut être due à la défaillance de la bobine RF, au champ RF non uniforme spatialement et temporellement, ou à la sensibilité non uniforme de la bobine de réception.

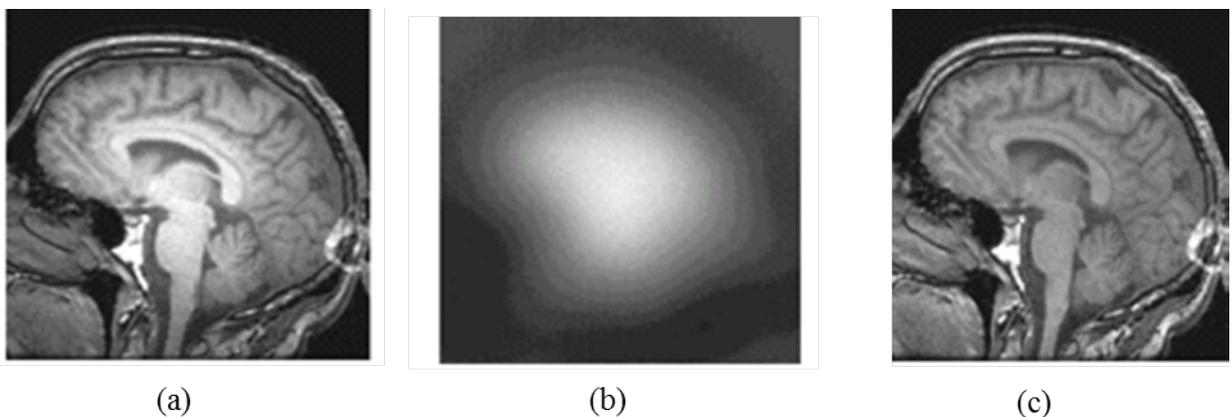


Figure 1.7: L'inhomogénéité RF : (a) Image affectée par une inhomogénéité RF, (b) L'artefact RF isolé, (c) Image sans artefact [5].

1.6 Conclusion

Nous avons présenté dans ce chapitre les notions essentielles liées à l'anatomie du cerveau humain ainsi que les principes d'acquisition de l'imagerie par résonance magnétique qui apporte une aide consistante en termes de diagnostic précoce et de caractérisation des tissus cérébraux.

Malgré le développement des techniques d'imagerie médicale, et en particulier l'IRM, qui est bien adaptée à l'observation des structures cérébrales, les praticiens rencontrent toujours des difficultés pour analyser les résultats d'acquisition, ceci est dû non seulement aux artefacts spécifiques à ce type d'imagerie (bruit, l'effet du volume partiel, etc.), mais aussi, à l'anatomie du cerveau humain qui est compliquée dans le sens où elle englobe de nombreuses structures cérébrales enchevêtrées, de formes et de composition variées.

Pour interpréter les images IRM cérébrales le médecin doit analyser avec précision les résultats afin d'aboutir à une décision finale. L'une des opérations indispensables et essentielles pour tout traitement d'image est évidemment la segmentation qui consiste à séparer les structures cérébrales (matière grise, matière blanche, liquide céphalorachidien), pour permettre d'identifier, comprendre et suivre l'évolution des pathologies. Dans le domaine médical, la précision est importante pour un diagnostic fiable. En termes de traitement d'images, il est nécessaire que la segmentation soit précise. Dans la littérature, différentes méthodes de segmentation des IRM cérébrales ont été développées. Elles feront l'objet du prochain chapitre.

Chapitre 2

Etat de l'art : Segmentation d'images IRM

Sommaire

2.1	Introduction	15
2.2	Définition	16
2.2.1	Définition générale	16
2.2.2	Définition formelle	17
2.3	Objectifs et applications de la segmentation	17
2.4	Spécificités de la segmentation IRM	18
2.5	Méthodes de segmentation	18
2.5.1	Approches contours	19
2.5.2	Approches régions	21
2.5.3	Segmentation par classification	24
2.6	Conclusion	31

2.1 Introduction

Face à l'accroissement du nombre d'examens pratiqués en imagerie médicale, les programmes de traitement d'images ont un rôle primordial à jouer afin de soutenir la tâche des praticiens. De tels programmes ne se limitent pas à des tâches de visualisation des images, ils ont plutôt pour objectif de parvenir à aider le praticien dans ses diagnostics, en lui signalant les risques pathologiques. L'une des opérations primordiales pour tout traitement d'images est évidemment la segmentation.

La segmentation est un processus important en traitement d'images, elle consiste à partitionner une image en plusieurs régions. Ce processus aidera à trouver des zones d'intérêt dans une image et également la rendre plus simple et plus significative à évaluer. Plus spécifiquement, la segmentation d'images IRM cérébrales, permet de délimiter les différents tissus cérébraux et éventuellement les structures pathologiques.

Nous commençons ce chapitre par définir formellement la segmentation d'images et présenter la spécificité des IRM par rapport à d'autres types d'images. Puis, nous présenterons un état de l'art sur les différentes méthodes pour la segmentation des IRM cérébrales. Toutefois, l'accent est mis sur les méthodes de segmentation par classification.

2.2 Définition

2.2.1 Définition générale

La segmentation des images est une étape essentielle dans les applications de vision par ordinateur. Elle est considérée comme la première étape de traitement d'image [6]. Elle consiste à diviser l'image en plusieurs régions homogènes selon un critère déterminé (niveau de gris, couleur, texture, etc.) [7] où chaque région est supposée correspondre à un objet de l'image.

La segmentation est dite *multimodale* si les informations traitées parviennent de plusieurs images obtenues sous différents paramètres d'acquisition par opposition à la segmentation *monomodale*.

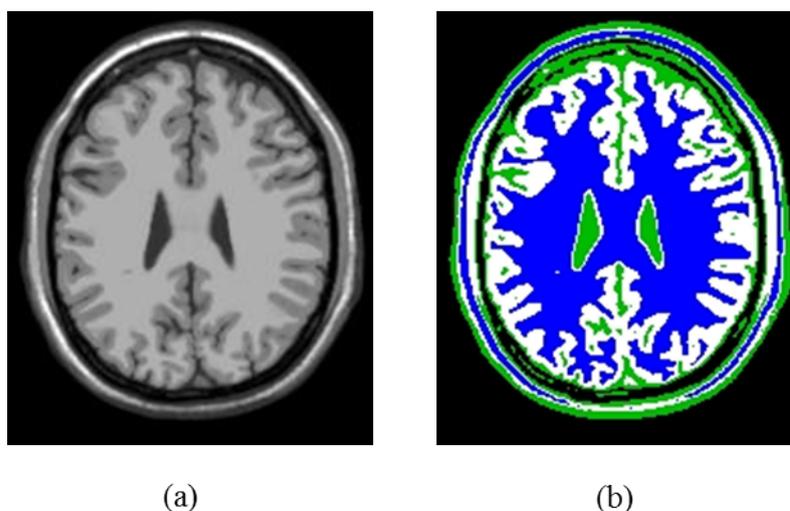


Figure 2.1: Exemple d'une segmentation : (a) image originale, (b) image segmentée.

2.2.2 Définition formelle

Une définition formelle de la segmentation d'images a été donnée par Horowitz et Pavlidis [8] :

Soit X le domaine de l'image et f la fonction qui associe à chaque pixel une valeur $f(x,y)$. Un prédicat d'homogénéité P est défini sur l'ensemble des parties de X , la segmentation de X est définie comme une partition de X en m sous-ensemble $\{S_1, \dots, S_m\}$ tels que :

- $X = \bigcup_{i=1}^m S_i$
- $S_i \cap S_j = \emptyset \quad \forall i, j : i \neq j$
- $P(S_i) = \text{vrai} \quad \forall i$
- $P(S_i \cup S_j) = \text{faux} \quad \forall i, j : i \neq j$

Quatre conditions sur la définition précédente ont été présentées dans [9] : la première condition implique l'appartenance de chaque pixel de l'image à une et une seule région. La seconde condition implique que toute région doit être connexe. La troisième condition exige que chaque région doit être homogène. Enfin, la quatrième condition indique que la fusion de deux régions ne doit pas produire une région homogène.

2.3 Objectifs et applications de la segmentation

La segmentation d'image est un traitement de bas niveau qui consiste à associer à chaque pixel d'une image un label, c'est une opération importante pour l'extraction des informations utiles comme : la forme, la taille et la position. La segmentation des structures cérébrales permet la caractérisation des différents tissus cérébraux (matière grise, matière blanche, liquide céphalorachidien, etc.) et aussi l'extraction des informations sémantiques pertinentes, qui seront utilisées dans un grand nombre de traitements.

Les principales applications de la segmentation en imagerie médicale sont la morphométrie, la cartographie fonctionnelle et la visualisation [10].

- **La morphométrie** : c'est une partie de la biométrie qui s'intéresse aux formes, elle étudie et analyse la géométrie d'organes ou de structures cérébrales afin d'identifier, comprendre et suivre la progression des pathologies comme l'épilepsie, l'Alzheimer, l'autisme ou la schizophrénie.
- **La cartographie fonctionnelle** : la segmentation des structures cérébrales visant à dévoiler le fonctionnement du cerveau pour localiser les signaux, les cartographier

et pour les visualiser.

- **Visualisation** : c’est la localisation des surfaces extérieures de structures anatomiques. Elle permet de donner une représentation précise (réaliste) de la structure cérébrale pour aider le médecin à planifier l’opération de chirurgie (chirurgie assistée par ordinateur).

2.4 Spécificités de la segmentation IRM

La segmentation des IRM cérébrales présente des spécificités par rapport à d’autres domaines d’applications de la segmentation, comme la segmentation d’images satellitaires, ou d’images faciales [10]. Ces spécificités sont liées aux particularités des objets à segmenter (l’anatomie du cerveau) et au processus d’acquisition des IRM. Ces spécificités sont caractérisées par :

- **Information d’intensité insuffisante** : plusieurs structures cérébrales peuvent avoir la même intensité pour une modalité donnée, ce qui engendre une difficulté de distinction de transition entre les tissus. L’utilisation simultanée de plusieurs modalités permet de pallier ce type de problème.
- **Perturbations pathologiques** : les changements pathologiques, comme les déformations des structures cérébrales peuvent perturber le processus de segmentation.
- **Connaissances *a priori*** : la position et la forme de quelques structures cérébrales internes/externes est assez prédictible. Ce qui permet d’utiliser ces connaissances *a priori* afin de faciliter la tâche de segmentation (supervisée). A noter que pour d’autres structures qui présentent une forte variabilité anatomique, l’utilisation de connaissances *a priori* pour guider la segmentation, pose des problèmes et perturbera le résultat final [10].
- **Nature des images IRM** : plusieurs artefacts apparaissent en IRM (Section 1.5.5) qui sont typiques à cette imagerie : le bruit lié au mouvement, l’effet de volume partiel et l’inhomogénéité de la radio fréquence RF.

2.5 Méthodes de segmentation

La segmentation d’images est un sujet de recherche très vaste qui fait partie des grands thèmes de traitement d’images. A ce titre, de nombreuses publications font état

de segmentation où les chercheurs ont déduit que la segmentation idéale n'existait pas, et le choix d'une méthode de segmentation est délicat [11]. Etant donné pour une seule image, on peut avoir plusieurs segmentations possibles. Une bonne méthode de segmentation sera donc celle qui simplifiera l'image sans pour autant réduire ou dégrader son contenu initial.

Dans ce qui suit, nous allons présenter de manière non exhaustive les diverses approches connues de segmentation, en les organisant selon la technique qui les régit. Ainsi, nous avons retenu trois catégories principales : contours, régions et classification.

2.5.1 Approches contours

Contrairement aux approches régions qui forment des zones homogènes, les approches contours se basent sur les brusques variations d'intensité dans l'image (discontinuités) afin d'extraire les contours des régions. Dans la suite, nous nous intéressons particulièrement aux méthodes dérivatives et aux modèles déformables.

2.5.1.1 Méthodes dérivatives

Les méthodes dérivatives permettent de repérer les ruptures dans une image et supposent que l'image est une fonction à deux variables. Un contour dans une image est défini comme l'endroit de variation significative de niveau de gris, il peut être présenté comme une *marche d'escalier* si le contour est net, comme une *rampe* si le contour est plus flou ou comme un *toit* s'il s'agit d'une ligne sur un fond uniforme [12] (Figure 2.2).

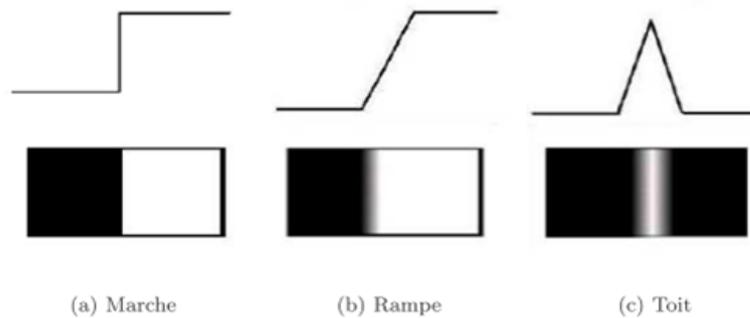


Figure 2.2: Les modèles de contours [12].

Les contours sont détectés à des points de forts gradients (dérivée première) ou de dérivées secondes nulles (Laplacien). On retrouvera alors des opérateurs classiques tels que les filtres de Roberts, Sobel ou de Prewitt. Il existe d'autres méthodes de détection de contour appelées méthodes optimales, comme les filtres de Canny [13], Deriche [14] ou Shen et Castan [15].

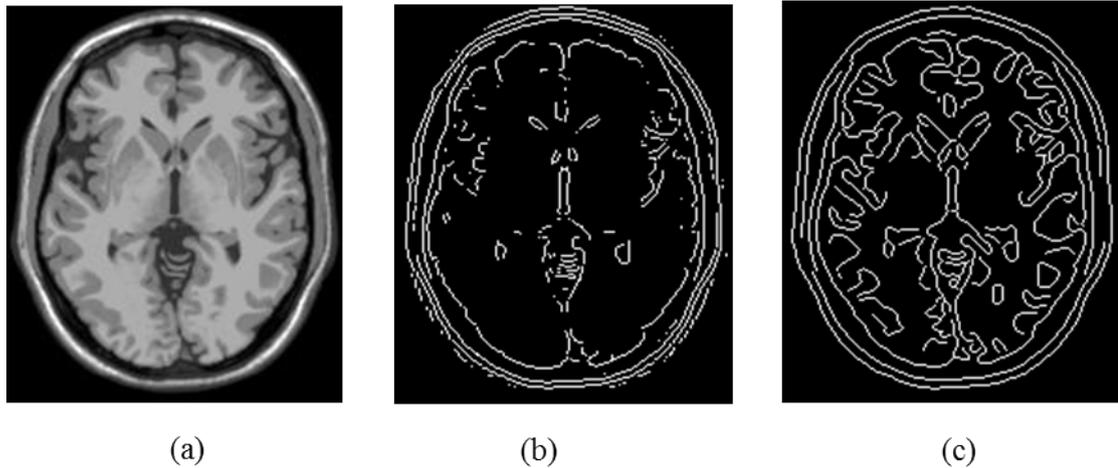


Figure 2.3: Segmentation par approches dérivatives : (a) image originale, (b) filtre Prewitt, (c) filtre Canny.

Dans la littérature, les contributions des méthodes dérivatives à la segmentation des images IRM cérébrales, sont peu nombreuses. Bien que ces méthodes sont simples et rapides, elles sont très sensibles aux bruits et elles ne génèrent pas des contours fermés. Leur efficacité repose sur l'existence de contraste entre les structures recherchées. Ces méthodes ont cependant des potentiels limités et ne sont plus guère utilisées de nos jours [12].

2.5.1.2 Modèles déformables (Snake)

Ces algorithmes dérivent des méthodes de contours dynamiques [16]. Le principe de base est de faire placer une courbe (en deux dimensions) formée d'une série de points mobiles (Figure 2.4)⁷ dans une zone d'intérêt de l'image. La courbe se déplace jusqu'à une position d'équilibre par l'intermédiaire de forces externes (bords, etc.) et internes (morphologie, courbure, etc.) par la minimisation itérative d'une fonctionnelle.

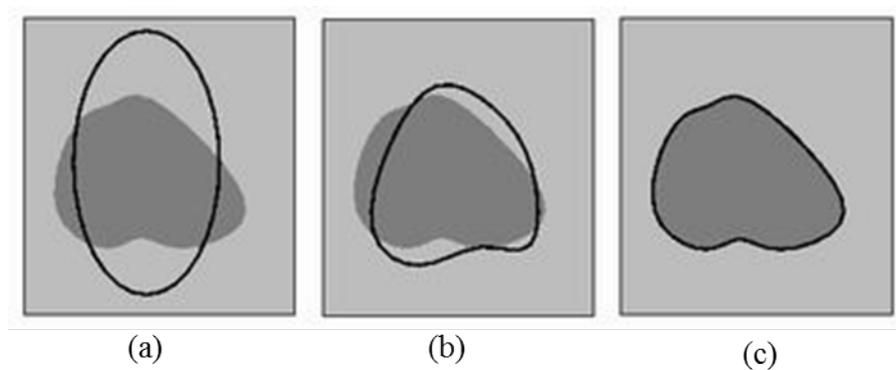


Figure 2.4: Principe du contour actif : (a) initialisation, (b) convergence, (c) résultat .

7. <http://sylvain.nicol.pagesperso-orange.fr/stage.htm> [Online; 2017]

Les modèles déformables ont été largement appliqués à la caractérisation des structures cérébrales en IRM [17–19]. Dans [17], un modèle déformable est utilisé pour la séparation des deux matières du cerveau (blanche / grise). La méthode s'avère efficace même lorsque l'initialisation est éloignée. Les résultats expérimentaux montrent la précision de cette approche, même avec la présence des artefacts liés à l'IRM. Mostaar *et al.* [18] ont introduit un nouveau modèle statistique de contour actif pour la segmentation d'images IRM en présence d'une inhomogénéité intense, avec la capacité de déterminer la taille et la position du contour initial, ce qui rend la segmentation plus rapide et plus précise. Leur modèle combine l'information des pixels voisins qui appartiennent à la même classe, ce qui permet de séparer les objets de l'arrière-plan. Dans cette étude, la détermination de la position, la taille du contour et du seuil de la fonction énergétique est basée sur une évaluation visuelle.

Les modèles déformables s'avèrent efficaces pour la segmentation de structures cérébrales particulières [19]. Contrairement aux méthodes dérivatives, ces approches produisent des contours ou des surfaces fermés en raison de leur topologie flexible et de leur robustesse aux variations d'intensités. Cependant, ces méthodes souffrent du problème d'initialisation, et face à ce problème, plusieurs travaux tendent à inclure des fonctions d'initialisation adaptatives [20].

2.5.2 Approches régions

Les approches régions visent à partitionner l'image en un ensemble de régions homogènes selon un ou plusieurs critères de similarité. Par la suite, nous allons présenter les méthodes à base de seuillage, les méthodes par croissance de régions, les méthodes division/fusion et finalement, celles fondées sur la morphologie mathématique.

2.5.2.1 Seuillage

Les méthodes de segmentation par seuillage visent à discriminer les pixels en fonction de leurs niveaux de gris. Le plus souvent, les seuils sont déterminés à partir des histogrammes. L'opération de seuillage permet d'isoler facilement les différentes régions contenues dans l'image, et qui correspondent aux différents pics de l'histogramme. On distingue trois types de seuillage : local, global, et dynamique.

L'opération de binarisation (seuillage simple), consiste à mettre à zéro tous les pixels dont le niveau de gris est inférieur à une valeur appelée seuil S , et à la valeur maximale les autres pixels. Ainsi le résultat du seuillage est une image binaire contenant des pixels

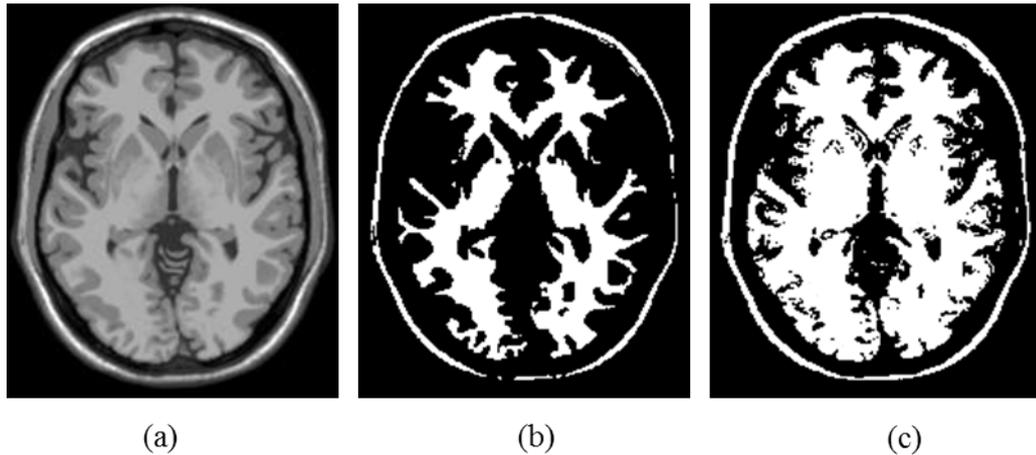


Figure 2.5: Segmentation par seuillage simple : (a) image originale, (b) image binaire (seuil=0.6), (c) image binaire (seuil=0.5).

noirs et blancs (Figure 2.5).

Soit $I(x,y)$, la valeur radiométrique du pixel de coordonnées (x,y) dans l'image, S étant le seuil fixé. Si $I(x,y) > S$, le pixel prend la valeur 1, sinon il prend la valeur 0. L'opération de multi-seuillage permet de définir plusieurs seuils afin de regrouper les pixels dans différentes classes, où chaque classe est délimitée par deux seuils.

Les méthodes de segmentation par seuillage ont été largement appliquées à la segmentation des IRM cérébrales. Par exemple, dans [21], Evelin *et al.* ont appliqué un algorithme de segmentation par seuillage, pour la séparation des structures cérébrales pathologiques et saines (MG, MB et LCR) sur des images IRM. Mozaffari *et al.* [22] ont proposé une nouvelle approche de seuillage basée sur l'optimisation par essaim de particules (PSO). Les auteurs ont utilisé les deux techniques de seuillage automatique Otsu [23] et Kapur [24] comme des fonctions objectifs de PSO. Les résultats de test de cette approche, montrent la robustesse et l'efficacité de cette approche. Dans [25], un algorithme de segmentation par seuillage est appliqué sur des images IRM, pour détecter la maladie d'Alzheimer.

Les méthodes de segmentation par seuillage sont des méthodes rapides et simples à implémenter. Néanmoins, elles sont très sensibles aux bruits et donnent des résultats insuffisants lorsque les structures à rechercher ne sont pas contrastées (recouvrement de structure). Leur utilisation nécessite, en particulier, la prise en compte des informations liées au voisinage d'un pixel. Enfin, le choix de la valeur du seuil S , n'est pas toujours facile.

2.5.2.2 Croissance de régions

Les méthodes dites croissance de régions sont fondées sur l'agrégation progressive des pixels à partir des points initiaux appelés *germes*, selon un critère d'homogénéité. Les pixels voisins de chaque germe sont ajoutés successivement à la région et deviennent à leur tour un germe, s'ils vérifient le critère d'agrégation. Dans le cas contraire, ils sont rejetés. La croissance s'arrête lorsque tous les pixels ont été traités. En général, le critère de similarité est basé sur les niveaux radiométriques des pixels.

Mohd Saad *et al.* [26] ont utilisé un algorithme de croissance de région automatique pour la détection des lésions présentes en IRM. La sélection des germes initiaux dans leur approche est déterminée à partir des histogrammes. Les résultats expérimentaux sur 4 types de lésions, montrent l'efficacité de cette approche. Dans [27], les différentes structures cérébrales sont segmentées par un algorithme automatique basé sur la croissance de régions sur des IRM saines et pathologiques.

Tout comme les méthodes à base de seuillage, les approches de segmentation par croissance de régions sont simples à implémenter mais parfois gourmandes en temps de calcul [19]. Leur avantage réside dans la prise en compte des informations contextuelles, qui permet la connexité des zones segmentées. Cependant, leur inconvénient est lié à la sélection des germes initiaux, surtout pour la détection des tumeurs, dont l'information *a priori* n'est pas connue.

2.5.2.3 Division/fusion (split & merge)

L'algorithme split & merge été proposé par Horowitz et Pavlidis en 1974 [28]. Dans la première étape, ou division, on teste d'abord le critère de similarité sur l'image entière, si le critère est vérifié, alors l'image est considérée comme segmentée. Sinon, l'image initiale sera découpée en blocs plus petits (généralement en 4 quadrants) et le processus est réappliqué sur chacun des nouveaux blocs obtenus. Dans la deuxième étape, ou fusion, à partir d'un graphe d'adjacence, si l'union de deux régions voisines vérifie le critère d'homogénéité, alors, on fusionne ces régions et on met à jour le graphe d'adjacence.

L'inconvénient de la méthode split & merge réside dans le choix de l'ordre dans lequel on commence la phase de fusion. Nous notons ici, son utilisation restreinte dans le contexte de la segmentation d'IRM cérébrales [29, 30].

2.5.2.4 Morphologie mathématique

Introduite par Serra [31] en 1982, la morphologie mathématique est une méthodologie ensembliste permettant l’analyse des structures géométriques. Elle permet d’étudier les objets en fonction de leur forme, taille, voisinage, et de leurs intensités (couleur, niveaux de gris, etc.) par des outils de base tels que l’érosion, la dilatation, la fermeture et l’ouverture. L’algorithme de morphologie mathématique le plus connu est LPE (Ligne de partage des eaux) [32].

En segmentation des IRM cérébrales, Brumer *et al.* [33] ont utilisé des érosions, des dilatations et des étiquetages en composantes connexes après une étape de seuillage afin de séparer les structures non cerveau de la structure cerveau et d’affiner le masque initial. L’algorithme a été testé avec succès sur un ensemble de 23 IRM de coupes coronales. Un nouvel algorithme de segmentation des IRM cérébrales d’un nouveau-né, basé sur la morphologie mathématique est présenté dans [34], l’avantage de cette contribution, est que la segmentation est faite automatiquement et ne nécessite aucune intervention manuelle.

Généralement, la morphologie mathématique est utilisée conjointement avec d’autres techniques pour segmenter les structures cérébrales [19].

2.5.3 Segmentation par classification

De manière formelle, les approches de segmentation par classification consistent à regrouper des pixels dans un ensemble de classes préalablement connues. Dans le cadre de la segmentation des images IRM cérébrales, les individus correspondent aux niveaux radiométriques des pixels étudiés, le nombre de classes (cluster) est égal aux nombres de tissus cérébraux.

Parmi les approches de classification, on distingue les méthodes supervisées pour lesquelles les caractéristiques de chaque classe sont connues *a priori* et les méthodes non supervisées (automatiques).

Dans cette partie nous allons présenter brièvement quelques méthodes supervisées, puis nous étudions quelques algorithmes non supervisés, tout en insistant particulièrement sur les algorithmes de classification floue.

2.5.3.1 Méthodes supervisées

Les méthodes de classification supervisée nécessitent des informations *a priori* sur les objets à segmenter, donc, un ensemble d’images étiquetées manuellement en C classes

par un ou plusieurs experts (radiologues). Chaque classe C_i est affectée à un ensemble d'apprentissage [35].

Parmi les méthodes de segmentation supervisées appliquées à la segmentation des images IRM, on peut citer : la segmentation bayésienne [36], la segmentation Markovienne [37], algorithme des k plus proches voisins (KPPV) [38], et les réseaux de Neurones [39].

Les approches supervisées sont trop dépendantes de l'opérateur et nécessitent de créer une base d'apprentissage pour chaque image, ce qui représente une tâche longue et pénible pour les experts, étant donné que la structure anatomique du cerveau est différente d'un patient à l'autre. Plus généralement, des auteurs notent que des petites différences d'étiquetage de l'expert dans de la phase d'apprentissage peuvent causer de grandes variations dans les résultats, ce qui demande une intervention manuelle constante de l'opérateur [35].

2.5.3.2 Méthodes non supervisées

Les méthodes non supervisées, visent à regrouper automatiquement les pixels en classes sans aucune base préalable d'apprentissage et aucune tâche d'étiquetage. L'expert intervient seulement à la fin du processus pour comparer les classes obtenues avec les classes réelles.

Parmi les méthodes non supervisées, on peut citer l'algorithme K-means [40], Floue c-means(FCM) [41], Possibilistic c-means(PCM) [42], Estimation-maximisation(EM) [43] et l'algorithme Mean shift [44].

Par la suite, nous allons décrire les algorithmes qui nous ont paru utiles pour notre étude. La présentation détaillée de l'algorithme FCM, permettra dans les chapitres suivants d'introduire la solution que nous avons adoptée.

Notons $O = \{o_1, o_2, \dots, o_N\}$ une image avec N pixels, qui sera partitionnée en C ($2 \leq C \leq N$) classes (clusters), est o_i représente le vecteur caractéristiques du pixel i . Z_k le k-ème centre de classe. $\|\cdot\|^2$ désigne la distance euclidienne.

(a) Algorithme K-means

Dans cet algorithme, un pixel o_i est attribué à la classe la plus proche parmi les C définies. Dans ce cas, la fonction objectif à minimiser est définie comme suit :

$$J = \sum_{k=1}^C \sum_{i=1}^N u_{ki} \|o_i - Z_k\|^2 \quad (2.1)$$

et les solutions au problème s'écrivent :

$$u_{ki} = \begin{cases} 1 & \text{si } \|o_i - Z_k\|^2 < \|o_i - Z_j\|^2 \forall k \neq j \\ 0 & \text{sinon} \end{cases} \quad (2.2)$$

$$Z_k = \frac{\sum_{i=1}^N u_{ki} o_i}{\sum_{i=1}^N u_{ki}} \quad (2.3)$$

La classification de tous les pixels s'effectue de manière itérative en répétant l'étape de classification (Eq. 2.2) et l'étape de mise à jour des centroïdes (Eq. 2.3), jusqu'à stabilisation (convergence) de la fonction objectif (Eq. 2.1).

L'algorithme ISODATA (Iterative Self-Organizing Data Analysis Technique Algorithm) [45] issu du K-means, permet de classer les pixels sans connaître *a priori* le nombre de classes qui pourra être modifié au cours des itérations. Il est basé sur le principe de division-fusion où deux classes sont fusionnées, si la distance entre leurs centres est inférieure à un seuil prédéfini, et une classe est divisée en deux, si la variance entre ses points est supérieure à un seuil prédéfini [46].

Algorithme 2.1 : K-means

```

1 début
2   Fixer le nombre de classes  $C$ , le critère d'arrêt  $\varepsilon$  ;
3   Initialiser aléatoirement le vecteur des centroïdes  $Z$  (espace niveaux de gris) ;
4   répéter
5     Calculer la distance de chaque pixel  $o_i$  aux centroïdes;
6     Affecter chaque pixel à la classe la plus proche (Eq. 2.2) ;
7     Mettre à jour le vecteur des centroïdes  $Z$  (Eq. 2.3);
8   jusqu'à  $|J^{(t)} - J^{(t+1)}| < \varepsilon$ ;
9 fin

```

L'algorithme K-means a été largement appliqué en segmentation des images IRM [47], son inconvénient majeur est que la classification finale dépend de la partition initiale qui est générée aléatoirement, et l'algorithme risque d'être stagné autour d'optima locaux [46]. Dans la littérature, plusieurs travaux [48, 49] ont été proposés pour résoudre ce problème en améliorant la phase d'initialisation des centres de classes.

Dans la méthode K-means, les pixels sont classés de façon certaine comme

appartenant à une et une seule classe. Cette affirmation ne reflète pas la réalité physique des images IRM cérébrales (chevauchement des tissus, bruit, volume partiel, etc.). La méthode Fuzzy c-means (FCM) qui sera présentée dans le paragraphe suivant, permet d'obtenir une segmentation floue qui prend en compte l'imprécision et l'incertitude des données.

(b) **Algorithme Fuzzy c-means (FCM)**

Le FCM, proposé par Dunn [50] et développé comme algorithme général de classification par Bezdek [41], est l'algorithme de classification le plus utilisé pour la segmentation d'images IRM cérébrales en raison de sa simplicité et de son applicabilité [51].

En FCM, le pixel o_i n'est plus assigné à une unique classe, mais à plusieurs par l'intermédiaire de degrés d'appartenance $U_{ik} \in [0, 1]$ du pixel o_i à la classe k , plus l'appartenance est forte, plus le degré est proche de 1, et inversement. A la fin de l'algorithme FCM, chaque pixel est associé à la classe dont le degré d'appartenance est le plus élevé. Dans la suite, on va suivre la description de FCM, présentée dans [35] :

- La matrice $U(N, C) = [u_{ik}]$ est appelée matrice de degrés d'appartenance si et seulement si elle satisfait les conditions suivantes :

$$\forall k \in \{1..C\}, \forall i \in \{1..N\}, u_{ik} \in [0, 1], \sum_{i=1}^N u_{ik} < N \quad (2.4)$$

$$\sum_{k=1}^c u_{ki} = 1 \quad (2.5)$$

Le degré d'appartenance u_{ki} est calculé comme suit :

$$u_{ki} = \frac{1}{\sum_{l=1}^c \left(\frac{\|o_i - Z_k\|}{\|o_i - Z_l\|} \right)^{2/(m-1)}} \quad (2.6)$$

- Les centres de classes sont calculés par la formule suivante :

$$Z_k = \frac{\sum_{i=1}^N u_{ki}^m o_i}{\sum_{i=1}^N u_{ki}^m} \quad (2.7)$$

- La fonction objectif à minimiser est définie comme suit :

$$J = \sum_{k=1}^C \sum_{i=1}^N u_{ik}^m \|o_i - Z_k\|^2 \quad (2.8)$$

où $m > 1$: est un paramètre contrôlant le degré de flou de la partition résultante, et $\|o_i - Z_k\|^2$ est la distance du pixel o_i au centre Z_k .

L'algorithme (FCM) (Algorithme 2.2) consiste alors à l'itération des équations (Eq. 2.6) et (Eq. 2.7) jusqu'à la convergence de la fonction objectif définie par l'équation (Eq. 2.8). Le critère d'arrêt définissant cette convergence, peut-être par exemple : la stabilité des centres de classe sur deux itérations successives ou un nombre maximum d'itération *itermax*.

Algorithme 2.2 : FCM

```

1 début
2   Fixer le nombre de classes  $C$ , le critère d'arrêt  $\varepsilon$ , le degré flou  $m$ ,  $itermax$  ;
3   Initialiser aléatoirement le vecteur des centroïdes  $Z$  (espace niveaux de gris) ;
4   Pour  $i \leftarrow 1$  à  $itermax$  faire
5     Mettre à jour la matrice  $U$  par (Eq. 2.6);
6     Mettre à jour le vecteur des centroïdes  $Z$  (Eq. 2.7);
7     Calculer la fonction objectif  $J$  (Eq. 2.8);
8     Si  $|J^{(t)} - J^{(t+1)}| < \varepsilon$  alors
9       break;
10    finsi
11  finpour
12 fin

```

A la fin du déroulement de FCM, une dernière étape est nécessaire lorsque le résultat souhaité est une classification dure (non floue). On parle alors de défuzzification. Cette étape considère la classe finale d'un pixel comme celle pour laquelle le degré d'appartenance est maximal.

L'algorithme FCM conventionnel a été beaucoup appliqué pour la segmentation des images IRM cérébrales [51–53]. L'avantage principal de cet algorithme, et son adaptation dans le cas où les données manipulées se chevauchent ou lorsque l'information disponible est imprécise et incertaine, ce qui est le cas des images utilisées dans notre recherche. Néanmoins, FCM n'est pas robuste face au bruit et son efficacité dépend fortement de l'étape d'initialisation, qui conduit à des optima

locaux si elle est mal choisie [54]. Un autre inconvénient, est que la segmentation en FCM, est déterminée uniquement par les intensités des pixels, c'est-à-dire sans prendre en compte l'information spatiale (notion de voisinage) [51].

Pour pallier les inconvénients de l'algorithme FCM, et surtout le problème d'initialisation, plusieurs travaux de recherche ont été présentés dans la littérature, qui feront l'objet du sixième chapitre.

(c) **Algorithme Possibilistic c-means (PCM)**

Krishnapuram et Keller dans [42], ont présenté une implémentation spécifique de la méthode possibiliste de Zadeh [55]. Cette implémentation affecte chaque pixel à plusieurs classes par des degrés d'appartenance qui ne vérifient pas la contrainte donnée par l'équation (Eq. 2.5). Donc, un nouvel ensemble de contraintes est défini :

$$\forall k \in \{1..C\}, \forall i \in \{1..N\}, \max_k(u_{ik}) > 0 \quad (2.9)$$

La condition (Eq. 2.9) assure que la matrice U résultante n'est plus une C-partition floue, puisque la contrainte (Eq. 2.5) n'est plus vérifiée.

Avec ces nouvelles contraintes, Krishnapuram et Keller [42] proposent de minimiser la fonction suivante :

$$J = \sum_{i=1}^N \sum_{k=1}^C u_{ik}^m \|o_i - Z_k\|^2 + \sum_{k=1}^C \eta_k \sum_{i=1}^N (1 - u_{ik})^m \quad (2.10)$$

où η_i est un paramètre positif déterminant la distance à laquelle le degré d'appartenance d'un vecteur à la classe k est égal à 0.5, il est calculé comme suit :

$$\forall k \in \{1..C\}, \eta_k = \frac{\sum_{i=1}^N u_{ik}^m \|o_i - Z_k\|^2}{\sum_{i=1}^N u_{ik}^m} \quad (2.11)$$

La matrice des degrés d'appartenance U et les centres de classes Z sont obtenus comme suit :

$$\forall k \in \{1..C\}, \forall i \in \{1..N\}, u_{ik} = \frac{1}{1 + \left(\frac{\|o_i - Z_k\|^2}{\eta_k}\right)^{\frac{1}{m-1}}} \quad (2.12)$$

$$\forall k \in \{1..C\}, Z_k = \frac{\sum_{i=1}^N u_{ik}^m o_i}{\sum_{i=1}^N u_{ik}^m} \quad (2.13)$$

Notons que dans la littérature, plusieurs variantes de la fonction objectif (Eq. 2.10) ont été proposées [55].

L'utilisation de l'algorithme PCM (Algorithme 2.3) seul en segmentation d'images IRM cérébrales est peu fréquente, il est généralement hybridé avec d'autres approches de segmentation. Masulli et Schenone [56] ont coopéré une approche par réseaux de neurones avec l'algorithme PCM pour segmenter les tissus cérébraux et des entités pathologiques (méningiomes). Dans [57], une nouvelle approche hybride appelée *enhanced possibilistic fuzzy c-means clustering* (EPFCM) est proposée pour segmenter des images IRM pathologiques et normales. La méthode proposée intègre le degré d'appartenance de FCM, la typicalité de PCM et l'information spatiale locale et globale pour classer chaque pixel. Cette incorporation est réalisée en modifiant la mesure de distance. Les résultats expérimentaux sur des images synthétiques et réelles montrent que l'algorithme proposé est plus précis et plus robuste que d'autres variantes FCM. Khotanlou et Afrasiabi [58] ont segmenté des images IRM contenant des lésions, par une approche appelée SCPFCM qui utilise le degré d'appartenance de FCM, la typicalité de PCM et l'information spatiale. La méthode proposée, repose sur une segmentation initiale (T1-T2) des lésions, puis la séquence T1 est utilisée comme un masque et comparée avec la séquence T2.

Contrairement à FCM, l'algorithme PCM est bien adapté à la segmentation des données aberrantes (bruit) présentes dans une image IRM. Cependant, PCM souffre des inconvénients suivants [59, 60] : les résultats de la classification dépendent fortement de la sélection et/ou de l'initialisation des paramètres ; la précision de la classification se détériore souvent en raison du problème de coïncidence des centres.

Algorithme 2.3 : PCM

```
1 début
2   Fixer le nombre de classes  $C$ , le critère d'arrêt  $\varepsilon$ , le degré flou  $m$ ,  $itermax$  ;
3   Initialiser aléatoirement le vecteur des centroïdes  $Z$  (espace niveaux de gris) ;
4   Pour  $i \leftarrow 1$  à  $itermax$  faire
5     Mettre à jour la matrice  $U$  par (Eq. 2.12);
6     Mettre à jour le vecteur des centroïdes  $Z$  (Eq. 2.13);
7     Calculer le paramètre  $\eta_k$  (Eq. 2.11);
8     Calculer la fonction objectif  $J$  (Eq. 2.10);
9     Si  $|J^{(t)} - J^{(t+1)}| < \varepsilon$  alors
10      |   break;
11      fin
12   finpour
13 fin
```

2.6 Conclusion

Dans ce chapitre, nous avons présenté d'une manière non exhaustive l'ensemble des approches de segmentation des images IRM cérébrales. Nous avons mis en évidence la diversité de ces approches selon la technique qui les régit. Ainsi, nous l'avons classé en trois catégories principales : contours, régions et classification.

La diversité des approches reflète la complexité et les difficultés du problème de segmentation, ces difficultés sont dues, d'une part à l'anatomie complexe du cerveau humain qui est composé de structures enchevêtrées à caractères variables, et d'autre part à la particularité des images IRM qui sont sujettes à des incertitudes et des imprécisions, liées en particulier, aux bruits d'acquisition, aux effets de volume partiel et à l'inhomogénéité de la radio fréquence.

D'après l'étude des différentes méthodes citées dans ce chapitre, nous avons relevé des inconvénients et des avantages pour chacune d'entre elles. Ce qui nous a permis de confirmer qu'il n'existe pas de méthode de segmentation idéale (résultats optimaux). Nous avons pensé donc à utiliser la coopération de méthodes pour tirer parti des avantages de chacune, l'intérêt de telles approches est qu'elles exploitent la complémentarité d'informations en proposant un système de segmentation efficace.

Chapitre 3

Algorithmes bio-Inspirés

Sommaire

3.1	Introduction	33
3.2	Définition	34
3.3	Le problème d'optimisation	35
3.4	Les méthodes d'optimisation	35
3.4.1	Les méthodes déterministes (exactes)	35
3.4.2	Les méthodes non-déterministes (métaheuristiques)	36
3.5	Les algorithmes bio-inspirés	37
3.5.1	Les automates cellulaires (CA)	37
3.5.2	Recuit simulé (SA)	38
3.5.3	Les algorithmes génétiques (GA)	39
3.5.4	Algorithme de colonies de fourmis (ACO)	41
3.5.5	Algorithme à évolution différentielle (DE)	43
3.5.6	Optimisation par essaim de particules (PSO)	43
3.5.7	Systèmes immunitaires artificiels (AIS)	44
3.5.8	Algorithme de recherche d'harmonie (HS)	45
3.5.9	Algorithme d'essaims de poissons artificiels (AFSA)	45
3.5.10	Optimisation par recherche bactérienne (BFO)	46
3.5.11	Algorithme de colonie d'abeilles (ABC)	46
3.5.12	Algorithme de Recherche gravitationnelle (GSA)	47
3.5.13	Algorithme Firefly (FA)	48
3.5.14	Algorithme de Chauve-souris (BA)	49
3.5.15	Autres algorithmes bio-inspirés	49
3.6	Conclusion	50

3.1 Introduction

Dans le chapitre précédent, nous avons posé la problématique de la segmentation d'images IRM, et on a remarqué que malgré la diversité des approches de segmentation,

les chercheurs ont conclu à l'inexistence d'une approche idéale, applicable à toutes les images.

Dans ce chapitre, nous allons considérer cette problématique du point de vue problème d'optimisation. On commence tout d'abord par présenter le problème d'optimisation en général. On passe par la suite aux différentes méthodes d'optimisation où on va dresser un état de l'art sur les différentes métaheuristiques bio-inspirées et leurs applications à la segmentation d'images.

3.2 Définition

Selon le sens mathématique, l'optimisation d'une fonction $f(s)$ où $s \in S$ (S est un ensemble de solutions) consiste à trouver une solution $s_0 \in S$, dite solution optimale, tel que $\forall s \in S$ [61] :

- Pour un problème de minimisation $f(s_0) \leq f(s)$
- Pour un problème de maximisation $f(s_0) \geq f(s)$

La fonction f , appelée la fonction objectif ou fitness, représente une mesure quantitative de la qualité des solutions.

A noter que chaque problème de minimisation peut se transformer sous la forme d'un problème de maximisation et inversement.

Dans le domaine d'optimisation on distingue deux types de minima [62] :

- **Minimum local** : la solution s est minimum local dans un voisinage N si $\forall s' \in N(s), f(s) \leq f(s')$
- **Minimum global** : la solution s est minimum global, si $\forall s' \in S, f(s) \leq f(s')$

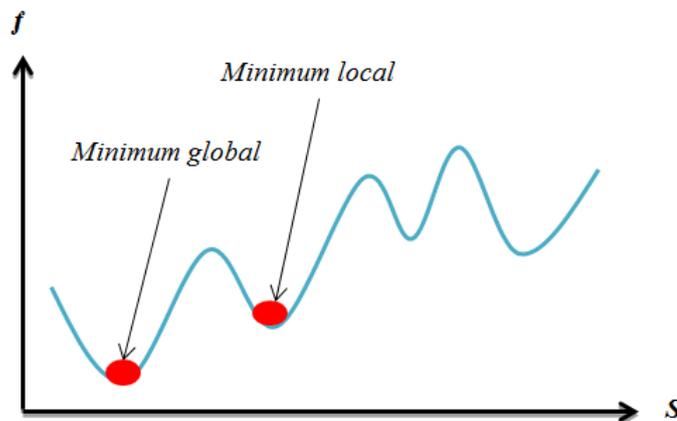


Figure 3.1: Changement de la fonction objectif pour un problème de minimisation.

3.3 Le problème d'optimisation

Un problème d'optimisation est défini par un ensemble de variables, un ensemble de contraintes sur ces variables et une fonction objectif que l'on cherche à minimiser/maximiser. Chaque problème d'optimisation possède un espace de recherche qui représente l'ensemble des solutions possibles du problème.

La résolution d'un problème d'optimisation, revient à chercher la ou les meilleures solutions, tout en satisfaisant l'ensemble de contraintes.

Un problème d'optimisation peut être classé de différentes manières [63] :

- Statique ou dynamique (la fonction objectif change avec le temps : les bourses économiques, le routage dans les réseaux, etc.).
- Mono-objectif ou multi-objectifs (plusieurs fonctions objectifs à optimiser).
- Avec ou sans contraintes

Les problèmes d'optimisation sont actuellement des axes de recherche importants. En effet, ce genre de problèmes intervient dans plusieurs domaines d'activité très divers, comme la recherche opérationnelle, le traitement d'images, l'économie, l'électronique, la conception mécanique, les réseaux informatiques, etc.

3.4 Les méthodes d'optimisation

Les méthodes d'optimisation peuvent être classées de différentes manières : mono-objectif/multi-objectifs, locales/globales, statiques ou dynamiques. Nous avons choisi de les classer en méthodes déterministes et non-déterministes.

3.4.1 Les méthodes déterministes (exactes)

Il existe plusieurs méthodes déterministes permettant de résoudre certains problèmes d'optimisation et d'obtenir des solutions optimales (exactes) en un temps de calcul raisonnable. Ces méthodes sont généralement adoptées quand la fonction objectif du problème possède un certain nombre de propriétés telles que la convexité, la continuité ou la dérivabilité [63].

Parmi les méthodes déterministes les plus connues, nous pouvons citer : la méthode du gradient [64], la méthode de Newton [65] ou encore la méthode du simplex [66].

3.4.2 Les méthodes non-déterministes (métaheuristiques)

Les méthodes déterministes ne sont pas suffisantes pour résoudre certains problèmes d'optimisation complexes. Parmi ces limites, nous pouvons citer l'absence de convexité, la non-dérivabilité, la discontinuité ou encore la difficulté de définir précisément la fonction objectif [63]. De plus, les méthodes déterministes peuvent avoir un temps de résolution trop long, le problème d'optimisation dans ce cas, est dit NP-difficile (NP-hard).

Pour résoudre de tels problèmes, il est nécessaire d'avoir recours à des méthodes approchées (heuristiques⁸), appelées aussi méthodes stochastiques, qui utilisent des stratégies et des astuces, afin de fournir des résultats approchés en un temps raisonnable sans garantie de l'optimalité [67].

3.4.2.1 Principe et propriétés

Plusieurs métaheuristiques⁹ étaient proposées dans la littérature, allant de la simple recherche locale (la Recherche Tabou [68]) à des algorithmes complexes de recherche globale. La plupart de ces algorithmes sont initialisés aléatoirement et manipulent itérativement une ou plusieurs solutions à la recherche de l'optimum (maximum/minimum) global. Les itérations successives permettent d'améliorer les solutions obtenues. Ces algorithmes s'arrêtent lorsqu'un critère d'arrêt est atteint (ex : nombre maximal d'itérations)

Blum et Roli [69] décrivent des propriétés relatives aux métaheuristiques :

- Les métaheuristiques sont des stratégies qui guident le processus de recherche.
- Le but, est d'explorer efficacement l'espace de recherche afin de trouver des solutions presque optimales.
- Les algorithmes métaheuristiques vont de procédures simples de recherche locale à des procédures complexes.
- Les métaheuristiques sont approximatives et généralement non déterministes.
- Ils utilisent des mécanismes pour éviter de stagner dans des optima locaux.
- Les métaheuristiques ne sont pas spécifiques à un problème donné.

3.4.2.2 Classification des métaheuristiques

Nous présentons brièvement les différentes façons pour la classification des métaheuristiques [69] :

8. Le terme heuristique dérive du grec ancien *heuriskêin* et signifie " trouver "

9. Le terme métaheuristique a été inventé par Fred Glover [68] en 1986

- **Bio-inspirée / non Bio-inspirée** : la classification ici, est basée sur l'origine ou la source de l'algorithme.
- **Population de solution/ solution unique** : le critère de classification est lié au nombre de solutions manipulées au même temps. Les méthodes travaillant sur une seule solution sont dites "trajectoires", la méthode Tabou [68], le Recuit simulé [70] sont des exemples typiques de méthodes trajectoires, contrairement à celles basées sur l'évolution d'un ensemble de points (solutions) dans l'espace de recherche. L'exemple le plus connu de ces méthodes est l'algorithme génétique [71].
- **Fonction objectif statique/dynamique** : c'est la façon d'utiliser la fonction objectif.

La plupart des algorithmes métaheuristiques sont inspirés de plusieurs processus naturels tels que la biologie (algorithmes génétiques), la physique (recuit simulé), ou l'éthologie¹⁰ (colonies d'abeilles [72], optimisation par essaim de particules [73], etc.), d'où le terme bio-inspiré est souvent employé.

3.5 Les algorithmes bio-inspirés

La recherche dans les algorithmes bio-inspirés, consiste à étudier et imiter les différents phénomènes, fonctions, et modèles des êtres vivants pour développer de nouvelles solutions dans plusieurs domaines d'application (industrie, informatique, médecine, etc.). Cette source d'inspiration, nous permettra de traiter des données imprécises ou incomplètes pour accéder à un espace plus large de solutions [74].

Dans la suite de ce chapitre, nous allons présenter les algorithmes bio-inspirés les plus connus et leur application dans le contexte de la segmentation d'images.

3.5.1 Les automates cellulaires (CA)

La méthode d'automate cellulaire (*Cellular Automaton*) présentée par J. Von Neumann [75], est une méthode inspirée des modèles physiques (réseau de spin, croissance des cristaux, etc.) et imitant d'une certaine manière les capacités autoreproductrices des êtres vivants. Dans cette méthode, un automate est une grille régulière de "cellules" contenant chacune un "état" choisi parmi un ensemble fini et qui peut évoluer au cours du temps. Ces cellules interagissent localement de manière synchrone dépendante de son propre état actuel et des états de cellules voisines.

10. L'éthologie est l'étude du comportement des diverses espèces animales.

- **Application à la segmentation d'images**

Les automates cellulaires ont été appliqués dans nombreux problèmes d'optimisation (modélisation physique et biologique, trafic routier, etc.). Cependant, on note leur application restreinte dans le contexte de la segmentation d'images. Dans [76], un nouvel algorithme utilisant les AC pour la segmentation des images est présenté. Les résultats de test sur un ensemble d'images IRM et TDM cérébrales, montrent l'efficacité de l'approche proposée.

3.5.2 Recuit simulé (SA)

Le recuit simulé (*Simulated Annealing*) développé par Kirkpatrick *et al.* [70] est une technique d'optimisation basée sur les perturbations aléatoires des solutions candidates et une décision probabiliste pour retenir les solutions mutées. L'origine de la méthode provient de la métallurgie, où pour atteindre les états de basse énergie d'un métal, on le chauffe jusqu'à des températures élevées, avant de le laisser refroidir lentement. Ce processus est appelé le recuit.

Le recuit simulé repose sur l'algorithme de Metropolis [77] qui décrit l'évolution d'un système thermodynamique. Cette procédure permet d'éviter des minima locaux avec une probabilité élevée si la température T est élevée et de conserver les états les plus probables quand l'algorithme atteint de très basses températures [70].

Algorithme 3.1 : Recuit simulé

```

1 début
2   Choisir une température initiale  $T = T_0$  et une solution de départ  $S = S_0$ ;
3   répéter
4     répéter
5       Générer une solution dans le voisinage de la solution actuelle;
6       Comparer les deux solutions selon la règle de Metropolis;
7     jusqu'à équilibre atteint;
8     Décroître la température;
9   jusqu'à condition d'arrêt;
10 fin

```

Dans le recuit simulé, les paramètres d'une solution sont initialisés aléatoirement. Si la fitness (énergie) de cette solution est inférieure à la fitness précédente, la nouvelle solution

remplace l'ancienne. Sinon, elle sera remplacée par l'ancienne solution avec une probabilité proportionnelle à la différence d'énergie et la température actuelle.

L'algorithme SA a été appliqué dans nombreux problèmes d'optimisation, du fait de sa facilité d'implémentation. En revanche, l'inconvénient de cet algorithme est qu'il possède un nombre élevé de paramètres (température initiale, règle de décroissance de la température, etc.) rendant le choix de ces derniers assez empirique [78]. L'autre inconvénient majeur de cette méthode est qu'elle est gourmande en temps de calcul. Mais grâce au développement des outils informatiques et de la puissance de calcul des ordinateurs, la convergence du recuit simulé est assurée en un temps de calcul raisonnable.

- **Application à la segmentation d'images**

Dans [79], un algorithme de recuit simulé basé sur l'algorithme Fuzzy c-means est développé pour la segmentation des images médicales. Les résultats expérimentaux sur plus de 180 IRM et TDM montrent l'efficacité de l'approche développée.

3.5.3 Les algorithmes génétiques (GA)

Une sous-classe des algorithmes évolutionnaires, les algorithmes génétiques (*Genetic Algorithms*) sont des méthodes inspirées des mécanismes biologiques tels que les lois de Mendel et le principe fondamental de sélection. Holland dans [71] a introduit les principes de ces algorithmes pour permettre aux ordinateurs de rechercher des solutions aux problèmes.

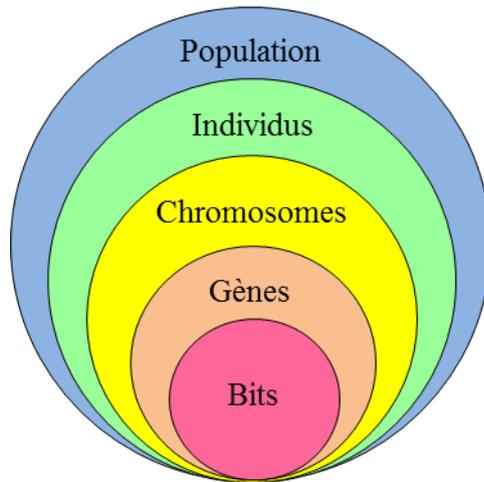


Figure 3.2: Composition d'une population de l'algorithme génétique.

Dans les GA, une population est formée d'un ensemble d'individus, qui sont évolués à chaque itération (appelée génération), jusqu'à ce qu'un critère d'arrêt soit vérifié. Les solutions du problème sont comparées à des chromosomes, qui sont constituées d'une suite

de gènes, chaque gène étant associé à une série de bits [80]. L'évaluation des individus est obtenue par une fonction de performance (fitness) que l'algorithme génétique cherche à maximiser ou minimiser. Pour chaque génération, la population est soumise à des opérateurs de sélection, croisement et de mutation.

3.5.3.1 Les opérateurs génétiques

- **Sélection** : il y a deux étapes de sélection : l'une pour déterminer quels seront les éléments qui vont se reproduire entre eux. Et une seconde étape pour déterminer les meilleurs individus (meilleurs fitness) qui vont constituer la prochaine génération.
- **Croisement** : Cet opérateur permet de générer des nouveaux individus héritant des caractéristiques de leurs parents. On distingue les croisements suivants : à un point, à deux points et croisement uniforme.
- **Mutation** : Ce sont des petites modifications aléatoires des gènes des enfants. Les gènes d'un individu sont inversés (un 0 devient un 1, et vice versa). La mutation permet d'explorer l'espace de recherche tout en échappant de se stagner dans des minima locaux [81].

3.5.3.2 Principe général de l'algorithme

L'algorithme génétique (Algorithme 3.2) commence au début, par choisir les individus autorisés à se reproduire, à partir d'une population de N individus. Ensuite, les individus choisis se croisent, de façon à obtenir une population d'enfants, tout en mutant aléatoirement certains gènes. La performance des individus est évaluée, grâce à la fonction fitness, qui permet de sélectionner les individus autorisés à survivre. Le processus se répète jusqu'à un critère d'arrêt, par exemple un certain nombre d'itérations est atteint ou aucune amélioration de la performance des individus.

3.5.3.3 Avantages et inconvénients

Les algorithmes génétiques présentent plusieurs solutions potentielles, ce qui leur permet d'explorer plusieurs zones dans l'espace de recherche. Ces algorithmes sont simples à mettre en œuvre, parallèles et robustes [82]. Cependant, comme toute approche d'optimisation, les algorithmes génétiques présentent des limites. Tout d'abord, ils sont coûteux en temps de calcul, puisqu'ils travaillent sur plusieurs solutions simultanément. Ainsi, l'ajustement des paramètres d'un algorithme génétique est délicat. En outre, ces algorithmes n'offrent aucune garantie quant à la qualité des solutions obtenues [82]. Un autre problème lié à ces algorithmes, est celui de la dérive génétique, où un

Algorithme 3.2 : Algorithme Génétique

```
1 début
2   Initialisation aléatoire de la population :  $N$  individus ;
3   Evaluer les  $N$  individus ;
4   répéter
5     Croisement des  $M$  individus sélectionnés;
6     Mutation des individus;
7     Sélectionner les individus pour le remplacement;
8   jusqu'à condition d'arrêt;
9 fin
```

individu favorisé par le hasard, en l'espace de quelques générations, peut envahir toute la population. On parle dans ce cas de convergence prématurée [80], ce qui mène l'algorithme à se stagner dans des optima locaux. Un autre problème surgit lorsque plusieurs individus se mettent à avoir des performances similaires et l'algorithme ne progresse plus [80].

- **Application à la segmentation d'images**

Les algorithmes génétiques étaient largement appliqués pour résoudre le problème de la segmentation des images. Bhanu *et al.* [83] ont utilisé les GAs pour adapter les paramètres d'un algorithme de segmentation basée région. Afin d'améliorer les performances de l'algorithme FCM, Aliyari *et al.* [84] ont appliqué un GA pour trouver les valeurs optimales de deux facteurs, qui sont la différence entre les pixels voisins et leur emplacement dans l'image. Leur approche été testée sur un ensemble d'images IRM bruitées.

3.5.4 Algorithme de colonies de fourmis (ACO)

Cet algorithme a été mis au point par Dorigo [85] au début des années 90. Son principe est inspiré du comportement des fourmis naturelles qui parcourent leur environnement à la recherche de nourriture, tout en traçant le plus court chemin entre le nid et la source de nourriture (Figure 3.3). Chaque fourmi dépose une substance chimique odorante appelée phéromone à l'attention de ses congénères. La communication indirecte entre les fourmis est appelée stigmergie¹¹.

On remarque sur la Figure 3.3 que si un obstacle est introduit sur le chemin reliant

11. Stigmergie : communication via l'environnement

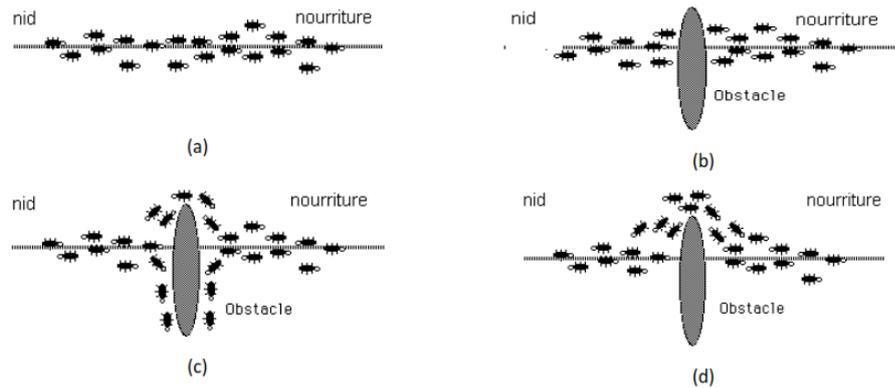


Figure 3.3: Détermination du plus court chemin par une colonie de fourmis [46] : (a) Situation initiale, (b) Introduction d'un obstacle, (c) Recherche du chemin optimal, (d) Prédominance du chemin optimal.

le nid et la source de nourriture (Figure 3.3(b)), les premières fourmis qui arrivent à l'intersection des deux chemins vont choisir le chemin à prendre avec une équiprobabilité. Les fourmis en parcourant les deux chemins, vont déposer de la phéromone qui s'évapore avec le temps. La quantité de phéromone déposée sur le chemin court va devenir plus grande que celle déposée sur le chemin long, car les fourmis parcourent le premier chemin plus vite. Après un certain moment, les fourmis vont toutes emprunter le plus court chemin (forte concentration de phéromone) et le deuxième chemin sera abandonné (Figure 3.3(d)).

Bien que conçu initialement pour résoudre le problème du voyageur de commerce, l'algorithme de colonies de fourmis (*Ant Colony Optimization*) a été adapté à plusieurs problèmes d'optimisation. Il présente plusieurs avantages, tels que le parallélisme, la robustesse et la décentralisation [78]. Le majeur inconvénient de l'algorithme ACO est le problème de paramétrage.

- **Application à la segmentation d'images**

Dans [86], Wang *et al.* ont développé un nouvel algorithme de segmentation d'images basé sur les contours actifs utilisant l'algorithme ACO pour obtenir un contour précis et continu (meilleur chemin) dans une région. En outre, une nouvelle règle de mise à jour de phéromone est introduite pour améliorer le temps d'exécution. Dans [87], Xumin Liu *et al.* ont amélioré l'algorithme ACO conventionnel pour la détection des contours dans une image. L'algorithme utilisait plusieurs règles de transfert et une stratégie de mise à jour de phéromone. Les expériences montrent l'efficacité, la rapidité et la précision de cette approche.

3.5.5 Algorithme à évolution différentielle (DE)

Une autre sous-classe des algorithmes évolutionnaires, l'algorithme de l'évolution différentielle (*Differential Evolution*) a été développé par Price et Storn [88] pour résoudre les problèmes d'optimisation à variables continues. Cet algorithme utilise les mêmes opérations que l'algorithme génétique (sélection, croisement, mutation) pour chercher des solutions optimales. La principale différence entre les deux algorithmes est que dans les algorithmes génétiques, l'opération de mutation est le résultat de petites modifications dans les gènes des individus d'une population, par contre la stratégie de l'algorithme DE, consiste à biaiser un opérateur de mutation pour un individu, en fonction des différences calculées entre d'autres individus sélectionnés aléatoirement [89], donc, la mutation dans l'algorithme DE n'est pas basée sur une fonction de probabilité.

L'algorithme DE a été largement appliqué dans plusieurs domaines d'optimisation, vu sa convergence rapide et sa robustesse, néanmoins, le contrôle des paramètres reste le problème majeur de cet algorithme.

- **Application à la segmentation d'images**

Omran *et al.* [90] ont développé une approche de classification basée sur l'algorithme DE. L'objectif de cette approche, est de minimiser la distance intra-clusters et maximiser la distance inter-clusters pour déterminer les meilleurs centres de classes. L'approche proposée a été appliquée avec succès sur un ensemble d'images IRM synthétiques. Zhao *et al.* [91] ont proposé une méthode hybridant l'algorithme de seuillage OTSU avec un algorithme DE pour segmenter des images bruitées. Dans [92], Soham Sarkar *et al.* ont modifié une fitness basée sur un algorithme d'évolution différentielle adaptatif pour la classification des pixels dans une image.

3.5.6 Optimisation par essaim de particules (PSO)

L'optimisation par essaim de particules (*Particle Swarm Optimization*) est un algorithme stochastique développé en 1995 grâce à la collaboration de Russell Eberhart (ingénieur en électricité) et James Kennedy (socio-psychologue) [73]. Cet algorithme est inspiré du comportement social des animaux vivant en essaim tels que les oiseaux migrateurs et les bancs de poissons, en se basant sur un modèle développé par Craig Reynolds [93] qui simule le comportement d'une nuée d'oiseaux en vol.

Dans la littérature, l'algorithme PSO a été abondamment appliqué pour résoudre

plusieurs problèmes d'optimisation. Une étude détaillée de l'algorithme PSO sera présentée dans le chapitre suivant.

3.5.7 Systèmes immunitaires artificiels (AIS)

Proposé par D. Dasgupta [94], l'algorithme AIS (*Artificial Immune Systems*) est une modélisation du système immunitaire naturel qui inclut : l'algorithme de la sélection négative, l'algorithme de la sélection clonale, les réseaux immunitaires, et la théorie de danger. L'algorithme est composé d'agents appelés leucocytes responsables de la défense de l'organisme contre les agents pathogènes (virus, bactéries, parasites). Ces algorithmes exploitent les caractéristiques du système immunitaire naturel liées à l'apprentissage et à la mémorisation. La sélection négative est une abstraction des mécanismes qui permettent aux systèmes immunitaires naturels de distinguer entre le soi (ses propres molécules) et le non soi (molécules étrangères). La sélection clonale est une abstraction des mécanismes de mémorisation des systèmes immunitaires [95].

Les AIS, possèdent un grand nombre d'avantages comme l'adaptabilité, le traitement distribué/parallèle, la mémorisation et la robustesse. Les AIS ont été utilisés dans divers domaines comme la classification, la reconnaissance des formes, l'extraction de connaissance, et la sécurité des réseaux contre les intrusions. Le principal inconvénient d'utilisation des AIS est la difficulté de trouver une correspondance entre le problème réel et sa modélisation.

- **Application à la segmentation d'images**

Dans [96] Erik Cuevas *et al.* ont introduit un algorithme de multi-seuillage basé sur les algorithmes immunitaires artificiels. Dans leur approche, les auteurs ont utilisé des fonctions de mélanges de gaussiennes pour calculer les différents seuils à partir d'un histogramme. Les paramètres de ces fonctions sont calculés par l'algorithme de la sélection clonale. Les expériences sur deux benchmarks d'images montrent la capacité de l'algorithme à déterminer automatiquement la valeur du seuil tout en préservant les principales caractéristiques de l'image originale. Valarmathy *et al.* [97] ont utilisé un AIS pour la sélection des caractéristiques afin d'identifier la démence (maladie neurodégénérative) sur des images IRM cérébrales.

3.5.8 Algorithme de recherche d'harmonie (HS)

L'Algorithme de recherche d'harmonie (*Harmony Search*) est une métaheuristique développée par Geem *et al.* [98]. Cette méthode est basée sur le processus de performance musical qui consiste à trouver une meilleure harmonie dans un orchestre musical où chaque musicien joue une note. D'une manière analogue, chaque musicien correspond à un attribut dans une solution candidate dans l'espace de recherche, et la hauteur et la portée de chaque instrument correspond aux limites et aux contraintes de la variable de décision. L'harmonie entre les musiciens est considérée comme une solution potentielle, et l'appréciation du public représente la fonction objectif (fitness) du problème. Les musiciens cherchent l'harmonie au fil du temps par de petites variations et improvisations, ce qui se traduit par une amélioration de la fitness.

En raison de son efficacité, sa rapidité, et qu'il nécessite peu de paramètres de configuration [98], l'algorithme RH a été appliqué avec succès dans plusieurs problèmes d'optimisation. Cependant, cet algorithme souffre de la faible capacité en recherche locale.

- **Application à la segmentation d'images**

Dans le contexte de la segmentation des images IRM cérébrales, et pour améliorer les performances de FCM, plusieurs contributions utilisant l'algorithme HS ont été présentées dans la littérature, elles seront présentées en détail dans la Section 6.2.

3.5.9 Algorithme d'essaims de poissons artificiels (AFSA)

Proposé par X. Li et J. Qian [99], l'algorithme d'essaim de poissons artificiels (*Artificial Fish-Swarm Algorithm*) est inspiré du comportement collectif et social des poissons naturels. L'algorithme fonctionne en imitant trois comportements du poisson naturel : le comportement pour rechercher les meilleurs endroits de nourriture (solutions), comportement grégaire pour améliorer sa chance de survie dans l'essaim en raison de protection mutuelle contre les ennemies, et le comportement de suivre le poisson ayant trouvé une source de nourriture.

Vu sa convergence rapide et sa flexibilité, l'algorithme AFSA, a été bien adapté pour résoudre des problèmes d'optimisation [100]. Son inconvénient majeur, réside dans la difficulté de trouver un compromis entre une recherche locale et globale [100].

- **Application à la segmentation d'images**

Un algorithme AFSA est présenté dans [101], les auteurs utilisaient les comportements

des poissons pour résoudre un problème de segmentation multi-seuillage. D'autres hybridations de l'algorithme AFSA avec le FCM sont présentées dans la littérature (Section 6.2), pour choisir une bonne initialisation des centres de classes.

3.5.10 Optimisation par recherche bactérienne (BFO)

L'optimisation par recherche bactérienne (*Bacterial Foraging Optimization*) est un algorithme d'optimisation stochastique proposé en 2002 par Passino [102]. L'algorithme est inspiré du comportement d'un certain type de bactérie (*Escherichia coli*) dans sa recherche des zones les plus nutritives. L'évolution d'un groupe de bactéries comprend quatre phases : 1) *chemotaxis*, qui est la phase d'avancement ou de culbute des bactéries vers les régions les plus prometteuses. 2) *regroupement* : la bactérie qui trouve la meilleure zone prometteuse, informe les autres bactéries pour se regrouper ensemble. 3) *reproduction* : des bactéries les plus adaptées (meilleurs fitness). 4) *élimination ou dispersion* : de tous les individus dans une zone mauvaise, et de nouveaux échantillons aléatoires sont insérés avec une faible probabilité [103].

- **Application à la segmentation d'images**

Un algorithme de multi-seuillage basé sur une modification de l'algorithme BFO est introduit dans [103]. Les résultats sur un ensemble de quatorze benchmark, montrent que l'algorithme proposé MBF dépasse d'autres techniques évolutionnistes en terme de qualité, stabilité et temps de calcul. Dans [104], M.Hongwei et Y.Yujing ont utilisé l'algorithme BFO pour améliorer la recherche globale de FCM par l'optimisation de sa fonction objectif.

3.5.11 Algorithme de colonie d'abeilles (ABC)

L'algorithme de colonies d'abeilles (*Artificial Bee Colony*), proposé par Karaboga et Basturk [72], simule le comportement de butinage des abeilles naturelles. L'idée de base de cette méthode consiste à créer un système d'abeilles artificielles qui collaborent les unes avec les autres afin de résoudre des problèmes complexes d'optimisation.

Dans cet algorithme, la position d'une source de nourriture représente une solution potentielle au problème d'optimisation et la quantité de nectar correspond à la qualité (fitness) de la solution associée. Au début, un essaim d'abeilles est généré et commence à se déplacer de façon aléatoire dans l'espace de recherche. La modification de la position d'une abeille (solution) dépend de sa position actuelle (information visuelle) et de la quantité de nectar (fitness) dans cette position. Si le nectar de cette source est supérieur

ou égal à celui de la source précédente, celle-ci est remplacée par la nouvelle. Dans le cas contraire l'ancienne source est conservée. A la fin du processus de recherche, les abeilles partagent les informations sur les sources de nourriture ainsi que leurs localisations avec les autres abeilles. Ces dernières évaluent ces informations, et choisissent les sources de nourriture en fonction d'une probabilité associée à cette source [105]. Si la position de la source de nourriture ne peut pas être améliorée durant un nombre de cycles prédéterminé appelé (limite), alors cette source sera abandonnée et remplacée par une autre source aléatoire. Le choix du paramètre limite est très important pour le bon déroulement de l'algorithme [72].

L'avantage de l'algorithme de colonie d'abeilles ABC est qu'il nécessite peu de paramètres, ce qui rend sa mise en œuvre très simple, aussi, le compromis exploitation-exploration est bien respecté. Néanmoins, son inconvénient est lié à la convergence prématurée, et sa complexité assez élevée, où plusieurs travaux d'amélioration ont été proposés dans la littérature.

- **Application à la segmentation d'images**

S. Ouadfel et S.Meshoul [106] ont décrit un algorithme de segmentation floue FCM basé sur une modification de l'algorithme de clonie d'abeilles (MABC) dans laquelle une nouvelle stratégie de mutation inspirée de l'algorithme de l'évolution différentielle est introduite afin d'améliorer le processus d'exploitation.

3.5.12 Algorithme de Recherche gravitationnelle (GSA)

L'algorithme de Recherche gravitationnelle (*Gravitational Search Algorithm*) est introduit par E. Rashedi *et al.* [107] pour résoudre les problèmes d'optimisation. Son principe est basé sur la loi de gravitation de Newton. L'algorithme est constitué d'un ensemble de particules de différentes masses qui interagissent les unes avec les autres par la force de gravité. Chaque particule est considérée comme un objet et sa performance est mesurée par sa masse. La force de gravité provoque un mouvement global où tous les objets se déplacent vers d'autres objets avec des masses plus lourdes, donc, vers des solutions optimales. En GSA, une particule dispose de quatre paramètres qui sont la position, la masse inertielle, la masse gravitationnelle active, et la masse gravitationnelle passive. La position de la masse représente la solution du problème, où les masses d'inertie et de gravitation sont déterminées en utilisant une fonction objectif (fitness).

L'avantage de l'algorithme GSA est qu'il nécessite que deux paramètres à ajuster,

qui sont la masse et la vitesse des particules pour trouver des solutions globalement optimales. Mais cet algorithme souffre du problème de la convergence prématurée et du temps de calcul élevé [108].

- **Application à la segmentation d'images**

Dans [109] E. Rashedi and H. Nezamabadi ont présenté un algorithme de recherche gravitationnelle pour la segmentation d'images par croissance de région où chaque particule de GSA se déplace dans l'espace de caractéristiques pour trouver des régions homogènes. G. Sun and A. Zhang [110] ont développé une approche (GA-GSA) qui combine un algorithme génétique avec l'algorithme GSA pour un multi-seuillage d'histogramme.

3.5.13 Algorithme Firefly (FA)

C'est une méta-heuristique inspirée du comportement des lucioles (en anglais *firefly*). Elle a été développée par Xin-She Yang [111]. Les lucioles sont de petits coléoptères ailés capables de produire une lumière clignotante pour une attraction mutuelle. Les femelles peuvent imiter les signaux lumineux d'autres espèces afin d'attirer les mâles qu'elles capturent et dévorent. Chaque luciole est attirée par les lucioles les plus brillantes qu'elle. L'attractivité est proportionnelle à l'intensité lumineuse, qui décroît avec la distance [108].

L'algorithme FA est constitué d'une population d'agents (lucioles) qui communiquent entre eux via une bioluminescence qui leur permet d'explorer l'espace de recherche. Ainsi, chaque lumière produite par un agent, attire ses partenaires (quel que soit leur sexe), ce qui mène à une exploration de l'espace de recherche. L'intensité lumineuse de luciole est déterminée par la valeur de la fitness.

L'algorithme FA a montré son efficacité dans plusieurs domaines d'optimisation multiobjectif. Son inconvénient, est lié au choix de la taille de la population initiale, qui doit être grande afin d'augmenter la diversité dans l'espace de recherche. Donc, la qualité des solutions est bonne mais le temps de calcul est élevé [108].

- **Application à la segmentation d'images**

Dans le domaine de la segmentation d'images, T. Hassanzadeh *et al.* [112] ont développé une approche de segmentation fondée sur l'optimisation de l'algorithme de seuillage Otsu par l'algorithme FA. Dans [113], P. Jitpakdee *et al.* ont récemment proposé de combiner l'algorithme FCM avec l'approche FA pour la segmentation des images. Dans cette approche, le FCM est utilisé pour guider le déplacement des lucioles. Les résultats

expérimentaux sur 7 images de test, ont montré l'efficacité de cette approche hybride comparée à K-means et à FCM standard.

3.5.14 Algorithme de Chauve-souris (BA)

Développé par Xin-She Yang [114], l'algorithme des chauves-souris (Bat Algorithm) est une méthode d'optimisation qui modélise le comportement d'écholocation¹² des microchiroptères (sorte de chauves-souris). Elle repose essentiellement sur le principe suivant : pour trouver de la nourriture, les chauves-souris volent aléatoirement sans aucune connaissance sur l'endroit de leurs proies, cependant elles utilisent l'écholocation pour calculer la distance qui sépare chacune d'entre elles de la cible. Leur objectif commun est non seulement d'atteindre leurs proies mais aussi de trouver la meilleure stratégie de chasse.

Formellement, chaque chauve-souris possède les propriétés suivantes : une position, une vitesse, une fréquence, une intensité et un taux d'émission [114]. La position de la chauve-souris est considérée comme une solution dans l'espace de recherche. Pour chaque itération, l'algorithme calcule la fitness de chaque chauve-souris, met à jour les meilleures solutions globales et locales et enfin met à jour les propriétés de chaque chauve-souris jusqu'à atteindre un certain nombre d'itérations.

- **Application à la segmentation d'images**

Un algorithme de chauves-souris modifié a été appliqué dans [115] pour résoudre le problème de segmentation par multi-seuillage. Les résultats sur un ensemble d'images de test, montrent l'efficacité de l'algorithme par rapport à cinq algorithmes standards. Dans [116], les auteurs ont utilisé l'algorithme de chauve-souris dans le processus de détection et de séparation des structures cérébrales et tumorales dans des images IRM.

3.5.15 Autres algorithmes bio-inspirés

Les algorithmes suivants, ont été appliqués dans d'autres domaines d'optimisation.

- Algorithme Intelligent water drops [117].
- Algorithme Cuckoo search [118].
- Algorithme Dolphin echolocation [119].
- Paddy field [120].

12. "Moyen de localisation des obstacles ou des proies, utilisé par divers animaux vivant dans l'obscurité (chauves-souris) ou dans l'eau (cétacés), et consistant à émettre des ultrasons ou des sons aigus et à apprécier le temps de retour de leur écho dans les diverses directions". Larousse.

- Monkey search [121].
- Krill herd [122].
- Lion Optimization [123].
- Penguin Search Optimization [124].

3.6 Conclusion

Nous avons présenté dans ce chapitre, une synthèse des différents algorithmes bio-inspirés, et leur exploitation en segmentation d'images. Ces algorithmes ont rencontré un vif succès grâce à leur simplicité, leur adaptabilité et leur capacité de trouver des solutions presque optimales.

Dans le prochain chapitre, nous allons étudier en détail l'algorithme d'optimisation par essaim de particules (PSO), algorithme que nous avons choisi pour notre problème d'optimisation.

Chapitre 4

Algorithme d'optimisation par essaim de particules

Sommaire

4.1	Introduction	51
4.2	Inspiration	52
4.2.1	Modèle de Reynolds	52
4.2.2	Métaphore	52
4.2.3	L'intelligence collective	53
4.3	Principe général	53
4.4	Les topologies de voisinage	54
4.4.1	Voisinage géographique	54
4.4.2	Voisinage social	55
4.5	Algorithme PSO	56
4.5.1	Formalisation	56
4.5.2	Paramètres de l'algorithme	57
4.6	Avantages et inconvénients de PSO	59
4.7	Conclusion	60

4.1 Introduction

L'optimisation par essaim de particules (PSO) est un algorithme qui utilise une population de solutions candidates pour chercher une solution optimale au problème.

Il s'inspire à l'origine du comportement social des animaux évoluant en essaim, tels que les oiseaux ou les bancs de poissons [73].

Dans la suite de ce chapitre, nous allons détailler l'algorithme PSO, algorithme que nous avons choisi pour notre problème de segmentation d'images IRM cérébrales. Nous présentons également les différents paramètres contrôlant cette métaheuristique. Enfin, nous exposons ses inconvénients et ses avantages.

4.2 Inspiration

4.2.1 Modèle de Reynolds

L'idée principale de l'algorithme PSO, s'appuie sur un modèle développé par Craig Reynolds [93] en 1986, simulant le comportement d'une nuée d'oiseaux en vol dans laquelle les agents (appelés *Boids*) obéissent à trois règles locales pour maintenir la stabilité de l'essaim (Figure 4.1) :

- **Séparation** : s'éloigner des voisins s'ils sont trop proches.
- **Cohésion** : se maintenir dans l'essaim en se déplaçant vers le centre des voisins.
- **Alignement** : se déplacer dans la même direction que l'essaim en moyennant les vitesses et les directions des voisins.

Ces trois règles de base, permettent l'attraction et la répulsion de chacun des individus et maintenir la cohésion de l'essaim.

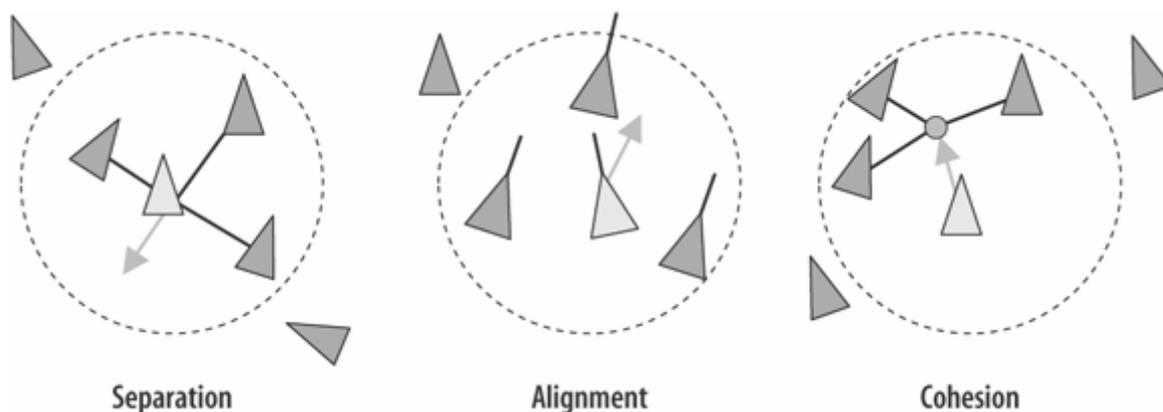


Figure 4.1: Les trois règles de Reynolds [125].

4.2.2 Métaphore

La métaphore dans l'algorithme PSO est celle d'un essaim d'oiseaux, particulièrement du fait qu'un oiseau qui possède une certaine capacité de mémorisation et de traitement,

lorsqu'il trouve un site prometteur (source de nourriture), il informe certains de ses voisins qui vont tenir compte de cette information pour leur prochain déplacement . En effet, individuellement chaque oiseau a une intelligence limitée et une connaissance seulement locale. L'intelligence globale de l'essaim est donc le résultat direct des interactions locales entre les différents oiseaux de l'essaim. La performance de l'essaim entier est supérieure à la somme des performances de ses oiseaux [78].

Dans la suite de ce manuscrit, chaque agent (individu) de l'essaim sera baptisé particule.

4.2.3 L'intelligence collective

C'est l'intelligence provenant des interactions multiples entre les particules simples dans un essaim, où :

- Chaque particule ne possède qu'une connaissance partielle de l'environnement global.
- Les particules ont un comportement très simple, mais elles peuvent accomplir des tâches très complexes grâce à un mécanisme stigmergique, où il n'y a aucun contrôle centralisé sur l'essaim.
- La coordination de l'essaim se fait implicitement à travers des règles individuellement simples, en se basant sur plusieurs concepts d'intelligence, comme la coordination et l'auto-organisation. Dans [126], K. Kordon a utilisé le terme *main invisible* pour expliquer cette coordination.

4.3 Principe général

Dans un essaim, les particules qui sont des solutions potentielles au problème d'optimisation, survolent dans l'espace de recherche pour trouver l'optimum global. Au départ de l'algorithme, les particules sont réparties au hasard dans l'espace de recherche, chaque particule ayant une vitesse aléatoire et également une petite mémoire, lui permettant de mémoriser sa meilleure position, et elle dispose d'un groupe d'informatrices, appelé voisinage [81]. Ensuite, à chaque pas de temps, chaque particule est capable de [127] :

- Evaluer la qualité de sa position (fitness) qui représente la distance à l'optimum global et retenir en mémoire sa meilleure performance, c'est-à-dire la meilleure position atteinte jusqu'ici.

- Interroger ses particules voisines et d’obtenir de chacune d’entre elles sa propre meilleure performance.
- Choisir la meilleure des meilleures performances dont elle a connaissance.

Le déplacement d’une particule est influencé par les trois composantes suivantes [63] :

1. Une composante d’inertie : la particule tend à suivre sa propre direction de déplacement.
2. Une composante cognitive : la particule tend à se diriger vers la meilleure position par laquelle elle est déjà passée.
3. Une composante sociale : la particule tend à se diriger vers le meilleur site déjà visité par ses voisins.

La Figure 4.2 illustre le principe de déplacement d’une particule.

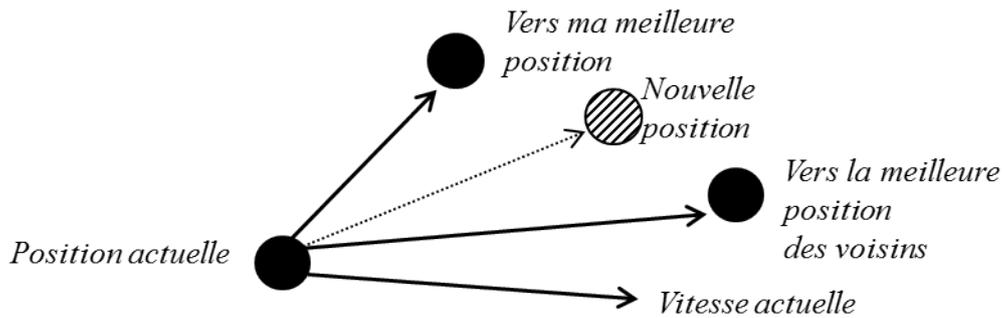


Figure 4.2: Principe de déplacement d’une particule.

4.4 Les topologies de voisinage

En PSO, la topologie constitue le réseau de communication entre les particules au sein de l’essaim. Elle permet de définir un groupe d’informatrices pour chaque particule, c’est ce que l’on appelle le voisinage. Différents voisinages ont été étudiés [128]. On distingue deux types :

4.4.1 Voisinage géographique

Dans cette topologie, le voisinage d’une particule i se compose des particules proches. Pour calculer cette proximité, les distances euclidiennes dans l’espace de recherche entre la particule i et les autres particules sont utilisées. A noter que le nombre de voisins doit être défini *a priori*.

Cette topologie est coûteuse en de temps de calcul, mais elle est plus proche de la réalité dynamique de communication dans un essaim, puisque les liens de communication entre les particules sont modifiés à chaque pas de temps.

4.4.2 Voisinage social

Le voisinage social d'une particule est l'ensemble des particules informatrices fixé au début de l'algorithme. Il permet d'améliorer la rapidité de propagation de l'information dans un essaim, en d'autres termes, améliorer la convergence de l'algorithme. Le voisinage social est plus simple à programmer et il est moins coûteux en temps calcul.

Dans le PSO original, deux modèles ont été définis :

- Modèle *gbest* (topologie globale) : dans ce modèle chaque particule est informée par la totalité des particules. Ainsi, la position de la meilleure particule (globale) est utilisée dans l'équation de la mise à jour des vitesses. Dans cette topologie, PSO converge rapidement, car toutes les particules sont attirées simultanément à la meilleure solution. Cependant, l'inconvénient majeur de cette version est qu'elle ne donne pas lieu à une exploration suffisante si l'optimum global n'est pas proche de la meilleure particule, donc, l'algorithme peut se stagner dans un optimum local [63].
- Modèle *lbest* (topologie locale) : dans ce modèle, chaque particule n'a que deux particules voisines. L'essaim convergera lentement, mais peut localiser l'optimum global avec une plus grande probabilité, ce qui réduit les risques d'une convergence prématurée [129].

D'autres modèles sociaux ont été proposés dans la littérature, afin d'améliorer les performances de l'algorithme PSO [128], ils peuvent être caractérisés en fonction de deux facteurs, le degré de connectivité K qui représente le nombre de voisins d'une particule, et le nombre de voisins d'une particule appartenant également à d'autres voisinages. Parmi ces modèles on peut citer : l'étoile, Von Neumann, graphe aléatoire, arbre, etc.

Le choix du voisinage a un effet majeur sur la propagation de la meilleure solution trouvée par l'essaim. Certaines topologies de voisinage sont meilleures pour une recherche locale (exploitation) et d'autres pour une recherche globale (exploration) [130].

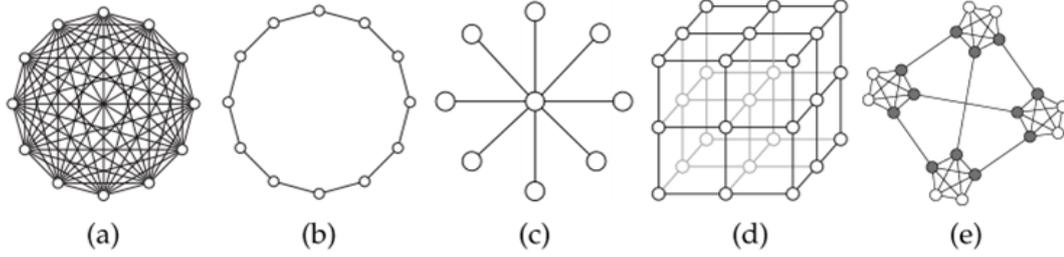


Figure 4.3: Topologies statiques : (a) Modèle gbest, (b) Modèle lbest, (c) Etoile, (d) Von Neumann, (e) Four-clusters [63].

4.5 Algorithmes PSO

On va présenter l'algorithme PSO pour un modèle *gbest*, où toutes les particules de l'essaim sont entièrement connectées.

Soit f la fonction objectif d'un problème d'optimisation (minimisation). Soit P le nombre de particules dans l'essaim. La performance de chaque particule i (distance à l'optimum global) est mesurée en utilisant la fonction f .

4.5.1 Formalisation

Dans un espace de recherche de dimension D , chaque particule i est représentée par les caractéristiques suivantes :

- vecteur position $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$
- vecteur vitesse ¹³ $\vec{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$
- la meilleure position déjà visitée par i , $\overrightarrow{Pbest}_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{iD})$ dans le temps $t + 1$ est calculée comme suit :

$$\overrightarrow{Pbest}_i(t+1) = \begin{cases} \overrightarrow{Pbest}_i(t), & \text{si } f(\vec{x}_i(t+1)) \geq f(\overrightarrow{Pbest}_i(t)) \\ \vec{x}_i(t+1), & \text{sinon} \end{cases} \quad (4.1)$$

- La meilleure position du voisinage (tout l'essaim) de la particule i , $\overrightarrow{Gbest}_i = (Gbest_{i1}, Gbest_{i2}, \dots, Gbest_{iD})$ obtenue comme suit :

$$\overrightarrow{Gbest}_i(t+1) = \arg \min_{\overrightarrow{Pbest}_i} f(\overrightarrow{Pbest}_i(t+1)), 1 \leq i \leq P \quad (4.2)$$

Le vecteur vitesse et le vecteur position sont mis à jour à partir des équations (Eq. 4.3) et (Eq. 4.4), respectivement :

13. On trouve aussi dans la littérature le terme *vélocité*

$$v_{i,j}^{t+1} = wv_{i,j}^t + c_1r_{1,i,j}^t [Pbest_{i,j}^t - x_{i,j}^t] + c_2r_{2,i,j}^t [gbest_j^t - x_{i,j}^t], j \in \{1, 2, \dots, D\} \quad (4.3)$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1}, j \in \{1, 2, \dots, D\} \quad (4.4)$$

où $v_{i,j}^t$ représente la vitesse de la particule i dans la dimension j à l'instant t , r_1 et r_2 sont des nombres aléatoires compris dans l'intervalle $[0, 1]$, w est une constante, appelée coefficient d'inertie et c_1 et c_2 sont des constantes, appelées coefficients d'accélération.

Le déplacement de la particule est influencé par les trois composantes citées ci-dessus :

1. la composante d'inertie $wv_{(i,j)}^t$ introduite par Shi and Eberhart [131], commande l'impact de la vitesse précédente (tendance à suivre la même direction) sur le déplacement futur, une grande valeur de w favorise l'exploration, tandis qu'une petite valeur favorise l'exploitation.
2. la composante cognitive $c_1r_{1,i,j}^t [Pbest_{i,j}^t - x_{i,j}^t]$ qui correspond à l'attraction linéaire vers la meilleure position déjà trouvée par la particule i .
3. la composante sociale $c_2r_{2,i,j}^t [gbest_j^t - x_{i,j}^t]$ qui correspond à l'attraction linéaire vers la meilleure position trouvée par le voisinage de la particule.

Algorithme 4.1 : PSO

```

1  début
2  | Initialiser aléatoirement  $P$  particules : positions et vitesses ;
3  | tant que critère d'arrêt n'est pas satisfait faire
4  | | Pour  $i \leftarrow 1$  à  $P$  faire
5  | | | Evaluer la fitness  $f(\vec{x}_i)$ ;
6  | | | Mettre à jour  $\overrightarrow{Pbest}_i$  et  $\overrightarrow{Gbest}_i$  selon (Eq. 4.1) et (Eq. 4.2);
7  | | | Déplacer les particules selon (Eq. 4.3) et (Eq. 4.4);
8  | | | Limiter les vitesses dans l'intervalle  $[-v_{max}, +v_{max}]$ 
9  | | finpour
10 | fin
11 fin

```

4.5.2 Paramètres de l'algorithme

4.5.2.1 Nombre de particules P

Le nombre de particules allouées à la résolution d'un problème d'optimisation, dépend essentiellement de la taille de l'espace de recherche. Il n'y a pas de règle pour déterminer

ce paramètre, faire de nombreux essais permet d'avoir la bonne valeur.

En général, l'augmentation du nombre de particules augmente la diversité, ce qui réduit la possibilité d'être piégé dans des minima locaux [130]. Selon Shi and Eberhart [131], la valeur de P , ne doit pas être trop petite.

4.5.2.2 Etape d'initialisation

Les positions des particules \vec{X} sont initialisées aléatoirement, mais d'une façon à couvrir tout l'espace de recherche. A noter que si les positions des particules ne sont pas initialisées dans tout l'espace de recherche, l'algorithme PSO trouve des difficultés à localiser l'optimum, s'il se trouve en dehors de la zone initiale.

4.5.2.3 Facteur d'inertie w

Le facteur d'inertie w , introduit par Shi et Eberhart [131], contrôle l'influence de la direction actuelle de la particule sur le déplacement futur. Ce paramètre a été introduit pour réaliser un compromis entre la recherche locale (exploitation) et la recherche globale (exploration). Une grande valeur du facteur d'inertie provoque une grande exploration de l'espace de recherche alors qu'une petite valeur concentre la recherche localement sur un petit espace [63].

Les études menées dans [132] indiquent une meilleure convergence pour $w \in [0.8, 1.2]$. Au-delà de 1.2, l'algorithme tend à avoir certaines difficultés à converger. Les auteurs ont proposé aussi un facteur d'inertie dynamique qui diminue au cours du temps, il commence par une valeur de 0.9 et décroît linéairement pour arriver à 0.4 selon la formule suivante :

$$w(t) = \frac{iter_{\max} - t}{iter_{\max}} (w_{\max} - w_{\min}) + w_{\min} \quad (4.5)$$

où t est l'itération courante et $iter_{\max}$ est le nombre maximal d'itérations, w_{\max} et w_{\min} désignent respectivement les valeurs maximales et minimales du facteur w .

A noter que la stratégie d'utiliser un w dynamique a beaucoup amélioré les performances de PSO pour plusieurs problèmes d'optimisation [63].

4.5.2.4 Confinement des particules

Lors du déplacement des particules, il peut arriver qu'une particule sorte de l'espace de recherche. Pour éviter cette "explosion", une technique appelée confinement (*velocity clamping*) [133] est utilisée pour limiter les vitesses dans l'intervalle $[-v_{\max}, +v_{\max}]$ et permet de ramener une particule sortie de l'espace de recherche à l'intérieur de celui-ci et de réaliser ainsi un compromis efficace entre l'exploitation et l'exploration.

La valeur de v_{max} est généralement dépendante de l'échelle du problème. Dans le cas d'un problème de segmentation des images, v_{max} prend la valeur 255, qui représente le plus grand niveau de gris.

4.5.2.5 Coefficients c_1, c_2

La combinaison des paramètres w, c_1 et c_2 permet de régler la balance entre une recherche locale et globale. Dans [134] les auteurs ont choisi les valeurs expérimentales suivantes : $w = 0.7298, c_1 = c_2 = 1.496$. Toutefois, d'autres valeurs ont été utilisées dans la littérature.

4.5.2.6 Critères d'arrêt

Plusieurs critères d'arrêt peuvent être envisagés pour terminer l'algorithme PSO :

- **Nombre maximum d'itération** : si le nombre maximum d'itérations est petit l'algorithme PSO génère des solutions qui ne sont pas optimales.
- **Convergence vers la solution recherchée** : souvent on utilise un seuil ε , si la valeur de ε est trop grande l'algorithme termine avec de mauvaises solutions, et si elle est trop petite, la recherche se termine lentement.
- **Aucune amélioration / nombre d'itérations** : il y a plusieurs façons de mesurer l'amélioration de la recherche, par exemple la variation de la vitesse est proche de 0 ou la variation des positions des particules est trop petite.

4.6 Avantages et inconvénients de PSO

L'algorithme PSO est similaire aux algorithmes évolutionnaires, dans le sens où les deux approches sont basées-population et pour chaque itération, les individus se déplacent dans l'espace de recherche afin d'améliorer leur fonction objectif.

En AG, le mécanisme de sélection permet de soutenir la survie du plus fort, le chromosome possédant une meilleure fonction objectif étant toujours présent dans la prochaine génération. C'est un concept central dans tous les algorithmes évolutionnaires [135]. Cependant, dans PSO il n'y a plus de mécanisme de sélection, une particule actuellement médiocre peut devenir la meilleure au futur. En d'autres termes, les particules sont en coopération à la recherche de l'optimum global plutôt qu'en compétition. Donc, l'algorithme PSO est influencé par le comportement social plutôt que la survie du plus fort [130]. Une autre différence importante est que dans PSO, chaque individu profite de son historique alors que ce mécanisme n'existe pas dans les algorithmes évolutionnaires.

Grâce à sa rapidité et sa robustesse, l'algorithme PSO a été appliqué dans une vaste gamme de problèmes d'optimisation (traitement d'images, industriel, etc.). Il présente l'avantage d'être simple à implémenter et il nécessite peu de paramètres à ajuster. On note son adaptation aux traitements parallèles et simultanés [108]. Cependant, PSO présente certains inconvénients qui sont les suivants : le problème de la convergence prématurée qui conduit à un optimum local ; le problème de l'influence du choix des paramètres sur la performance de l'algorithme [130].

Dans la littérature, plusieurs travaux ont été proposés pour remédier à ces problèmes et améliorer les performances de PSO. La plupart de ces travaux sont basés sur l'hybridation avec d'autres méthodes. Pour une revue plus complète de ces travaux, le lecteur peut se référer à [136].

4.7 Conclusion

Dans ce chapitre, nous avons présenté d'une manière détaillée, l'algorithme d'optimisation par essaim de particules PSO. Cet algorithme stochastique qui s'inspire à l'origine du comportement social des animaux évoluant en essaim, a connu un franc succès dans le domaine d'optimisation, vu sa simplicité et son adaptabilité pour résoudre des problèmes d'optimisation continue. Dans la littérature, plusieurs méthodes de segmentation d'images (contour, région, classification) ont été hybridées avec l'algorithme PSO pour tirer profit de ses avantages.

Dans le prochain chapitre, nous allons exposer notre première contribution qui se porte sur l'amélioration du temps de calcul de l'algorithme FCM pour un problème de segmentation d'images IRM.

Chapitre 5

FCM rapide pour la segmentation d'images IRM

Sommaire

5.1	Introduction	61
5.2	Hybridation d'algorithmes	62
5.2.1	Définition	62
5.2.2	Objectifs de l'hybridation	62
5.2.3	Types d'hybridation	62
5.3	Contribution	64
5.3.1	Motivation	64
5.3.2	Description	64
5.4	Validation et résultats	67
5.5	Discussion	69
5.6	Conclusion	73

5.1 Introduction

Ce chapitre est consacré au développement d'une méthode de segmentation rapide, basée sur l'algorithme FCM. Le principe consiste à estimer la configuration initiale des centres de classes à partir des centres obtenus par FCM standard, sur un ensemble d'images IRM possédant des propriétés similaires à l'image à segmenter. L'objectif visé est de minimiser le temps d'exécution de la segmentation.

Par la suite, nous décrivons l'hybridation des algorithmes de segmentation. Puis, nous présentons la nouvelle approche de segmentation nommée MFCM (*Modified FCM*). Ensuite, l'étude comparative de notre approche avec l'algorithme de base FCM sur un ensemble d'images IRM est exposée. Nous terminons ce chapitre par une discussion traitant les résultats obtenus et les observations.

5.2 Hybridation d'algorithmes

5.2.1 Définition

L'hybridation des algorithmes est un ensemble de méthodes associées, mettant en œuvre la combinaison de deux ou plusieurs algorithmes. Elle peut être définie formellement par un ensemble d'algorithmes et un ensemble de relations entre ces algorithmes [137].

5.2.2 Objectifs de l'hybridation

L'hybridation joue un rôle important dans l'amélioration de la capacité de recherche des algorithmes. Elle vise à combiner et exploiter la complémentarité qui existe entre les différents algorithmes, pour fournir un résultat plus complet et plus précis que celui fourni par les algorithmes pris indépendamment [138]. L'hybridation favorise une approche modulaire et répartie d'un système. En général, son résultat peut faire des améliorations en termes de vitesse de calcul ou de précision [139].

5.2.3 Types d'hybridation

L'étude menée par Germond [138] sur les méthodes coopératives (hybrides) pour la segmentation d'images IRM cérébrales, a abouti à la définition de trois grands types d'hybridation :

5.2.3.1 Hybridation par initialisation

Cette hybridation exploite le résultat d'une méthode pour initialiser une seconde méthode (Figure 5.1). L'image originale est segmentée par la méthode 1, en générant des informations (solutions) qui vont être exploitées par une autre méthode et ainsi de suite.

5.2.3.2 Hybridation par fusion

Elle permet de tirer parti de la fusion d'informations complémentaires dans le but d'améliorer le processus de segmentation. L'image originale est segmentée par plusieurs

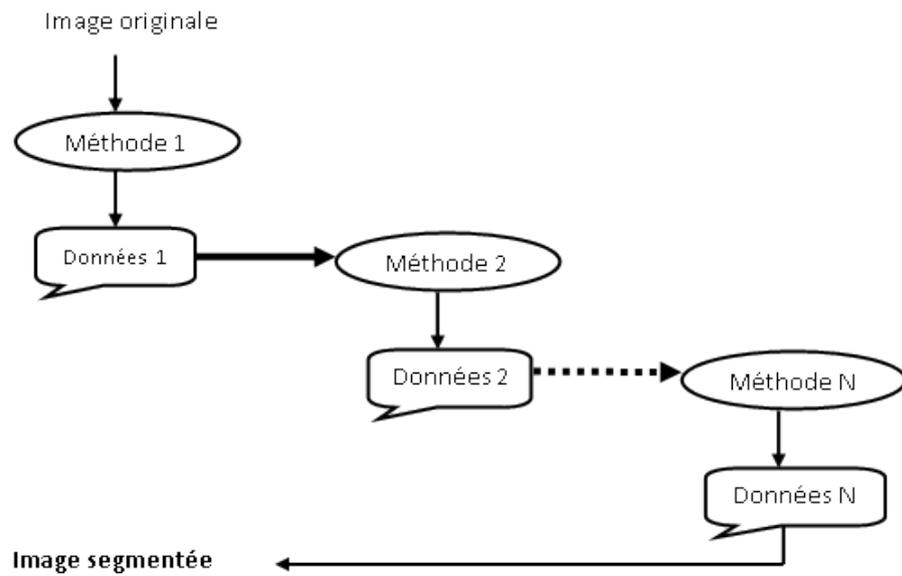


Figure 5.1: Hybridation par initialisation (séquentielle).

méthodes : *méthode1*, *méthode2*, etc. Chaque méthode va générer des données qui vont être fusionnées pour reconstruire l'image finale segmentée.

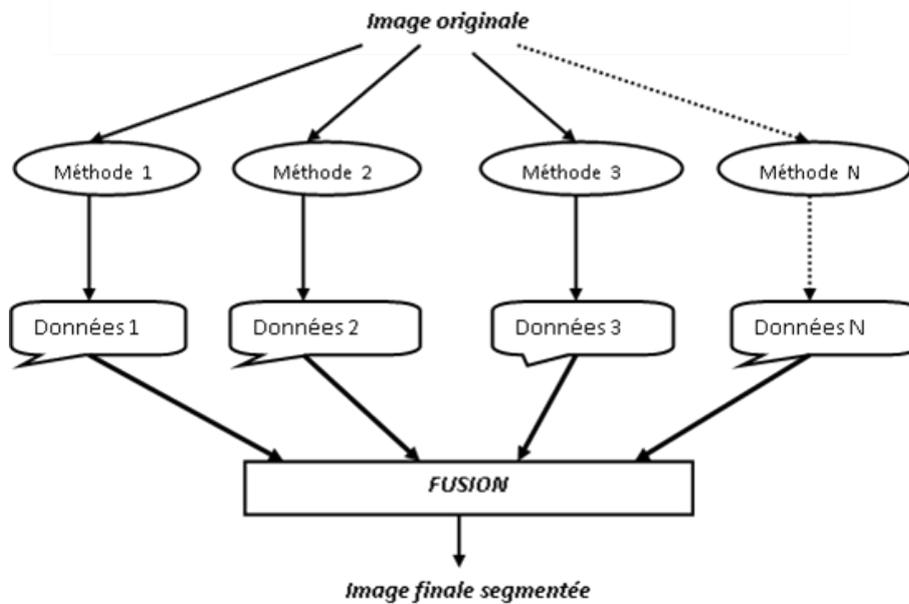


Figure 5.2: Hybridation par fusion (parallèle).

5.2.3.3 Hybridation par rétroaction

L'hybridation par rétroaction permet de revenir sur le processus de segmentation pour affiner les résultats obtenus.

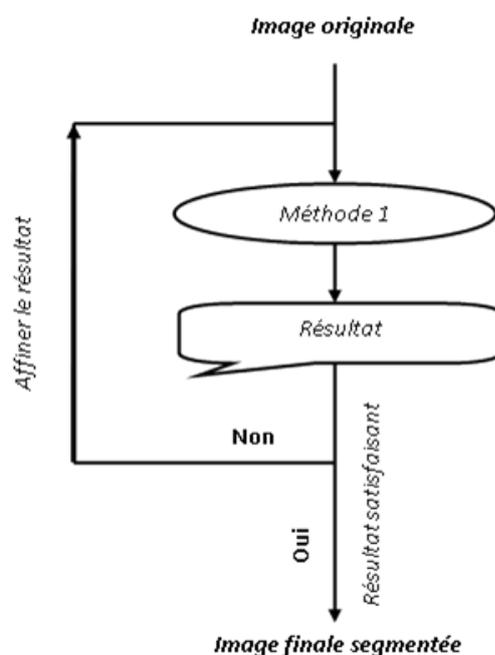


Figure 5.3: Hybridation par rétroaction.

5.3 Contribution

5.3.1 Motivation

Lors d'un examen IRM, le changement des valeurs des différents paramètres de séquence : temps d'écho, temps de répétition et l'angle de basculement, permet de générer une grande diversité de volumes variant par le contraste pour le même sujet, où chaque séquence IRM (T1, T2 ou DP) contient des informations spécifiques, et le choix entre les différentes séquences dépend de la structure à étudier. Dans ce contexte, nous avons pensé à exploiter cette diversité pour accélérer le processus de la segmentation.

L'idée principale vient du fait qu'une coupe IRM pondérée en T1 englobe la même structure anatomique que celle en T2 ou DP, la différence réside seulement au niveau du contraste. Ainsi, dans un volume IRM les coupes sont successives et elles sont proches anatomiquement les unes des autres. Donc, au lieu d'initialiser aléatoirement les centroïdes de FCM pour des images pondérées en T2 ou DP, nous allons intégrer *a priori* les centres de classes relatives aux images pondérées en T1 correspondantes.

5.3.2 Description

Parmi les méthodes de segmentation d'images par classification, l'algorithme FCM a été largement étudié dans la littérature vu l'intérêt qu'il présente à manipuler des

données floues et imprécises. Néanmoins, ses performances sont liées aux valeurs initiales des centres de classes [51].

Dans le but de réduire le temps d'exécution de FCM, nous avons proposé une nouvelle approche [54, 140] d'initialisation des centres de classes, qui se déroule en deux étapes :

1. Construction de la matrice des centres

Cette étape consiste premièrement à appliquer le FCM classique sur un volume IRM de X coupes d'une modalité donnée (exemple T1), et au fur et à mesure que l'algorithme FCM converge, les centres de C classes de chaque coupe seront stockés dans une matrice $Z(X, C)$. Puis la matrice Z sera divisée en Nb groupes, où chaque groupe contient un nombre équitable de coupes successives. A la fin, la matrice $M(Nb, C)$ est calculée, cette matrice contient les valeurs moyennes des C centres de chaque groupe (Figure 5.4). L'avantage de cette étape, est qu'elle permet de calculer une fois pour toutes la matrice $M(Nb, C)$ qui sera utilisée pour segmenter n'importe quelle image de modalité différente (T2, DP, etc).

2. Etape de segmentation

Dans cette étape (Algorithme 5.1), les centres de classes obtenus dans l'étape 1, seront utilisés comme information *a priori* pour initialiser le FCM.

L'approche proposée MFCM [54, 140] permet d'optimiser les positions initiales des centres de classes, contrairement à FCM où ces positions sont estimées aléatoirement. En fait, les centres calculés dans l'étape 1 sont presque proches des centres réels. Cela permet à MFCM de minimiser la fonction objectif dans le plus bref délai.

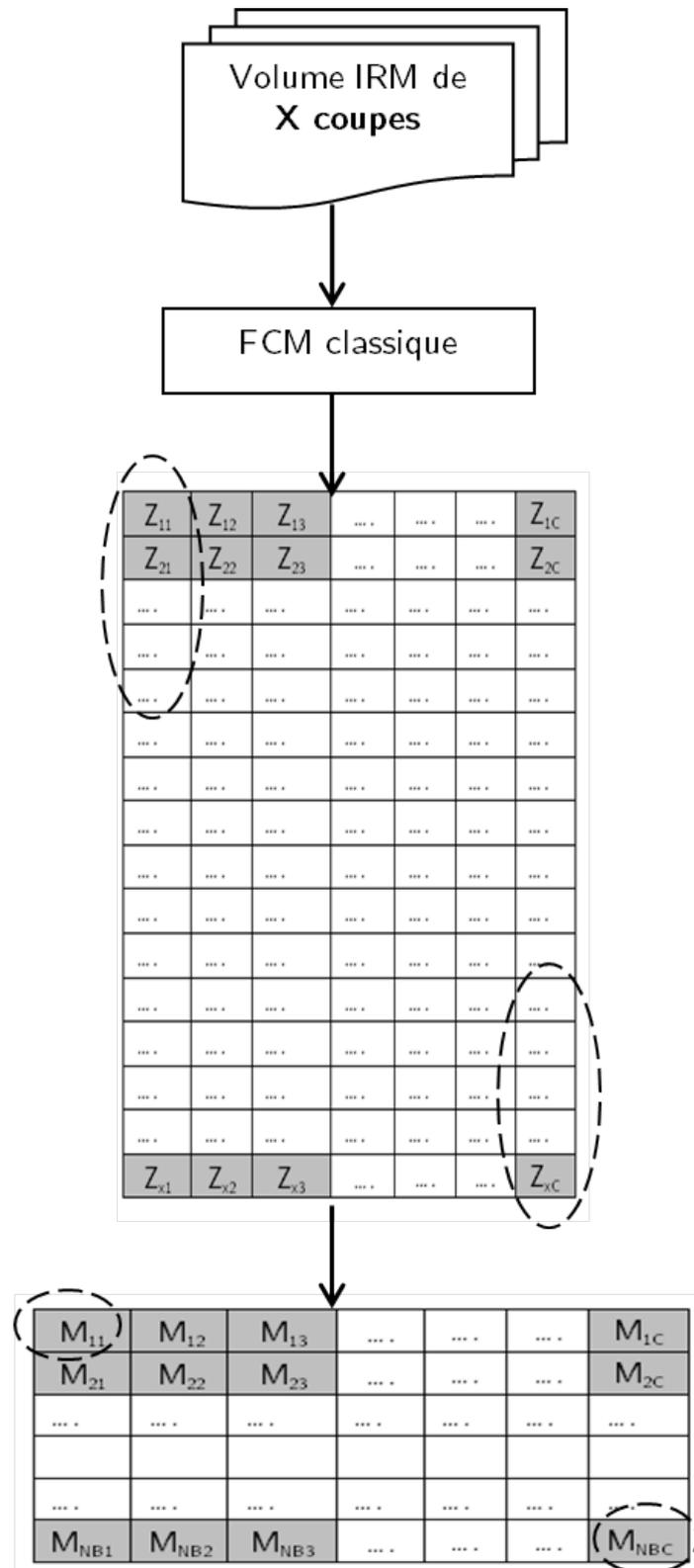


Figure 5.4: Construction de la matrice des centres.

Algorithme 5.1 : MFCM

Données : image IRM originale, Matrice des centres $M(Nb,C)$ **Résultat** : image segmentée

```
1 début
2   Fixer le critère d'arrêt  $\varepsilon$ , le degré flou  $m$ ,  $itermax$  ;
3   Récupérer le numéro du groupe (NG) de l'image originale;
4   Initialiser le vecteur des centroïdes par  $M(NG,...)$  ;
5   Pour  $i \leftarrow 1$  à  $itermax$  faire
6     Mettre à jour la matrice  $U$  par (Eq. 2.6);
7     Mettre à jour le vecteur des centroïdes  $Z$  (Eq. 2.7);
8     Calculer la fonction objectif  $J$  (Eq. 2.8);
9     Si  $|J^{(t)} - J^{(t+1)}| < \varepsilon$  alors
10      | break;
11    finsi
12  finpour
13  Défuzzification de la matrice  $U$ ;
14 fin
```

5.4 Validation et résultats

Pour évaluer l'approche proposée MFCM, nous avons utilisé 27 volumes IRM générés à l'aide du simulateur Brainweb [141]. Chaque volume constitué de l'empilement de 181 coupes de 217x181 pixels, affectés de trois niveaux de bruit Gaussien (0% 3% 5%) et trois niveaux d'inhomogénéité RF (0% 20% 40%).

Les algorithmes FCM standard et MFCM ont été implémentés sous l'environnement Matlab version 7, et exécutés dans un PC Pentium Dual-core E5700 / 3.00 GHz / 2.00 GB RAM. Les paramètres choisis : $C = 4$, $\varepsilon = 10^{-5}$, $itermax = 100$, $m = 2$.

Pour estimer la matrice des centres de classes M (Section 5.3.2), nous avons divisé le volume IRM en Nb groupes, où chaque groupe contient 10 coupes successives.

Les temps de calcul moyens (en secondes) après 50 exécutions indépendantes de FCM et MFCM sont présentés dans les Tableaux 5.1 à 5.3. Les meilleurs résultats sont indiqués en gras. L'écart type (entre parenthèses) est calculé pour vérifier la stabilité de l'approche proposée.

Tableau 5.1: Comparaison de temps d'exécution (secondes) de FCM et MFCM sur des images pondérées en T1.

Bruit	0%		3%		5%	
RF	FCM	MFCM	FCM	MFCM	FCM	MFCM
0%	0,4593 (0,0922)	0,1428 (0,0054)	0,4671 (0,1070)	0,2253 (0,0077)	0,5071 (0,1174)	0,2043 (0,0042)
20%	0,4031 (0,0718)	0,1643 (0,0078)	0,3728 (0,0822)	0,2415 (0,0077)	0,5284 (0,1416)	0,3081 (0,0221)
40%	0,5975 (0,3275)	0,2262 (0,0084)	0,5368 (0,0813)	0,2668 (0,0052)	0,6578 (0,1472)	0,3893 (0,0061)

Tableau 5.2: Comparaison de temps d'exécution (secondes) de FCM et MFCM sur des images pondérées en DP.

Bruit	0%		3%		5%	
RF	FCM	MFCM	FCM	MFCM	FCM	MFCM
0%	0,4840 (0,0802)	0,3893 (0,0061)	0,6287 (0,4214)	0,4093 (0,0069)	0,5856 (0,2421)	0,4928 (0,0113)
20%	0,4781 (0,2189)	0,3712 (0,0066)	0,6734 (0,5856)	0,4134 (0,0163)	0,8893 (0,6721)	0,4703 (0,0182)
40%	0,4421 (0,0992)	0,3259 (0,0062)	0,5043 (0,1028)	0,4721 (0,0078)	0,6146 (0,1270)	0,8478 (0,0235)

Tableau 5.3: Comparaison de temps d'exécution (secondes) de FCM et MFCM sur des images pondérées en T2.

Bruit	0%		3%		5%	
RF	FCM	MFCM	FCM	MFCM	FCM	MFCM
0%	0,8168 (0,3023)	0,4543 (0,0098)	0,6712 (0,1583)	0,4118 (0,0097)	0,7737 (0,1267)	0,5140 (0,0064)
20%	0,5928 (0,1500)	0,3662 (0,0083)	0,7315 (0,1384)	0,4931 (0,0202)	0,6800 (0,0975)	0,5290 (0,0069)
40%	0,6725 (0,1207)	0,8575 (0,0175)	0,8262 (0,1498)	0,8228 (0,0080)	0,8425 (0,1627)	0,9053 (0,0224)

Les histogrammes présentés dans Les Figures 5.5 à 5.7, illustrent les temps de calcul moyens, obtenus après 50 exécutions de FCM et MFCM sur une base de 36 images IRM, pour différents niveaux de bruit et d'inhomogénéité RF (0% 0%, 3% 20%, 5% 40%). On note ici que les quatre images de chaque figure représentent les coupes du même groupe.

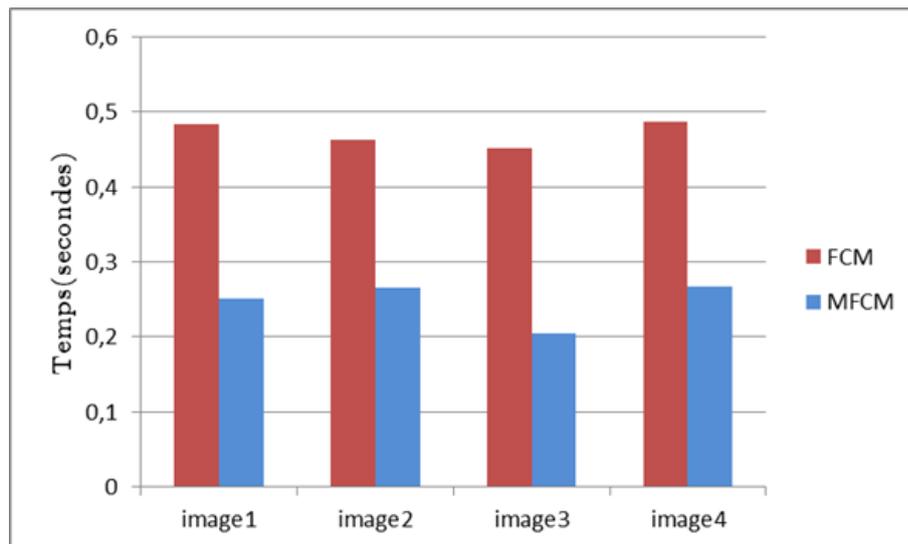
5.5 Discussion

Les Tableaux 5.1 à 5.3 donnent la moyenne et l'écart type des temps d'exécution obtenus par FCM et MFCM sur des images pondérées respectivement en T1,DP et T2. D'après ces tableaux, on peut remarquer que le temps d'exécution moyen de MFCM est considérablement réduit pour la plupart des images de test. Ainsi, les valeurs de l'écart-type sont faibles pour tous les tests, ce qui indique que l'approche MFCM est plus stable. Cependant, lorsque les niveaux de bruit et d'inhomogénéité RF sont élevés, on remarque que le MFCM est moins performant en terme de temps de calcul, ceci est dû aux valeurs aberrantes qui influent considérablement sur l'estimation de la matrice des centres de classes et conduisent à une convergence plus lente. Pour réduire l'influence de ces valeurs aberrantes, nous avons remplacé dans [142] la moyenne des valeurs par la médiane.

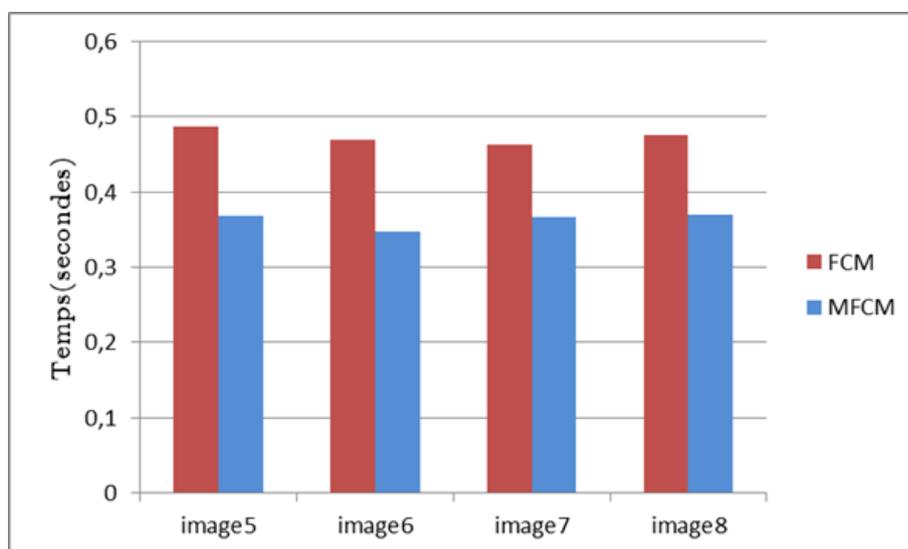
D'après les Figures 5.5 et 5.6, il apparaît clairement que l'approche MFCM permettait de réduire considérablement les temps de calculs pour toutes les coupes. Par ailleurs, comme nous pouvons le constater à la Figure 5.7, l'approche proposée n'a apporté aucun gain de temps. Bien entendu, le calcul est largement influencé par les niveaux élevés de bruit (5%) et d'inhomogénéité RF (40%) qui perturbent considérablement les distributions des niveaux de gris.

L'approche MFCM présente l'avantage d'être peu couteuse en temps de calcul. Elle présente cependant des limites. D'une part, les caractéristiques imprévisibles et variables des images IRM, qui changent d'un imageur à l'autre ou d'un patient à un autre, revient donc à estimer à chaque fois la matrice des centres de classes (étape1). D'autres part, la sensibilité de cette approche aux données aberrantes qui influent largement sur les valeurs moyennes des centres de classes.

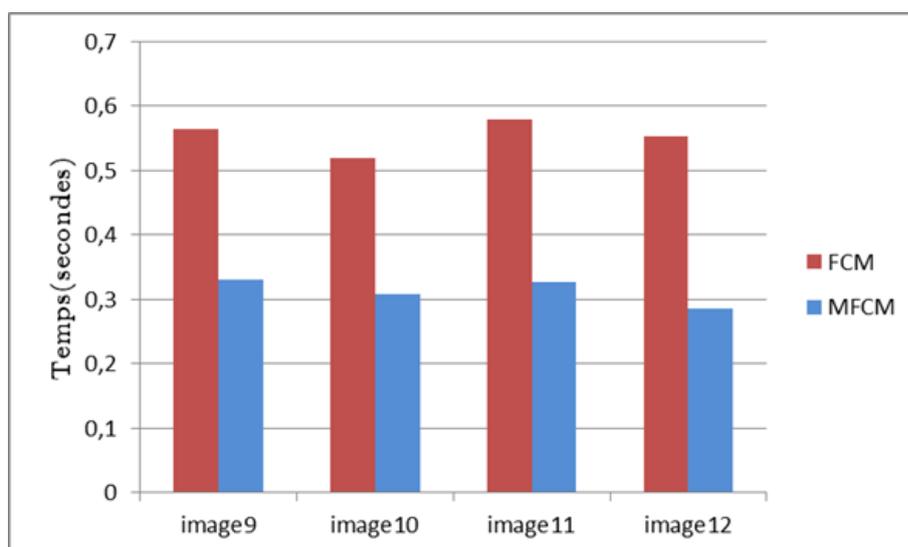
On note enfin, que les résultats obtenus par MFCM et FCM sont similaires en termes de qualité de segmentation sur l'ensemble des images de test.



(a) Séquences T1

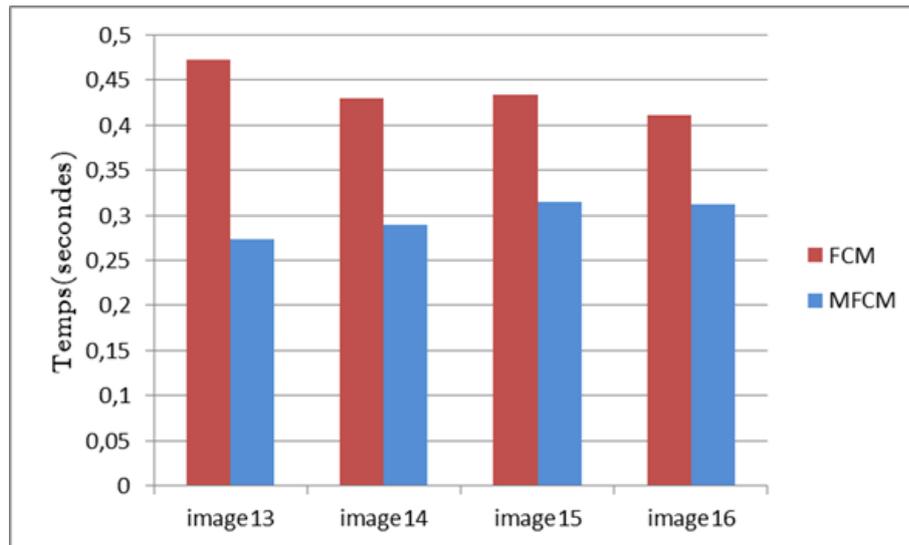


(b) Séquences DP

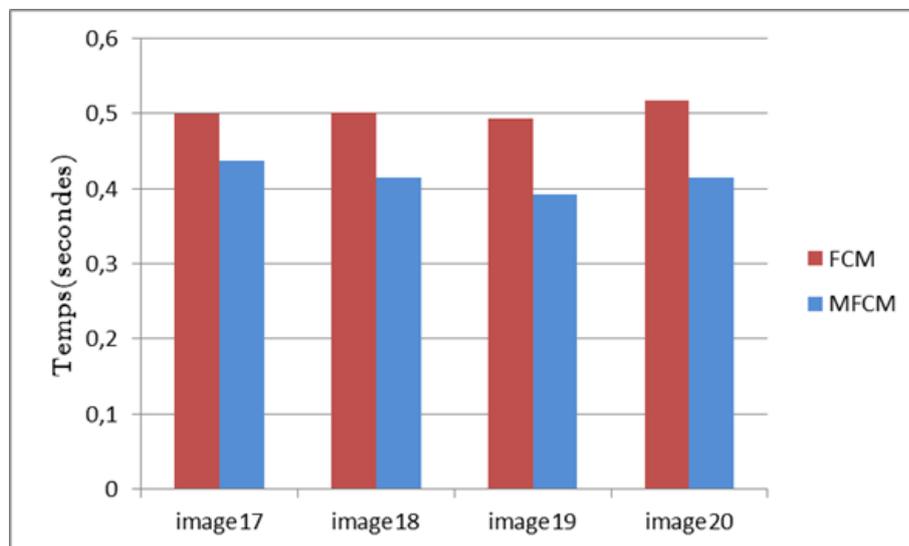


(c) Séquences T2

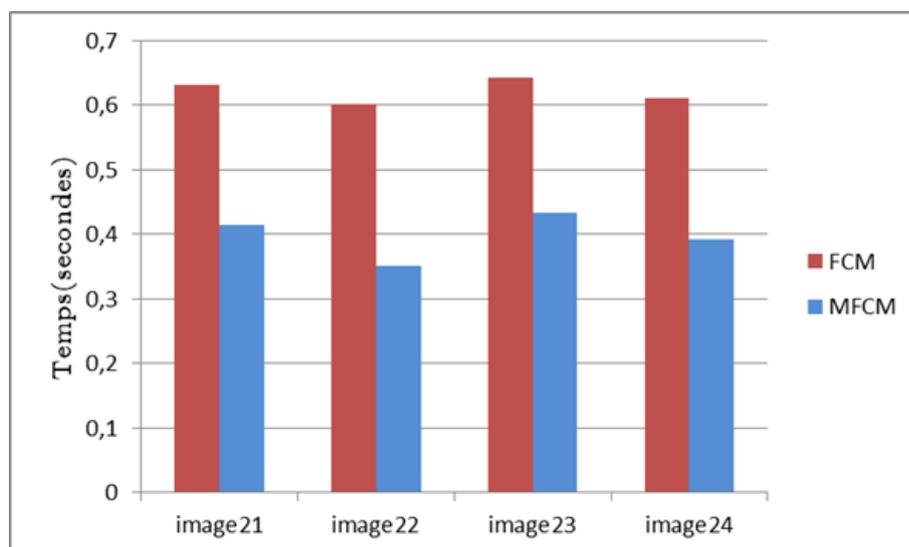
Figure 5.5: Comparaison de temps d'exécution (secondes), bruit=0%, Rf=0%.



(a) Séquences T1

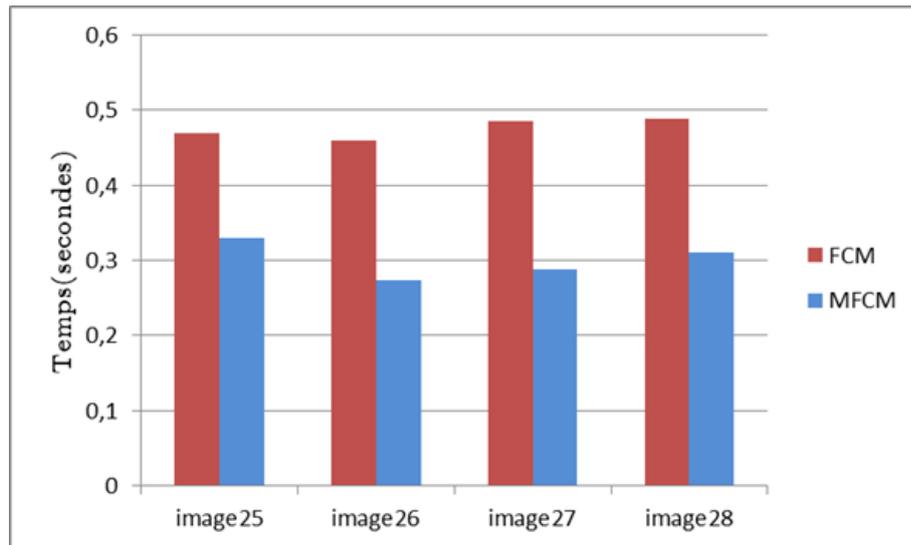


(b) Séquences DP

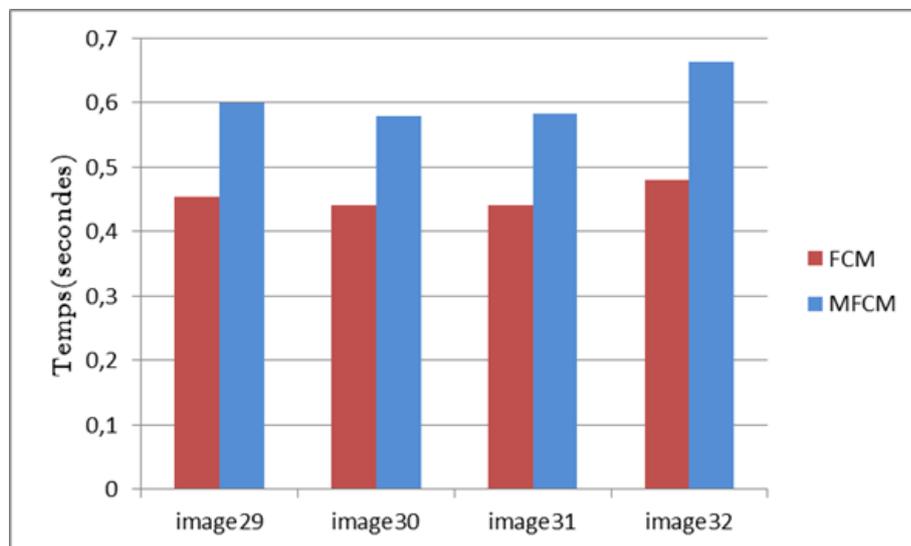


(c) Séquences T2

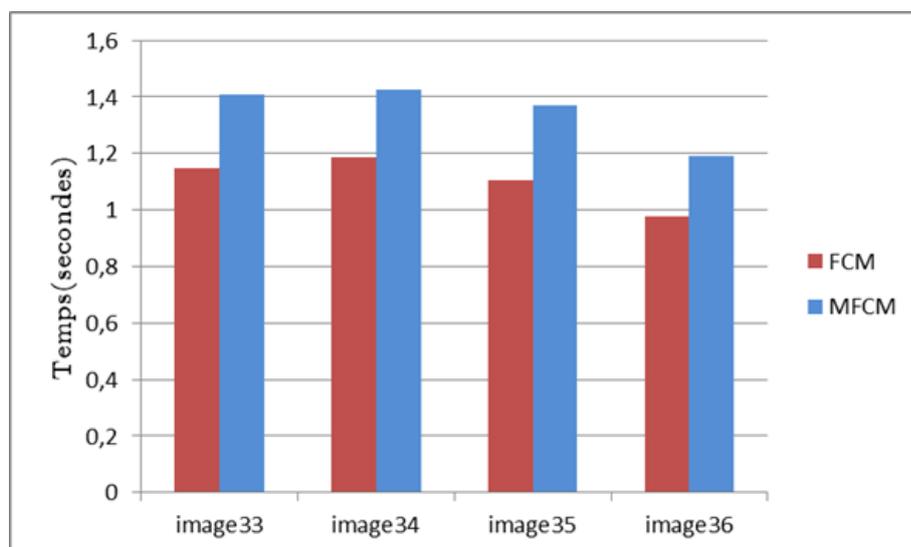
Figure 5.6: Comparaison de temps d'exécution (secondes), bruit=3%, Rf=20%.



(a) Séquences T1



(b) Séquences DP



(c) Séquences T2

Figure 5.7: Comparaison de temps d'exécution (secondes), bruit=5%, Rf=40%.

5.6 Conclusion

Pour réduire le temps d'exécution de la segmentation, nous avons présenté dans ce chapitre une nouvelle approche d'initialisation des centres de classes de FCM. Notre solution est basée sur l'estimation *a priori* de la configuration initiale, à partir des images IRM dont les caractéristiques sont semblables à l'image originale.

Les résultats de l'évaluation sur plusieurs images IRM de différentes pondérations, montrent que notre approche MFCM a réduit considérablement le temps de calcul. Cependant, elle ne permet pas d'avoir de bonnes performances sur des images fortement bruitées.

Dans le prochain chapitre, nous allons introduire les travaux d'hybridation de FCM avec les algorithmes bio-inspirés, appliqués au problème de la segmentation d'images. Ensuite, nous nous intéressons à la présentation de notre deuxième contribution, qui consiste au développement d'une méthode hybride PSO-FCM pour résoudre la problématique d'initialisation de l'algorithme FCM, ainsi, pallier l'inconvénient de stagner dans des optima locaux et rendre la segmentation des images IRM cérébrales plus efficace.

Chapitre 6

Contribution à la segmentation d'images par hybridation PSO-FCM

Sommaire

6.1	Introduction	75
6.2	Hybridation algorithmes bio-inspirés-FCM	76
6.3	Contribution	80
6.3.1	Motivation	80
6.3.2	Description	81
6.3.3	Algorithme FPSOFCM	84
6.3.4	Choix des paramètres de FPSOFCM	87
6.4	Validation et Résultats	88
6.4.1	Introduction	88
6.4.2	Base d'images	90
6.4.3	Evaluation et étude comparative	90
6.5	Conclusion	121

6.1 Introduction

Dans le cadre de l'amélioration des performances de FCM, nous allons décrire dans ce chapitre, un algorithme hybride de segmentation d'images IRM, nommé FPSOFCM, basé sur la méthode FCM et la métaheuristique PSO. L'objectif de cette hybridation est de remédier au problème de la dépendance de FCM à la configuration initiale des centres

de classes. L’algorithme FPSOFCM se base sur deux idées principales. La première idée consiste à définir une nouvelle fonction objectif de l’algorithme FPSO (Fuzzy PSO) [143], basée sur des indices de validité, afin de chercher les meilleurs centres de classes. La deuxième idée repose sur l’initialisation de FCM avec ces centres.

Dans la Section 6.2, nous exposons un état de l’art sur les différentes hybridations de FCM avec les algorithmes bio-inspirés. Dans la Section 6.3, nous détaillons notre algorithme de segmentation d’images FPSOFCM [144]. Le chapitre sera terminé par une étude comparative de résultats de segmentation sur des images IRM simulées et réelles.

6.2 Hybridation algorithmes bio-inspirés–FCM

Comme nous l’avons déjà mentionné dans le chapitre deux (Section 2.5.3.1), FCM a été abondamment utilisé en segmentation, vu son adaptation et son efficacité où les données manipulées se chevauchent ou lorsque l’information disponible est imprécise et incertaine, ce qui est le cas des images IRM cérébrales. Cependant, son inconvénient majeur est lié à l’initialisation aléatoire des centres de classes qui provoque une convergence prématurée, et mène à des optima locaux, sans atteindre l’optimum global. Pour remédier à cette problématique, plusieurs variantes hybrides de FCM ont été proposées dans la littérature [145–148].

Puisque le thème de cette thèse est l’étude des algorithmes bio-inspirés pour la segmentation d’images, nous allons présenter dans la suite, un état de l’art sur les différentes approches hybrides de FCM avec ces métaheuristiques.

- **CA-FCM**

Pour améliorer la segmentation des images IRM, W. Haijun *et al.* [149] ont proposé un automate cellulaire d’une seule dimension, minimisant la fonction objectif de FCM. Dans [150], C. Sompong et S. Wongthanavasut ont aussi proposé une nouvelle approche hybride CA-FCM pour la localisation des tumeurs cérébrales sur des images IRM. Les résultats expérimentaux sur plusieurs images de la base BraTS2013 [151], montrent une amélioration significative des performances de segmentation.

- **SA-FCM**

Un nouvel algorithme de segmentation d’images combinant FCM avec le recuit simulé est étudié dans [152]. Les auteurs utilisaient une fonction objectif basée sur le principe de classification de FCM et sur le système de refroidissement du recuit simulé, afin de

chercher les optima globaux. Li Qing Yang *et al.* [153] ont appliqué une hybridation de FCM avec une variante de recuit simulé CSA (*chaotic simulated annealing*) pour améliorer les résultats de la segmentation, aussi bien en terme de temps de calcul qu'en précision de segmentation.

- **GA-FCM**

Dans [154], une approche de segmentation floue des images IRM cérébrales basée sur un GA est développée. Les centres de classe représentent les individus de GA, et leur performance est calculée par un indice nommé FSym-index (*fuzzy symmetry index*) qui mesure la qualité de la partition obtenue. Afin d'éviter une mauvaise initialisation de FCM, M. A. Balafar *et al.* [155] ont hybridé l'algorithme FCM avec un GA pour séparer les structures cérébrales dans des IRM. L'algorithme génétique est utilisé pour trouver les centres initiaux de FCM, par minimisation d'une fonction objectif représentant la somme des distances de chaque pixel au centre de classes. Venkateswar *et al.* [156] ont hybridé une variante de FCM spatial avec un algorithme génétique pour la caractérisation des structures cérébrales. L'inconvénient de leur approche est d'avoir une complexité plus élevée par rapport à FCM classique. Pour déterminer automatiquement le nombre et les centres de classes dans images IRM cérébrales, Omid Jamshidi *et al.* [157] ont combiné le FCM avec un algorithme génétique. La fonction objectif de FCM a été utilisée comme la fitness de chaque individu.

- **ACO-FCM**

Yanfang Han et Pengfei Shi [158] ont combiné l'algorithme ACO avec un algorithme de classification floue. La métaheuristique est introduite pour ajuster la position initiale des centres de classes. Trois informations : niveau de gris ; gradient ; et le voisinage du pixel ont été utilisées dans le processus de segmentation. Les résultats expérimentaux, montrent l'efficacité de l'approche hybride. Dans [159], un modèle pour la segmentation d'images est présenté. Dans cette approche, chaque fourmi mémorise un objet de référence, qui sera mis à jour lorsqu'elle trouve une autre cible. Une mesure de connectivité floue est utilisée pour évaluer la similitude entre la cible et l'objet de référence. Afin de caractériser les tissus cérébraux d'une image IRM, M. Karnan et T. Logheshwari [160] ont aussi travaillé sur un algorithme hybridant ACO avec une segmentation floue sur deux étapes. Dans la première étape, l'image IRM cérébrale est segmentée par l'algorithme hybride pour extraire des régions pathologiques. Dans la deuxième étape une comparaison est effectuée

entre le résultat obtenu et la segmentation faite par un expert.

- **HS-FCM**

Afin d'éviter des solutions localement optimales, O. M. Alia *et al.* [161] ont présenté un nouvel algorithme flou basé sur la recherche d'harmonie, baptisé HFISA (*Harmony Fuzzy Image Segmentation Algorithm*). Dans HFISA, les centres de classes et les degrés d'appartenances des pixels à ces centres, sont utilisés comme des variables de décision. Les résultats sur des benchmarks de test, montrent l'efficacité de HFISA par rapport à FCM. Les mêmes auteurs, dans [162] ont utilisé l'algorithme HS pour trouver les centres de classes initiaux de FCM pour segmenter des images IRM normales et pathologiques.

- **AFSA-FCM**

X.Chu *et al.* [163] ont travaillé sur la sélection adaptative des centres de classes de FCM par la métaheuristique AFSA, pour un problème de segmentation. Le même principe a été utilisé dans [164] pour détecter les tumeurs dans les IRM cérébrales.

- **ABC-FCM**

M. Taherdangkoo *et al.* [165] ont décrit une méthode de segmentation d'images IRM cérébrales utilisant l'algorithme ABC, pour calculer deux paramètres de FCM. Les auteurs ont montré, non seulement que l'algorithme proposé avait des performances en terme de qualité de segmentation, mais aussi en temps de calcul. A. Alrosan *et al.* [166] ont appliqué la métaheuristique ABC pour chercher les centres de classes initiaux de FCM. Les résultats de la segmentation sur un ensemble d'images IRM cérébrales, montrent l'efficacité et la robustesse de leur méthode (ABC-FCM) par rapport à FCM standard.

- **GSA-FCM**

Pour améliorer les performances de l'algorithme FCM, E. Mozafari *et al.* [167] ont proposé une nouvelle approche hybride pour la segmentation d'images. Les auteurs ont incorporé l'algorithme de recherche gravitationnelle dans FCM, pour chercher les centres optimaux de classes.

- **Firefly-FCM**

Dans [168], P. Jitpakdee *et al.* ont récemment proposé de combiner FCM avec l'approche FA pour la segmentation des images. Dans cette approche, le FCM est appliqué pour guider le déplacement des lucioles. Les résultats expérimentaux sur 7 images de test, ont montré l'efficacité de cette approche hybride, comparée à K-means et FCM standard. W. Khamees *et al.* [169] ont également utilisé l'algorithme des lucioles pour trouver les

centres initiaux de FCM, afin d'éviter l'initialisation aléatoire qui mène à une stagnation dans des minima locaux. L'algorithme nommé FFA a été efficacement testé sur des images IRM simulées et réelles.

- **PSO-FCM**

En segmentation d'images, et afin de résoudre le problème de blocage dans des minima locaux dans la méthode FCM, plusieurs travaux ont été proposés dans la littérature, qui exploitent l'avantage de l'algorithme PSO. Das *et al.* [170] ont utilisé une variante modifiée de PSO pour la segmentation floue d'images. Leur approche est basée sur l'incorporation de l'information spatiale dans le calcul des degrés d'appartenance. Wensheng *et al.* [171] ont proposé une approche PSO floue, où chaque particule est considérée comme un centre de classe, et les particules parcourent l'espace de recherche pour trouver les meilleurs centres. Forghani *et al.* [172] ont, quant à eux, hybridé une variante de FCM nommée IFCM (*improved fuzzy c-means*) avec PSO, pour trouver les valeurs optimales de deux facteurs : la différence entre les caractéristiques des pixels voisins et leur location. Chun et Fang [173] ont décrit un algorithme hybride FCM-PSO pour la segmentation d'images. Le FCM est utilisé pour trouver les centres de classes maximisant une fonction de similarité, le PSO est appliqué pour affecter chaque pixel à une classe. Pour résoudre le problème de la sensibilité de FCM à l'initialisation aléatoire des centres de classes, Zhou [174] a développé un algorithme hybride basé sur une variante de PSO, appelée PPPSO (*Predator-Prey PSO*) et l'algorithme FCM. Zang et Bo [175] ont travaillé sur l'hybridation d'un FCM rapide (*Fast FCM*) avec PSO, permettant de déterminer automatiquement le nombre et les centres de classes pour un problème de segmentation d'images. Gopal *et al.* [52] ont aussi présenté deux étapes pour la détection des zones tumorales dans des IRM cérébrales. Le prétraitement et l'amélioration de l'image sont effectués dans la première étape, puis la segmentation par FCM-PSO dans la seconde. Dans [176], Yanling and Shen ont publié un algorithme de segmentation FCM basé sur une variante de PSO, baptisée PWLCPSO (*PieceWise Linear Chaotic map PSO*). D'abord PWLCPSO est utilisé pour avoir les centres initiaux. Puis, les images sont segmentées par le FCM standard. Benaichouche *et al.* [51] ont introduit une méthode de segmentation d'images IRM sur trois étapes. La première étape consiste à améliorer l'initialisation de FCM par un algorithme PSO. La deuxième étape, concerne le regroupement des pixels en utilisant le critère de FCM modifié. La dernière étape est une post-segmentation. Récemment, S. Mirghasemi *et al.*

[177] ont hybridé FCM avec PSO pour définir une nouvelle métrique de similarité qui combine plusieurs caractéristiques de voisinage. L'algorithme a été testé avec succès sur des images réelles et synthétiques.

6.3 Contribution

6.3.1 Motivation

Comme nous l'avons déjà mentionné, l'utilisation d'un algorithme supervisé pour la segmentation d'images IRM, nécessite de créer une base d'apprentissage pour chaque classe et pour chaque patient, cette tâche s'avère difficile pour les experts. Aussi, le résultat de la segmentation est très dépendant de la phase d'apprentissage. Ces raisons nous ont fait préférer un algorithme non supervisé, et plus précisément FCM sur lequel nous sommes basé dans cette thèse.

L'algorithme FCM a été largement appliqué en segmentation d'images IRM cérébrales, au regard de son adaptation et son efficacité à traiter des données de nature imprécise et incertaine [140, 142]. Cependant, ses performances dépendent considérablement de la configuration initiale des centres des classes. Si cette configuration est choisie aléatoirement, l'algorithme peut rester piégé dans des optima locaux, sans atteindre l'optimum global. Pour pallier à cet inconvénient, nous avons pensé à utiliser une métaheuristique d'optimisation qui permet d'estimer les positions initiales de ces centres.

Les métaheuristicques sont des approches très robustes pour résoudre des problèmes d'optimisation aux données incertaines et imprécises (bruitées). Comme nous l'avons déjà présenté dans le chapitre 3, il existe une multitude de métaheuristicques dans la littérature. L'objectif de chaque approche métaheuristique et de parcourir efficacement l'espace de recherche, tout en respectant un compromis entre l'exploration (recherche globale) et l'exploitation (recherche locale). Le principe de diversification (l'exploration) et d'intensification (exploitation) est un aspect critique pour toute approche métaheuristique. L'exploration consiste à parcourir complètement de nouvelles zones dans l'espace de recherche, tandis que l'exploitation est le processus de parcourir le voisinage au sein de ces zones déjà visitées [178]. En général, plus un algorithme est intensifie plus il explore localement des solutions proches aux celles trouvées, et plus il converge rapidement. Cependant, si l'intensification est trop grande, l'algorithme risque d'être stagné autour d'optima locaux.

Pour notre problématique, nous avons choisi l'algorithme PSO, en tenant compte de ses avantages à résoudre les problèmes d'optimisation à variables continues. Ce qui est le cas pour FCM où chaque centre de classe représente une variable continue dans l'espace des attributs. Afin de préserver l'équilibre entre l'intensification et la diversification, nous avons pensé à développer un algorithme hybride PSO-FCM pour la segmentation d'images IRM cérébrales. Dans ce cas, une nouvelle fonction objectif est définie pour l'algorithme FPSO (*Fuzzy PSO*) qui agit comme une approche d'optimisation globale pour trouver les centres de classes. Ensuite, le FCM effectue une recherche locale pour la classification des pixels.

6.3.2 Description

L'algorithme de segmentation que nous allons décrire, nommé FPSOFCM [144] (*Fuzzy Particle Swarm Optimization for Fuzzy C-Means*) se déroule en 2 étapes :

1. Etape de calcul des centres de classes par FPSO, basée sur une nouvelle fonction objectif.
2. Etape de segmentation d'image par FCM, dont la configuration initiale des centres est celle obtenue dans l'étape 1.

Le Tableau 6.1 présente les notations utilisées dans notre algorithme.

Tableau 6.1: Notations.

C	<i>nombre de classes</i>	Z	<i>centres de classes</i>
P	<i>nombre de particules</i>	\bar{Z}	<i>centres de classes (meilleure particule)</i>
N	<i>nombre de pixels dans l'image</i>	U	<i>matrice des degrés d'appartenances</i>
O_i	<i>pixel i</i>	<i>Best-global-fitness</i>	<i>meilleure fitness obtenue</i>
X	<i>positions des particules</i>	J	<i>fonction objectif de FCM</i>
V	<i>vitesses des particules</i>	ε	<i>erreur</i>
PC	<i>partition coefficient</i>	c_1, c_2	<i>coefficients d'accélération</i>
PE	<i>partition entropie</i>	w	<i>coefficient d'inertie</i>
P_{best}	<i>meilleure position de la particule</i>	V_{max}	<i>vitesse maximale</i>
G_{best}	<i>meilleure position de l'essaim</i>	I_{termax}	<i>nombre maximum d'itérations</i>
<i>Fitness</i>	<i>nouvelle fonction objectif</i>	t	<i>compteur d'itérations</i>

6.3.2.1 Reformulation

Dans cette section, nous allons reformuler le problème de segmentation d'images en un problème d'optimisation, et pour cela, nous allons définir pour l'algorithme FPSO une

nouvelle fonction objectif basée sur plusieurs indices d'évaluation.

L'algorithme FPSO (Fuzzy Particle Swarm Optimisation), proposé par Pang *et al.* [143] pour résoudre le problème du voyageur de commerce, a été appliqué avec succès pour la classification floue des données par H. Izakian et A. Abraham [179]. L'étude de ces deux contributions, a mené à l'idée de définir une nouvelle fonction objectif basée sur des indices de validité. Dans la suite de cette section, nous allons décrire l'algorithme FPSO pour un problème de segmentation d'images.

Dans FPSO la position de chaque particule est représentée dans une matrice $X(N \times C)$ qui contient les degrés d'appartenance floue des pixels $\{o_1, o_2, \dots, o_N\}$ aux centres de classes $\{Z_1, Z_2, \dots, Z_c\}$, la matrice X est décrite comme suit :

$$X = \begin{bmatrix} u_{11} & \cdots & u_{1c} \\ \vdots & \ddots & \vdots \\ u_{N1} & \cdots & u_{Nc} \end{bmatrix} \quad (6.1)$$

où u_{ik} est le degré d'appartenance du pixel i au centre de la classe k , avec les conditions définies dans les équations (Eq. 2.4) et (Eq. 2.5). Donc, la matrice de position X de chaque particule est similaire à la matrice des degrés d'appartenance U dans l'algorithme FCM standard. Ainsi, la vitesse de chaque particule est représentée dans la matrice $V(N \times C)$.

La mise à jour de X et V est donnée par les équations (Eq. 6.2) et (Eq. 6.3) :

$$V(t+1) = \omega \otimes V(t) \oplus (c_1 r_1) \otimes (Pbest(t) \ominus X(t)) \oplus (c_2 r_2) \otimes (Gbest(t) \ominus X(t)) \quad (6.2)$$

$$X(t+1) = X(t) \oplus V(t+1) \quad (6.3)$$

où $Pbest$ est la meilleure position déjà visitée par la particule i (Eq. 6.4), $Gbest$ est la meilleure position de tout l'essaim (Eq. 6.5).

$$Pbest(t+1) = \begin{cases} Pbest(t), & \text{si } f(X(t+1)) \geq Pbest(t) \\ X(t+1), & \text{sinon} \end{cases} \quad (6.4)$$

$$Gbest(t+1) = \underset{Pbest}{\operatorname{argmin}} f(Pbest_i(t+1)), 1 \leq i \leq P \quad (6.5)$$

les symboles \oplus et \ominus indiquent l'addition et la soustraction entre matrices. Le symbole \otimes indique la multiplication de tous les éléments de la matrice par un nombre réel.

Après la mise à jour de la matrice X , et pour respecter les conditions de l'équation (Eq. 2.4) et (Eq. 2.5), il est nécessaire de normaliser la matrice X . Premièrement, tous les éléments négatifs deviennent 0. Si tous les éléments d'une ligne de X sont égaux à 0, on les recalcule en affectant des nombres aléatoires dans l'intervalle $[0, 1]$ et la matrice subit les transformations suivantes :

$$X_{normalisee} = \begin{bmatrix} u_{11}/\sum_{k=1}^c u_{1k} & \dots & u_{1c}/\sum_{k=1}^c u_{1k} \\ \vdots & \ddots & \vdots \\ u_{N1}/\sum_{k=1}^c u_{Nk} & \dots & u_{Nc}/\sum_{k=1}^c u_{Nk} \end{bmatrix} \quad (6.6)$$

6.3.2.2 Fonction objectif

En FPSO, comme dans l'algorithme standard PSO, il est nécessaire d'avoir une fonction objectif pour évaluer les solutions trouvées. En étudiant les différentes méthodes hybrides PSO-FCM (Section 6.2), on a remarqué que la plupart de ces approches utilisaient la fonction objectif de FCM (Eq. 2.8) comme une fonction d'évaluation.

Dans notre approche, nous avons défini une nouvelle fonction objectif décrite par l'équation suivante :

$$Fitness = \frac{PE + \alpha}{PC + \beta} \quad (6.7)$$

où PE (partition entropy), PC (partition coefficient) sont deux indices de validité de classification floue proposés par Bezdek [180, 181], définis comme suit :

$$PE = \frac{-\sum_{i=1}^N \sum_{k=1}^c u_{ki} \log(u_{ki})}{N} \quad (6.8)$$

$$PC = \frac{\sum_{i=1}^N \sum_{k=1}^c u_{ki}^2}{N} \quad (6.9)$$

La meilleure classification sera celle pour laquelle PE est minimal et PC est maximal.

α c'est la somme des distances de chaque pixel o_i aux centres de classes Z_k , définie par :

$$\alpha = \sum_{k=1}^c \sum_{i=1}^N \|o_i - z_k\|^2 \quad (6.10)$$

β c'est la somme des distances entre les différentes classes définie par [182] :

$$\beta = \sum_{\forall k=1..C, j, k \neq j} \|z_k - z_j\|^2 \quad (6.11)$$

Donc la fonction objectif sera décrite comme suit :

$$Fitness = \frac{-\sum_{i=1}^N \sum_{k=1}^c u_{ki} \log(u_{ki})}{N} + \sum_{k=1}^c \sum_{i=1}^N \|o_i - z_k\|^2}{\sum_{i=1}^N \sum_{k=1}^c u_{ki}^2}{N} + \sum_{\forall k=1..C, j, k \neq j} \|z_k - z_j\|^2 \quad (6.12)$$

L'algorithme proposé va minimiser la nouvelle fonction objectif (Eq. 6.12) en respectant la troisième et la quatrième conditions de la segmentation présentées dans la Section 2.2.2. Donc, une valeur minimale de *Fitness* est obtenue lorsque la valeur du numérateur $PE + \alpha$ est minimale, et la valeur du dénominateur $PC + \beta$ est maximale. En d'autres termes, les particules de FPSO vont parcourir l'espace de recherche pour trouver les meilleurs centres de classes qui augmentent à la fois l'homogénéité intra classe (3^{ème} condition), et l'hétérogénéité inter classes (4^{ème} condition).

Pour avoir une meilleure partition, la fonction objectif proposée prend en considération deux aspects importants, qui sont : la cohésion, où les données dans une classe doivent être aussi similaires que possible les unes des autres, et la séparation, où les classes doivent être dissimilaires les unes des autres. Donc, cette fonction respecte les deux premières règles de Reynolds (Section 4.2.1)

6.3.3 Algorithme FPSOFCM

Dans l'algorithme proposé FPSOFCM [144], premièrement, les P particules sont initialisées aléatoirement. La position de chaque particule représente la matrice des degrés d'appartenance U . Pour chaque itération, les particules se déplacent dans l'espace de recherche afin de minimiser la fonction objectif *Fitness* (Eq. 6.12). Après la convergence de FPSO (étape 1), les centres de classes \bar{Z} correspondant à la meilleure solution G_{best} obtenue, seront utilisés comme configuration initiale de FCM, qui sera appliqué pour segmenter l'image originale. A la fin du déroulement de FCM, une dernière étape est nécessaire pour reconstruire l'image segmentée. C'est l'étape de *défuzzification*, où chaque pixel i sera affecté à la classe pour laquelle le degré d'appartenance u_{ki} est maximal.

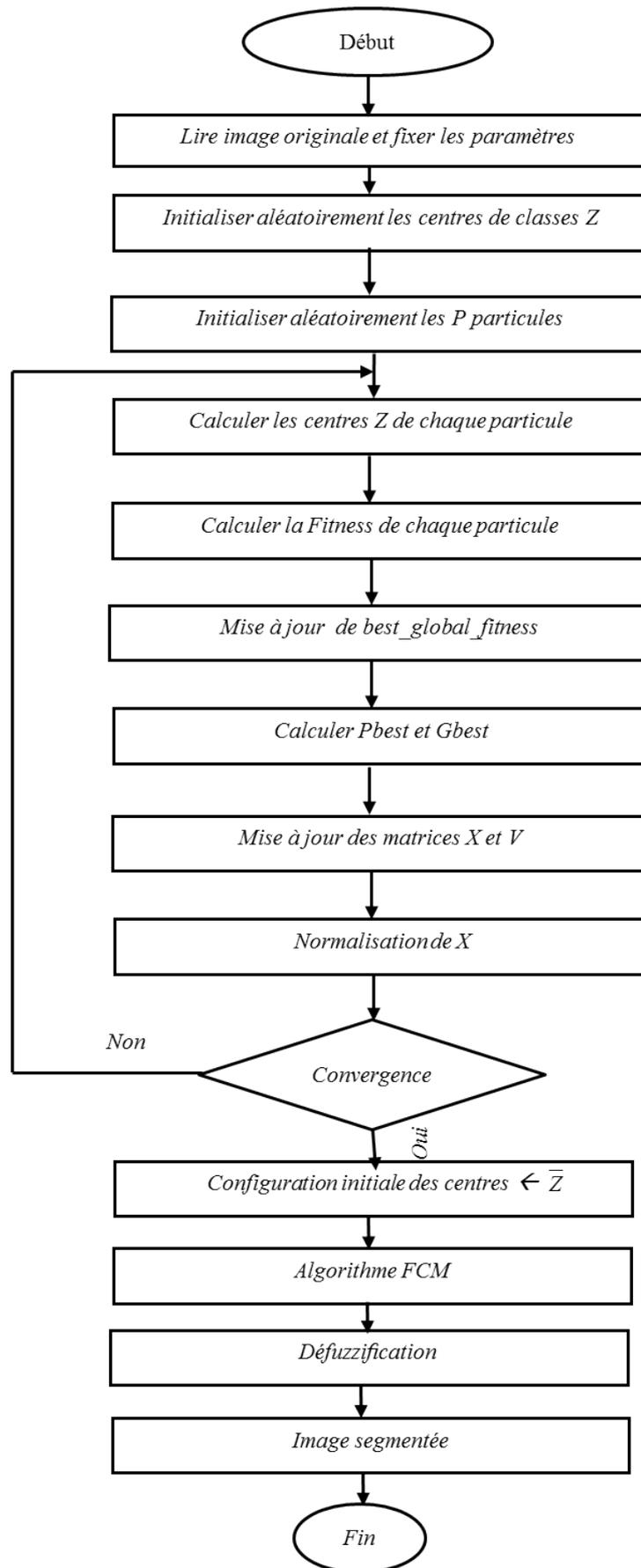


Figure 6.1: Schéma général de l'approche proposée.

Algorithme 6.1 : FPSOFCM [144]**Données :** image IRM originale**Résultat :** image segmentée

```

1 début
2   Fixer  $C, m, itermax, \varepsilon, P, c_1, c_2, w, v_{max}$ ,  $best\_global\_fitness = +\infty$ ;
   /*          Algorithme FPSO          */
3   Créer un essaim de  $P$  particules : initialiser  $X(N \times C \times P)$  et  $V(N \times C \times P)$ ;
4   Initialiser les centres de classes  $Z(C \times P)$ ;
5   Pour  $i \leftarrow 1$  à  $itermax$  faire
6     Calculer les centres de classes  $Z$  de chaque particule par (Eq. 2.7);
7     Calculer la Fitness de chaque particule par (Eq. 6.12);
8     Mise à jour de la fitness globale  $best\_global\_fitness$ ;
9     Calculer  $Pbest$  et  $Gbest$  par (Eq. 6.4) et (Eq. 6.5);
10    Mise à jour de  $V$  de chaque particule par (Eq. 6.2);
11    Limiter la vitesse dans l'intervalle  $[-v_{max}, +v_{max}]$ ;
12    Mise à jour de  $X$  de chaque particule par (Eq. 6.3);
13    Normaliser  $X$  par (Eq. 6.6);
14    Si  $|best\_global\_fitness^{(t)} - best\_global\_fitness^{(t+1)}| < \varepsilon$  alors
15      | break;
16    finsi
17  finpour
18  Centres de classes  $Z \leftarrow \bar{Z}$ ;
   /*          Algorithme FCM          */
19  Pour  $i \leftarrow 1$  à  $itermax$  faire
20    | Mettre à jour la matrice  $U$  par (Eq. 2.6);
21    | Mettre à jour le vecteur des centroïdes  $Z$  (Eq. 2.7);
22    | Calculer la fonction objectif  $J$  (Eq. 2.8);
23    | Si  $|J^{(t)} - J^{(t+1)}| < \varepsilon$  alors
24      | break;
25    | finsi
26  finpour
27  Défuzzification de la matrice  $U$ ;
28 fin

```

6.3.4 Choix des paramètres de FPSOFCM

- **Nombre de particules P**

Le nombre de particules de l'essaim est fixé à 20 particules (choix empirique).

- **Topologie de voisinage**

Pour des raisons de rapidité, nous avons défini une topologie $Gbest$ (Section 4.4) où chaque particule de l'essaim est informée par la totalité des particules.

- **Facteur d'inertie w**

Nous avons utilisé un facteur d'inertie dynamique, présenté dans [132] qui diminue au cours du temps, il commence par une valeur de $w_{max} = 0.9$ et décroît linéairement pour arriver à $w_{min} = 0.4$ selon l'équation (Eq. 4.5).

- **Coefficients c_1, c_2**

Les valeurs des coefficients d'accélération $c_1 = c_2 = 1.496$ (Section 4.5.2).

- **Critères d'arrêt**

Dans les deux étapes de l'algorithme FPSOFCM, nous avons utilisé deux critères, qui sont : le nombre maximum d'itérations $itermax = 100$, et l'erreur $\varepsilon = 10^{-5}$.

- **Vitesse maximale $Vmax$**

Comme nous l'avons déjà mentionné dans la Section 4.5.2, la valeur de $Vmax$ est généralement dépendante de l'échelle du problème. Dans notre cas, $Vmax$ prend la valeur 255, qui représente le plus grand niveau de gris dans une image.

- **Degré de flou m**

Ce paramètre ($m > 1$) contrôle le degré de flou de la matrice U . Si $m = 1$ alors l'algorithme n'est pas flou (k-means). Le paramètre m n'a pas une influence majeure sur le résultat de la segmentation [183]. Nous avons fixé $m = 2$, valeur qui est souvent choisie dans la littérature.

- **Mesure de distance**

Nous avons choisi la distance euclidienne, qui est la plus simple et la plus rapide à calculer.

- **Nombre de classes**

L'objectif de notre travail est de segmenter des images IRM cérébrales. Nous cherchons donc à séparer les différents tissus cérébraux (matière grise, matière blanche, etc.). L'indice le plus utilisé pour déterminer le nombre optimal de classe est l'indice de Xie-Beni [184] (XB) défini comme suit :

$$XB = \frac{\sum_{k=1}^c \sum_{i=1}^N u_{ki}^2 \|o_i - z_k\|^2}{N \times \min_{k \neq j} \|z_k - z_j\|^2} \quad (6.13)$$

le nombre optimal de classes correspond à une valeur minimale de XB.

Pour déterminer le nombre de classes, nous avons testé l'algorithme FCM standard pour différentes valeurs de $C \in \{3, 4, 5, 6\}$ sur un échantillon de 8 images IRM simulées (normales) issues de la base *Brainweb* [141]. Ces images sont affectées de différents niveaux de bruit Gaussien (0% 3% 5%) et différents niveaux d'inhomogénéité RF (0% 20% 40%).

Les résultats de test sont présentés dans le Tableau 6.2. Les valeurs en gras correspondent aux meilleurs résultats obtenus.

Tableau 6.2: Changement de l'indice XB par rapport au nombre de classes C.

	<i>Image 1</i>	<i>Image 2</i>	<i>Image 3</i>	<i>Image 4</i>	<i>Image 5</i>	<i>Image 6</i>	<i>Image 7</i>	<i>Image 8</i>
$C=3$	0,096169	0,070612	0,070298	0,072799	0,098441	0,073239	0,065983	0,087612
$C=4$	0,05912	0,04613	0,052531	0,04756	0,056433	0,052495	0,04516	0,053752
$C=5$	0,086211	0,054672	0,092823	0,059419	0,102207	0,061104	0,072265	0,100119
$C=6$	0,079772	0,076301	0,092644	0,075179	0,116992	0,060779	0,077173	0,101861

D'après le Tableau 6.2 et la Figure 6.2, on remarque que l'indice XB prend une valeur minimale pour $C = 4$ pour toutes les images. Le nombre de classe C sera donc pris égal à 4, il correspond aux classes suivantes : 1) matière grise, 2) matière blanche, 3) liquide céphalo-rachidien, et 4) le fond de l'image. Cependant, on peut augmenter ce nombre dans le cas des images IRM pathologiques ($C = 5$ ou plus).

6.4 Validation et Résultats

6.4.1 Introduction

Dans cette section, l'algorithme FCM et l'approche proposée FPSOFCM [144] sont mis en œuvre et évalués dans le contexte de la segmentation des images IRM cérébrales. Nous allons premièrement décrire les différentes images IRM simulées et réelles, qu'on a utilisées pour évaluer les résultats de la segmentation. Ensuite, une évaluation qualitative et quantitative de la segmentation est présentée selon différents critères. Enfin, nous allons comparer FPSOFCM [144] avec l'approche FPSO présentée dans [179].

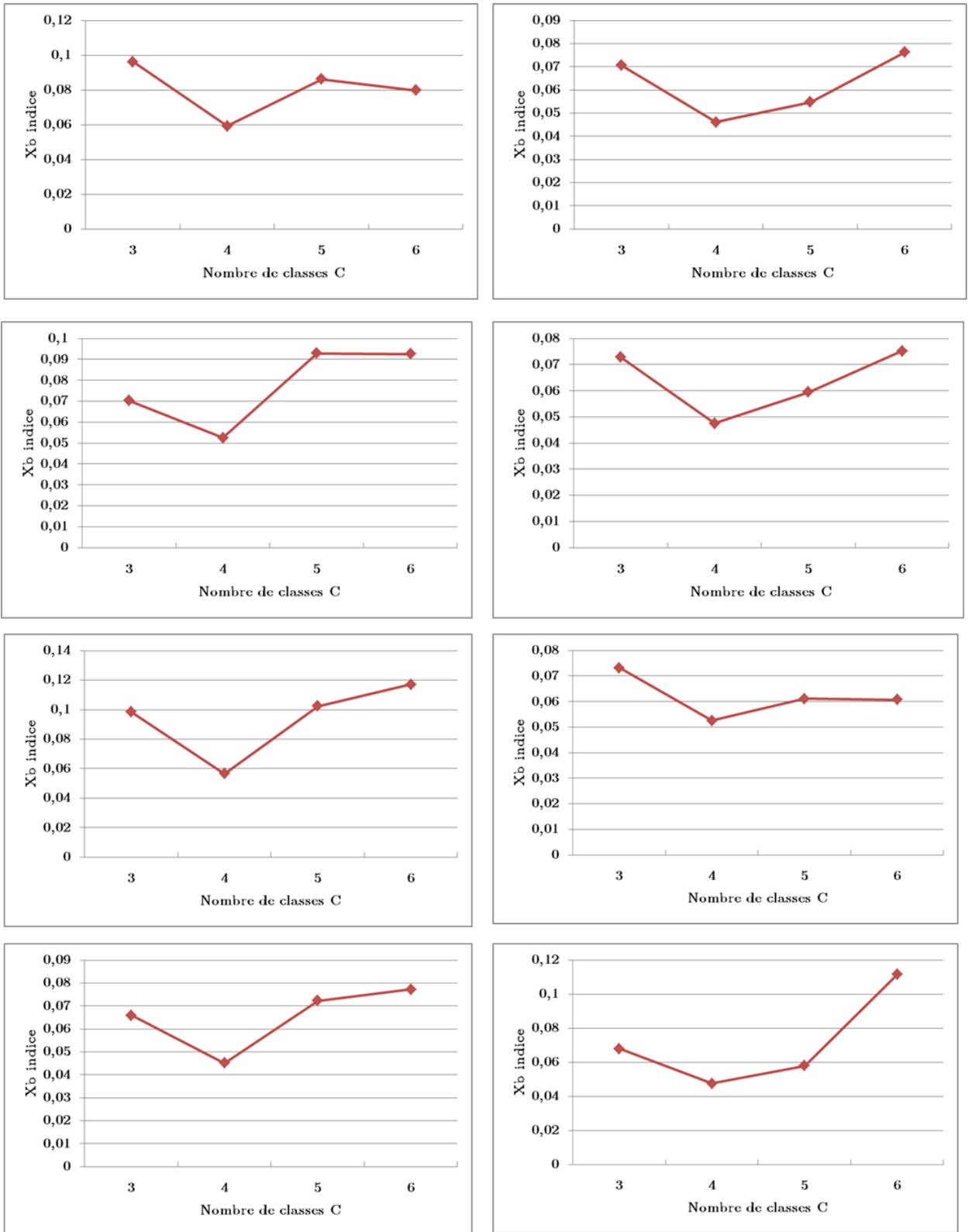


Figure 6.2: Changement de l'indice XB par rapport au nombre de classes C.

6.4.2 Base d'images

L'évaluation de notre approche de segmentation a été réalisée à la fois sur des images IRM simulées et réelles.

6.4.2.1 Images simulées

Ce sont des images de synthèse, générées à l'aide d'un simulateur du centre d'imagerie cérébrale de l'institut neurologique de l'université McGill à Montréal [141]. Avec le simulateur *Brainweb*, l'utilisateur a la possibilité, de sélectionner une modalité d'acquisition (T1, T2, DP), de choisir une épaisseur de coupe et d'ajouter du bruit et/ou d'hétérogénéité RF.

Pour nos évaluations, nous avons construit, par le simulateur *Brainweb*, 18 fantômes¹⁴ (T1, DP) de taille 181x217x181, la taille de voxel 1x1x1 mm³. Chaque volume est affecté par trois différents niveaux de bruit Gaussien (0% 3% 5%) et trois niveaux d'inhomogénéité RF (0% 20% 40%). Ces images sont accompagnées d'une vérité terrain (*ground truth*) rendant possible l'évaluation et la comparaison de l'approche proposée. Les images simulées sont divisées en 6 échantillons dont les caractéristiques sont décrites dans l'Annexe C.

6.4.2.2 Les images réelles

Les images réelles sur lesquelles nous avons travaillé dans le cadre de cette thèse, sont des images pathologiques acquises de différentes sources :

- SICAS Medical Image Repository [151, 185] : est une plateforme pour la gestion des images médicales (T1, T2, DP, taille 240x240). A travers cette plateforme, les utilisateurs peuvent télécharger les images de test et uploader leurs résultats.
- Images pathologiques téléchargées du web : image1¹⁵ (180x218), image2¹⁶ (176x176), image3 (160x199) [186].

6.4.3 Evaluation et étude comparative

Les algorithmes FCM standard et FPSOFCM ont été implémentés sous l'environnement Matlab version 7, et exécutés dans un PC Pentium Dual-core E5700/3.00 GHz/2.00 GB RAM. Les paramètres choisis pour ces deux algorithmes, sont décrits dans la Section 6.3.4.

14. Un fantôme est un volume cérébral simulé.

15. <https://www.mathworks.com/matlabcentral/fileexchange/> [Online; 2017]

16. <http://clinical.netforum.healthcare.philips.com/global/Explore/Case-Studies/MRI/Brain-tumor> [Online; 2017]

Pour évaluer et comparer les performances des algorithmes développés, nous avons utilisé une base composée de 60 images saines et pathologiques (54 simulées + 6 réelles).

6.4.3.1 Evaluation de la classification

(a) Indices de validité

Pour mesurer la qualité de la classification, nous avons choisi quatre indices de validité les plus utilisés pour la classification floue : PC, MPC, PE, XB décrits respectivement par les équations : (Eq. 6.9), (Eq. 6.14), (Eq. 6.8) et (Eq. 6.13). Une bonne classification est obtenue, pour des valeurs de PC, MPC élevées et des valeurs PE, XB faibles.

L'indice MPC (*Modified Partiton Coefficient*) proposé par Dave [187] pour réduire l'évolution monotone de l'indice PC au nombre de classes C , il est défini comme suit :

$$MPC = 1 - \frac{C}{C-1} (1 - PC) \quad (6.14)$$

Les résultats moyens après 10 exécutions indépendantes de FCM et FPSOFCM sont présentés dans les Tableaux 6.3 et 6.4. Les meilleurs résultats sont indiqués en gras. L'écart type (entre parenthèses) est calculé pour vérifier la stabilité de notre approche par rapport à FCM standard.

• Résultats sur des images IRM (coupes Axiales)

Les résultats obtenus dans cette section concernent les images de l'échantillon 1 (Tableau C.1), qui sont affectées de bruit Gaussien (0% 3% 5%) et d'inhomogénéité RF (0% 20% 40%). Ces images sont segmentées en quatre classes (régions) : fond, liquide céphalorachidien (LCR), matière grise (MG) et matière blanche (MB).

Les Tableaux 6.3 et 6.4 donnent la moyenne et l'écart type des indices PC, MPC, PE, XB obtenus par FCM et FPSOFCM sur des images pondérées respectivement en T1 et DP. D'après ces tableaux, on constate que notre algorithme FPSOFCM est plus robuste et plus stable que l'algorithme FCM vis-à-vis les différents niveaux de bruit et d'inhomogénéité RF.

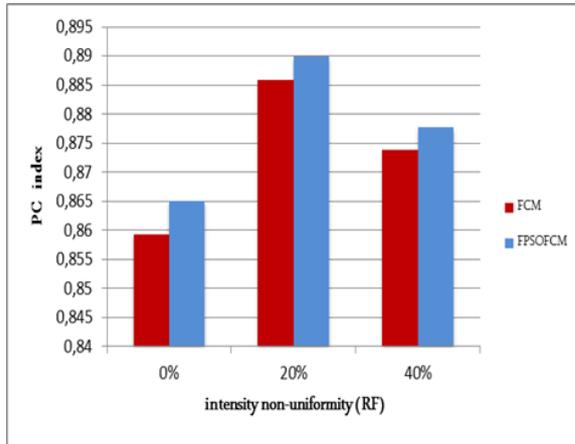
Les Figures 6.3 à 6.10 sont les interprétations graphiques des Tableaux 6.3 et 6.4. Elles donnent les valeurs moyennes des indices d'évaluation (PC, MPC, PE, XB) obtenus par FCM et FPSOFCM sur l'ensemble de la base des images (Tableau C.1 et C.2) simulées pondérées respectivement en T1 (échantillon 1) et DP (échantillon 2). Ces indices sont présentés selon différents niveaux de bruit Gaussien et différents niveaux d'inhomogénéité RF. Ces figures montrent que la méthode développée FPSOFCM

Tableau 6.3: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en T1.

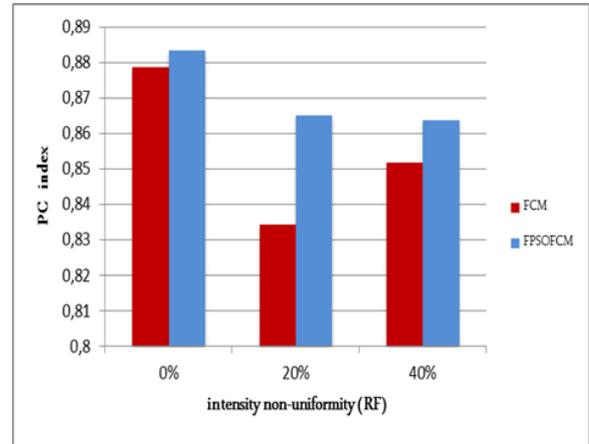
Bruit	0%		3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM	
0%	<i>PC</i>	0.859307	0.865014	0.883779	0.887281	0.866189	0.870844
		(0.017119)	(0.000003)	(0.010504)	(0.000002)	(0.009304)	(0.000005)
	<i>MPC</i>	0.812410	0.820019	0.845038	0.849708	0.821585	0.827791
		(0.022825)	(0.000004)	(0.014006)	(0.000003)	(0.012405)	(0.000007)
<i>PE</i>	0.274140	0.262982	0.228918	0.222143	0.264099	0.255692	
	(0.033470)	(0.000005)	(0.020319)	(0.000005)	(0.016805)	(0.000009)	
<i>XB</i>	0.059096	0.056250	0.047482	0.046129	0.053554	0.052526	
	(0.008536)	(0.000004)	(0.004057)	(0.000003)	(0.002051)	(0.000006)	
20%	<i>PC</i>	0.885871	0.889928	0.878348	0.881722	0.849101	0.849106
		(0.012171)	(0.000001)	(0.010117)	(0.000002)	(0.000003)	(0.000005)
	<i>MPC</i>	0.847829	0.853238	0.837797	0.842297	0.798801	0.798809
		(0.016228)	(0.000002)	(0.013489)	(0.000003)	(0.000005)	(0.000007)
<i>PE</i>	0.222145	0.214137	0.238707	0.232244	0.295455	0.295445	
	(0.024022)	(0.000004)	(0.019375)	(0.000004)	(0.000006)	(0.000010)	
<i>XB</i>	0.048853	0.047558	0.047464	0.046461	0.053754	0.053749	
	(0.003883)	(0.000004)	(0.002999)	(0.000003)	(0.000003)	(0.000004)	
40%	<i>PC</i>	0.873905	0.877773	0.868160	0.872280	0.831641	0.831645
		(0.011606)	(0.000003)	(0.012353)	(0.000004)	(0.000003)	(0.000003)
	<i>MPC</i>	0.831874	0.837031	0.824213	0.829707	0.775521	0.775527
		(0.015475)	(0.000004)	(0.016471)	(0.000006)	(0.000004)	(0.000004)
<i>PE</i>	0.246382	0.238932	0.256293	0.248716	0.328349	0.328341	
	(0.022358)	(0.000007)	(0.022717)	(0.000008)	(0.000006)	(0.000006)	
<i>XB</i>	0.046414	0.045157	0.052483	0.051851	0.056434	0.056432	
	(0.003776)	(0.000004)	(0.001889)	(0.000005)	(0.000002)	(0.000001)	

Tableau 6.4: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en DP.

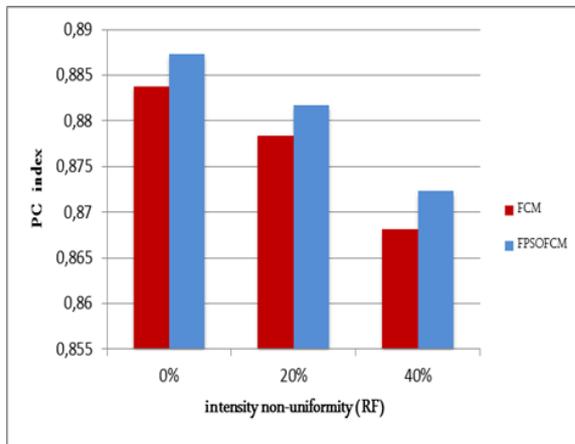
Bruit	0%		3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM	
0%	<i>PC</i>	0.878638	0.883454	0.843431	0.866682	0.840327	0.844059
		(0.014452)	(0.000001)	(0.028480)	(0.000004)	(0.014926)	(0.011195)
	<i>MPC</i>	0.838184	0.844606	0.791241	0.822243	0.787103	0.792079
		(0.019269)	(0.000001)	(0.037974)	(0.000005)	(0.019901)	(0.014926)
<i>PE</i>	0.226305	0.217303	0.290395	0.248917	0.300199	0.293649	
	(0.027008)	(0.000006)	(0.050808)	(0.000010)	(0.026195)	(0.019647)	
<i>XB</i>	0.097531	0.077069	0.164122	0.080158	0.093923	0.086644	
	(0.061385)	(0.000001)	(0.102838)	(0.000005)	(0.029114)	(0.021835)	
20%	<i>PC</i>	0.834348	0.865104	0.834806	0.850475	0.848169	0.855135
		(0.030755)	(0.000004)	(0.025585)	(0.015668)	(0.013932)	(0.000001)
	<i>MPC</i>	0.779130	0.820138	0.779741	0.800634	0.797559	0.806846
		(0.041006)	(0.000005)	(0.034114)	(0.020891)	(0.018576)	(0.000001)
<i>PE</i>	0.306138	0.250298	0.306282	0.278565	0.286783	0.274710	
	(0.055836)	(0.000011)	(0.045254)	(0.027714)	(0.024149)	(0.000004)	
<i>XB</i>	0.148764	0.078658	0.123405	0.088639	0.092216	0.074629	
	(0.070107)	(0.000002)	(0.056773)	(0.034766)	(0.035177)	(0.000002)	
40%	<i>PC</i>	0.851844	0.863569	0.839254	0.844708	0.830741	0.830742
		(0.017910)	(0.000000)	(0.012491)	(0.008178)	(0.000005)	(0.000004)
	<i>MPC</i>	0.802458	0.818092	0.785672	0.792943	0.774321	0.774323
		(0.023881)	(0.000000)	(0.016654)	(0.010904)	(0.000006)	(0.000006)
<i>PE</i>	0.279198	0.259070	0.300728	0.290824	0.329365	0.329363	
	(0.030740)	(0.000001)	(0.022676)	(0.014847)	(0.000009)	(0.000008)	
<i>XB</i>	0.092358	0.065810	0.094509	0.083754	0.056475	0.056475	
	(0.040555)	(0.000000)	(0.024662)	(0.016170)	(0.000002)	(0.000002)	



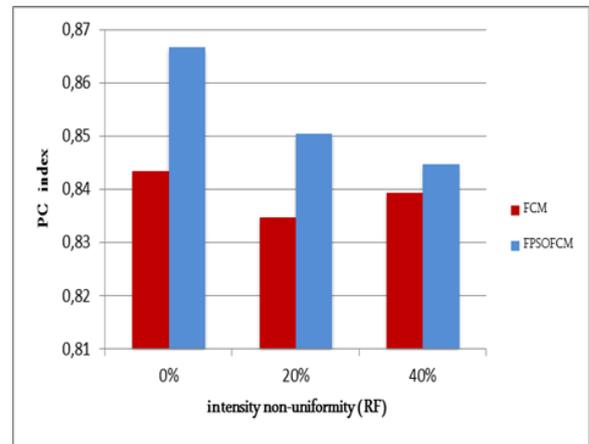
(a) Bruit= 0%



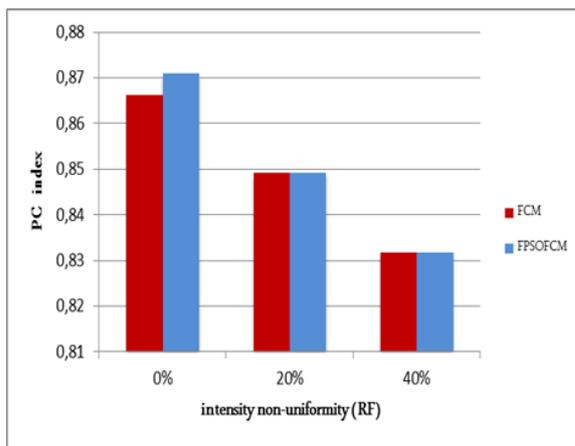
(a) Bruit= 0%



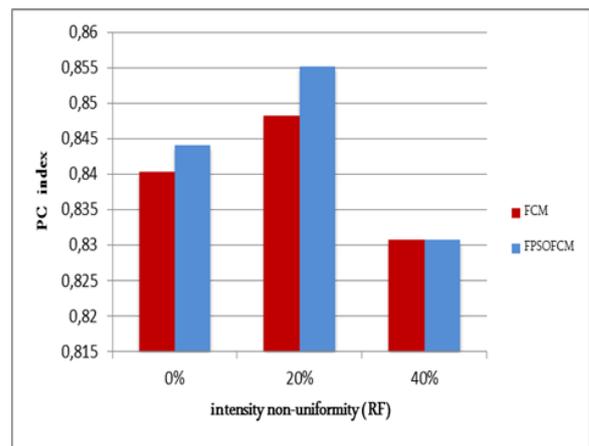
(b) Bruit= 3%



(b) Bruit= 3%



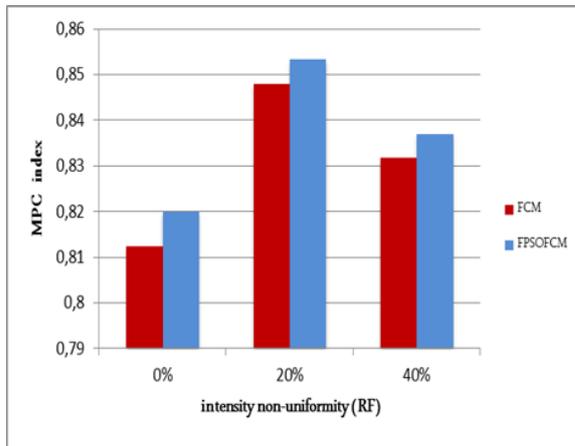
(c) Bruit= 5%



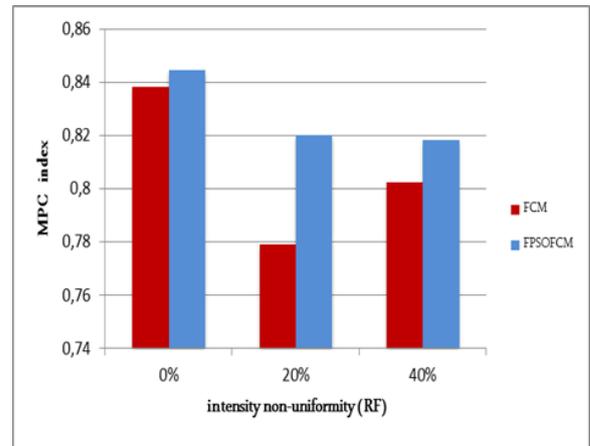
(c) Bruit= 5%

Figure 6.3: Comparaison de l'indice PC sur des images simulées pondérées en T1.

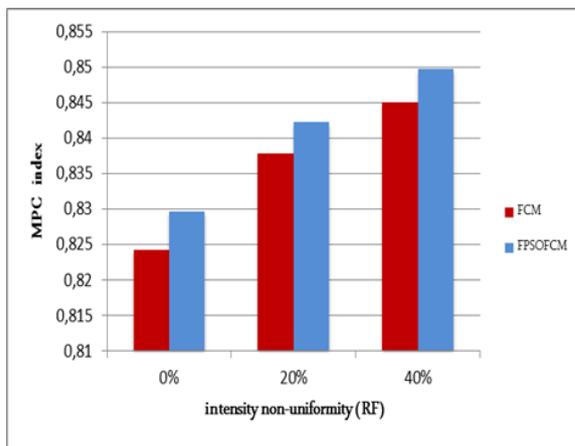
Figure 6.4: Comparaison de l'indice PC sur des images simulées pondérées en DP.



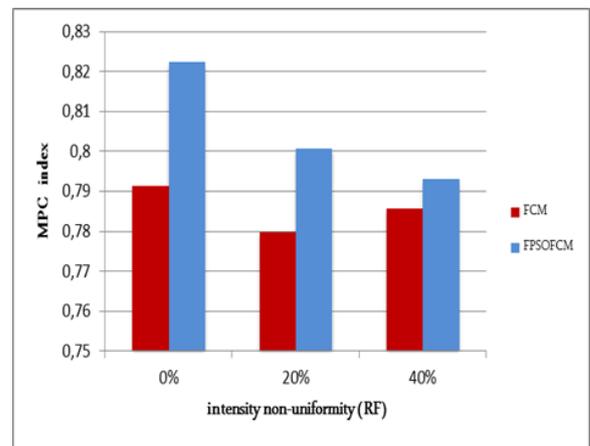
(a) Bruit= 0%



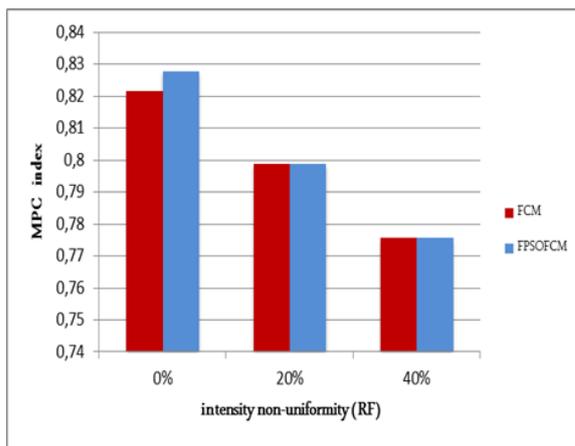
(a) Bruit= 0%



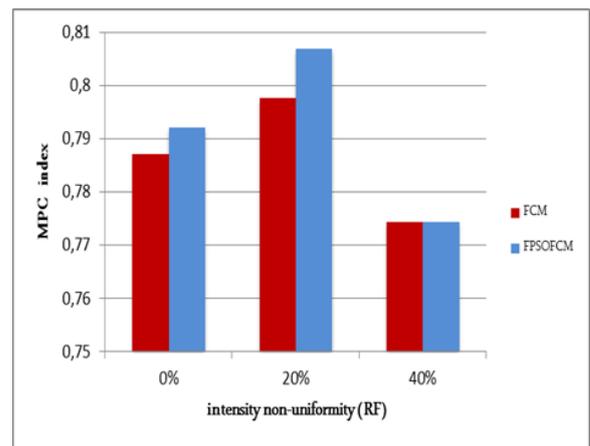
(b) Bruit= 3%



(b) Bruit= 3%



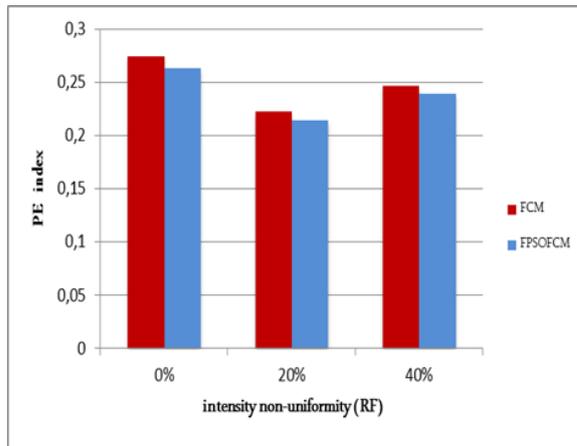
(c) Bruit= 5%



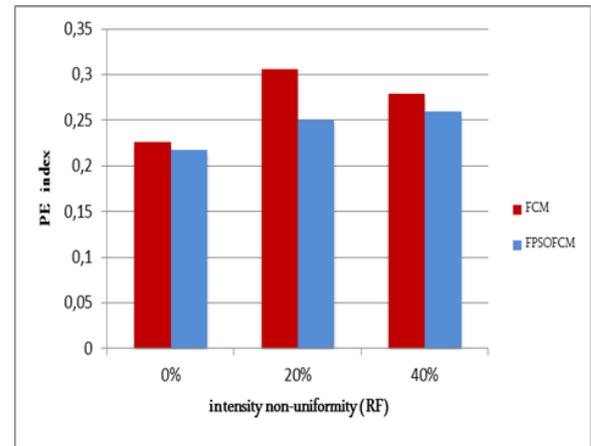
(c) Bruit= 5%

Figure 6.5: Comparaison de l'indice MPC sur des images simulées pondérées en T1.

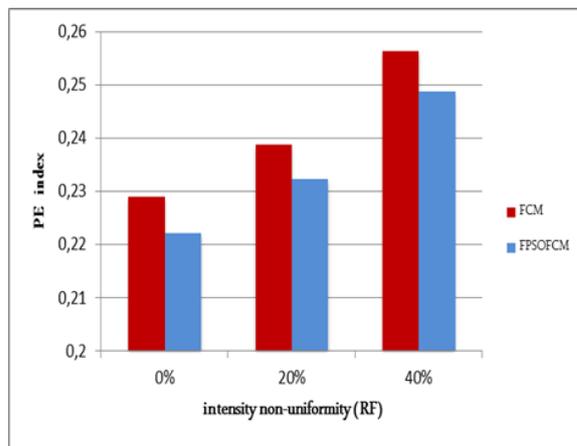
Figure 6.6: Comparaison de l'indice MPC sur des images simulées pondérées en DP.



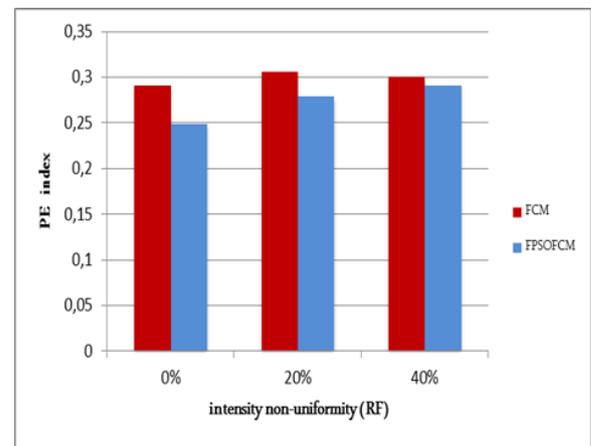
(a) Bruit= 0%



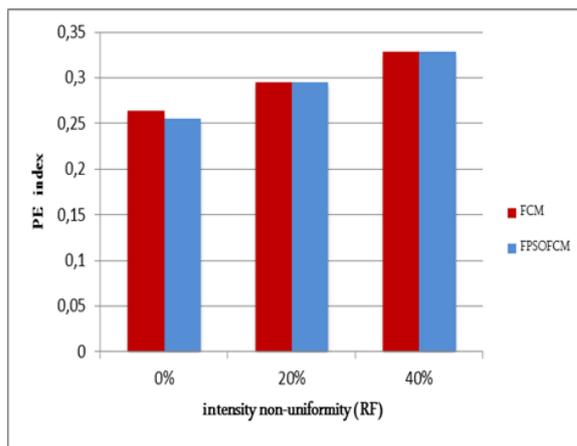
(a) Bruit= 0%



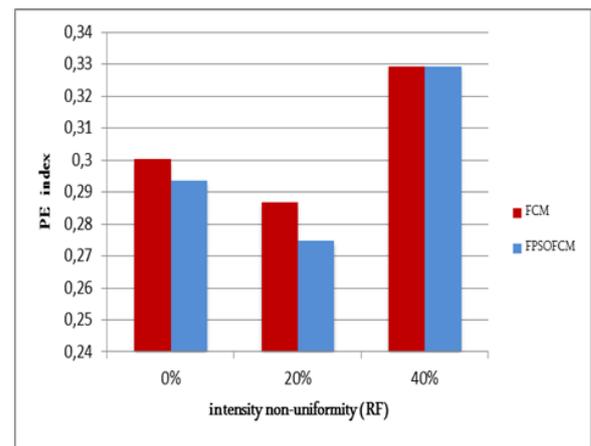
(b) Bruit= 3%



(b) Bruit= 3%



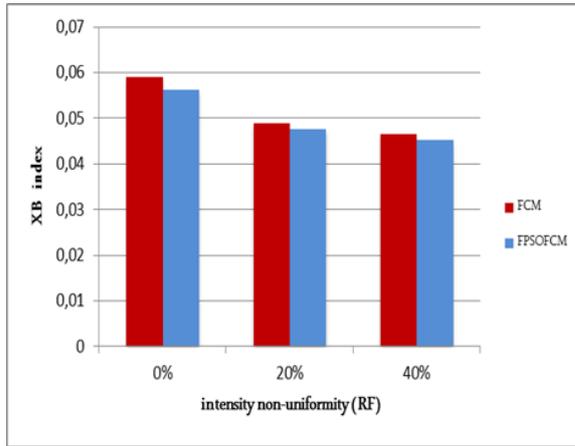
(c) Bruit= 5%



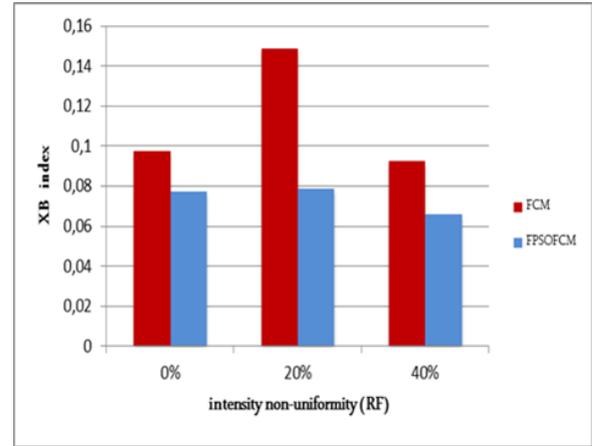
(c) Bruit= 5%

Figure 6.7: Comparaison de l'indice PE sur des images simulées pondérées en T1.

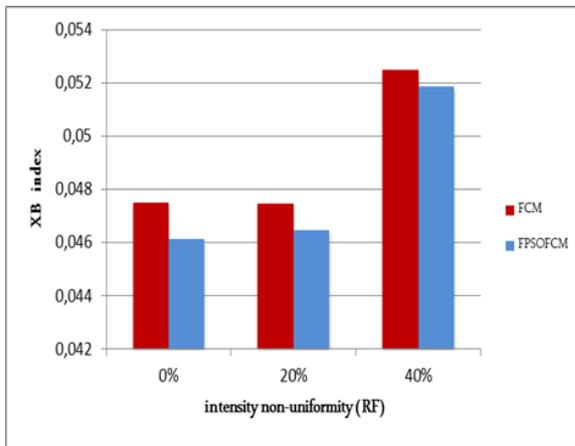
Figure 6.8: Comparaison de l'indice PE sur des images simulées pondérées en DP.



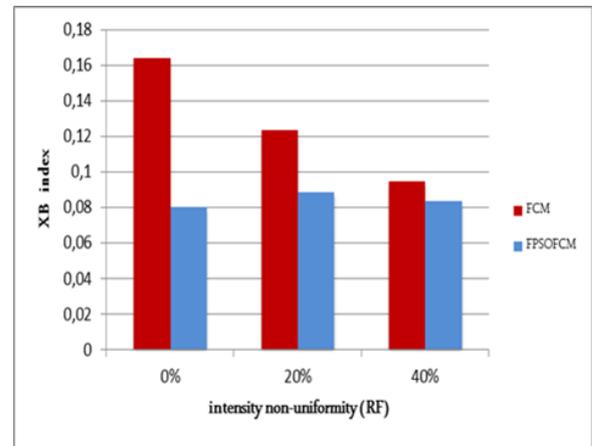
(a) Bruit= 0%



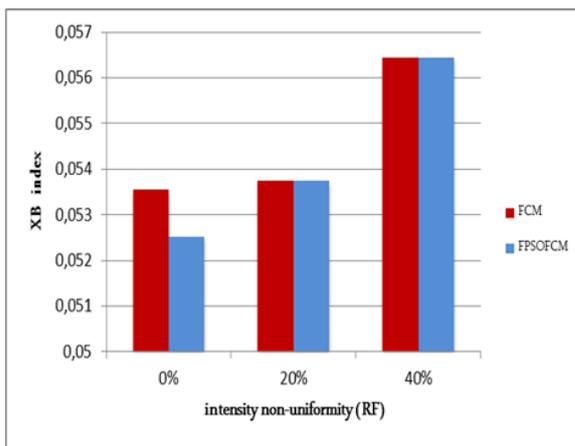
(a) Bruit= 0%



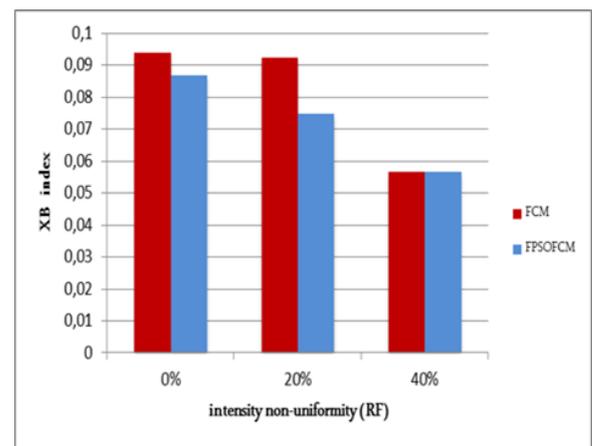
(b) Bruit= 3%



(b) Bruit= 3%



(c) Bruit= 5%



(c) Bruit= 5%

Figure 6.9: Comparaison de l'indice XB sur des images simulées pondérées en T1.

Figure 6.10: Comparaison de l'indice XB sur des images simulées pondérées en DP.

donne les meilleurs résultats pour différentes valeurs de bruit et d'inhomogénéité. Sauf le cas où les niveaux du bruit et d'inhomogénéité RF sont élevés, on remarque (Figures 6.3(c), 6.4(c), 6.5(c), 6.7(c), 6.9(c) et 6.10(c)) que la différence des performances entre FPSOFCM et FCM n'est pas significative, parce que la valeur 40% d'inhomogénéité RF, perturbe trop les distributions des niveaux de gris. Dans la réalité, l'inhomogénéité se situe plutôt aux environs de 10% [138]. D'autres résultats de test sur des images de coupes coronales sont présentés dans l'Annexe A.

Pour vérifier l'efficacité et la robustesse de notre approche contre le bruit impulsionnel, nous avons testé les algorithmes FCM et FPSOFCM sur un ensemble de 12 images IRM (axiales et coronales) simulées pondérées en T1 et DP, ces images sont affectées de différents niveaux de bruit Salt & pepper (3% 5%) et d'inhomogénéité RF (0% 20% 40%). Les résultats moyens obtenus (Annexe B) montrent que FPSOFCM permet d'avoir de bons résultats de classification, même à la présence de ce type de bruit.

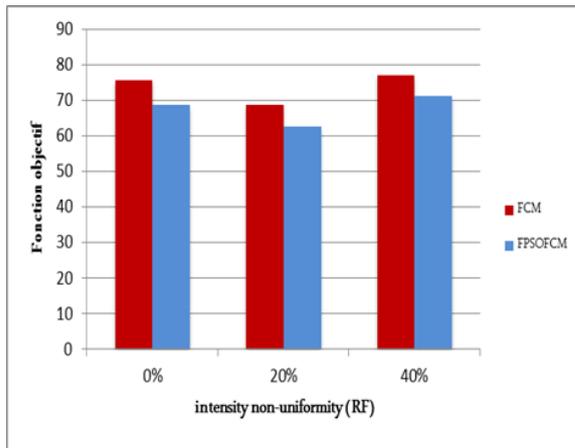
(b) Fonction objectif

Les Figures 6.11 et 6.12 montrent les valeurs moyennes de la fonction objectif (Eq. 2.8), après 10 exécutions indépendantes de FCM et FPSOFCM sur la même base d'images utilisée dans la simulation précédente (échantillon 1). Comme nous pouvons le remarquer sur ces figures, les résultats de FPSOFCM s'avèrent significatifs pour la majorité des images de test. Pour les cas où les niveaux de bruit et d'inhomogénéité RF sont élevés (Figures 6.11.(c) et 6.12.(c)), la méthode proposée donne de moins bon résultats que FCM standard.

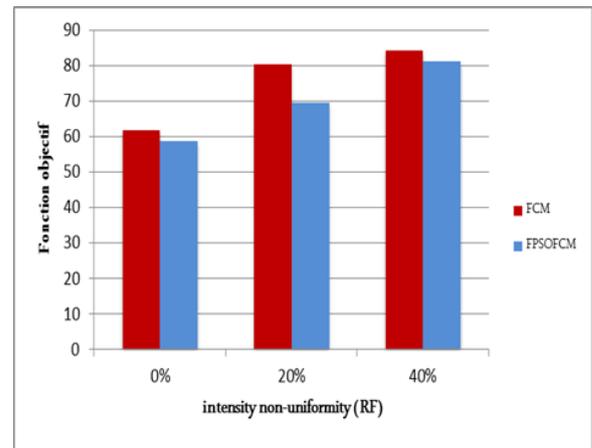
A titre comparatif, nous avons également étudié la convergence de la fonction objectif de FCM et FPSOFCM sur un échantillon de 6 images IRM simulées. Les figures 6.13 et 6.14 illustrent les résultats obtenus. Ces résultats montrent que l'algorithme FPSOFCM converge lentement par rapport à FCM. Mais, il atteint des valeurs minimales de la fonction objectif. Cela est dû principalement à la configuration initiale des centres de classes, qui est calculée par la méthode de recherche globale FPSO. Cette configuration ajustée, a permis à FPSOFCM d'éviter d'être bloqué dans des minima locaux. Les valeurs moyennes de la fonction objectif sur 18 images IRM de coupes coronales ainsi les comparaisons des convergences sont présentées dans l'Annexe A (Figures A.9 à A.12).

6.4.3.2 Evaluation visuelle

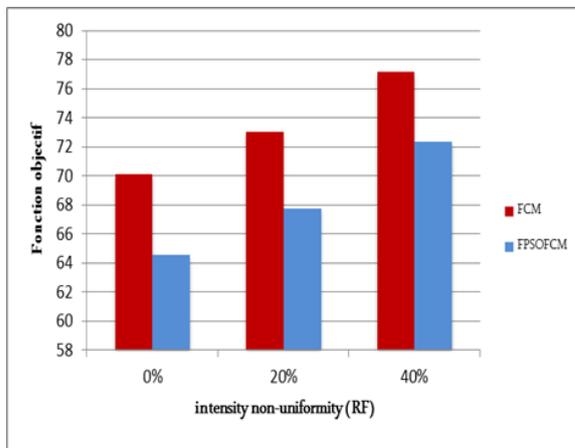
- **Résultats sur des images IRM (coupes Axiales)**



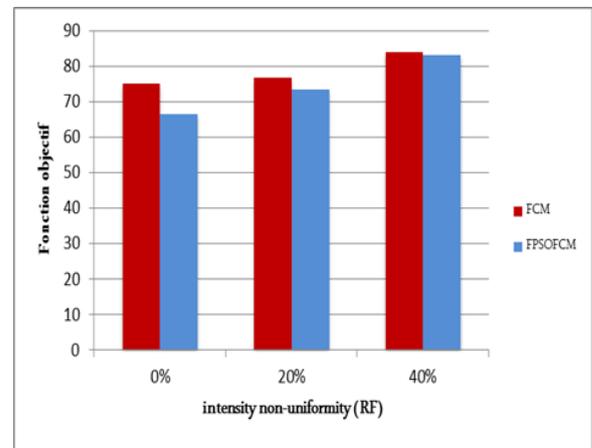
(a) Bruit= 0%



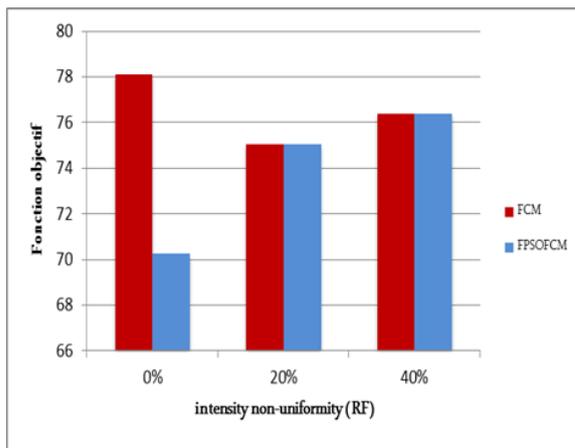
(a) Bruit= 0%



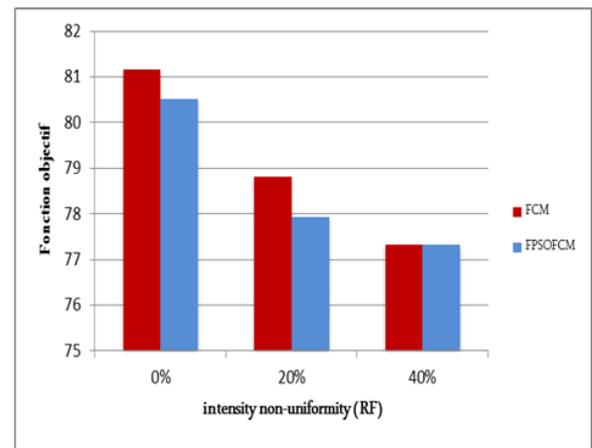
(b) Bruit= 3%



(b) Bruit= 3%



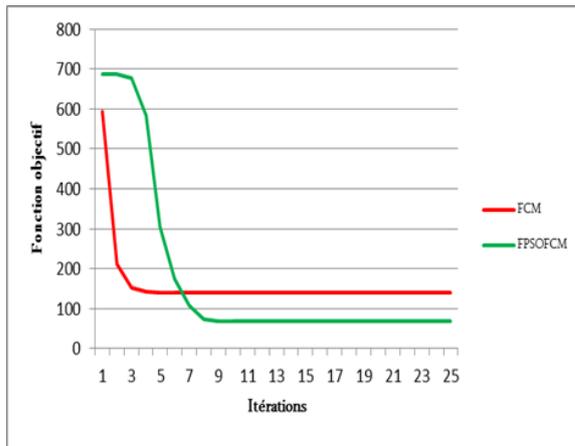
(c) Bruit= 5%



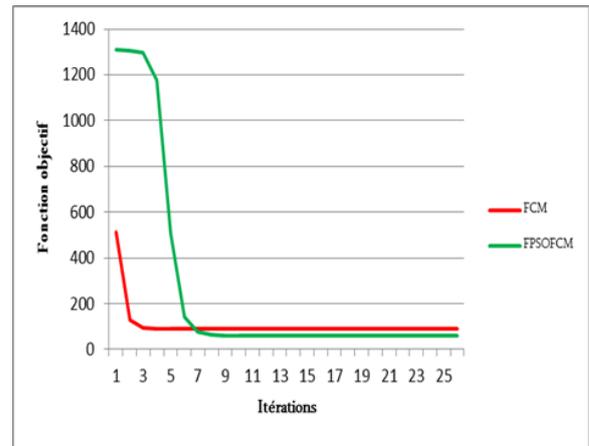
(c) Bruit= 5%

Figure 6.11: Comparaison de la fonction objectif moyenne sur des images simulées pondérées en T1.

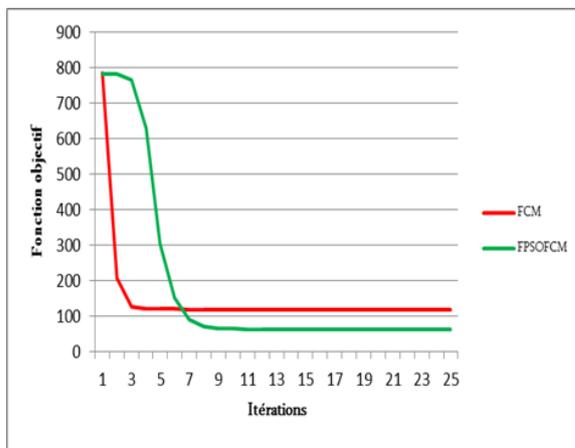
Figure 6.12: Comparaison de la fonction objectif moyenne sur des images simulées pondérées en DP.



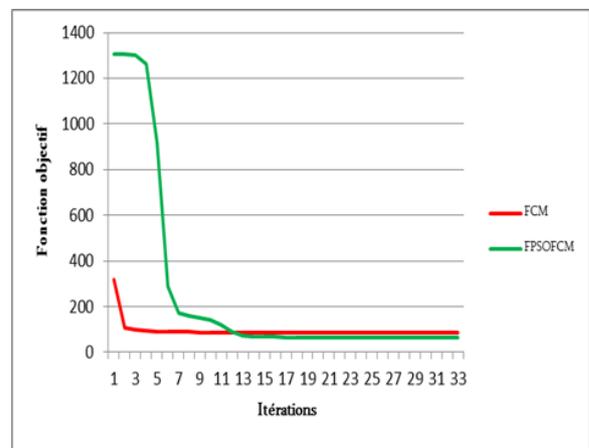
(a) Bruit= 0%



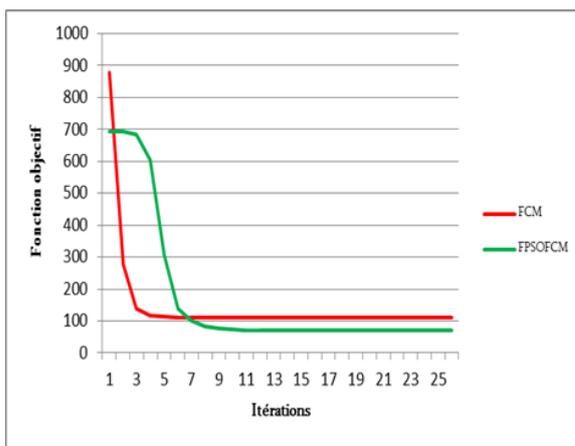
(a) Bruit= 0%



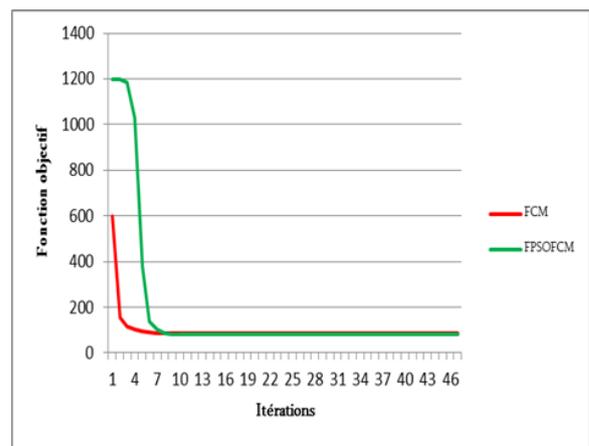
(b) Bruit= 3%



(b) Bruit= 3%



(c) Bruit= 5%



(c) Bruit= 5%

Figure 6.13: Convergence de FCM et FPSOFCM sur des images simulées pondérées en T1.

Figure 6.14: Convergence de FCM et FPSOFCM sur des images simulées pondérées en DP.

Les Figures 6.15 à 6.20 illustrent les résultats visuels de la segmentation d'images IRM pondérées en T1 (Figures 6.15 à 6.17) et en DP (Figures 6.18 à 6.20). Les images segmentées en 4 classes par FCM et FPSOFCM, sont représentées respectivement sur la deuxième et la troisième colonne de chaque figure.

Visuellement, on remarque que FPSOFCM permet de bien caractériser les différents tissus cérébraux (fond, LCR, MG et MB) contrairement à FCM. Dans les Figures 6.15(c), 6.15(f), 6.15(i), 6.17(i) et 6.17.(f) on remarque la visibilité de la classe LCR en rouge (invisible dans les Figures 6.15(b), 6.15(e), 6.15(h), 6.17(e) et 6.17.(h)). Si on compare les Figures 6.19(b), 6.19(e), 6.20(b) et 6.20.(e), on remarque que l'algorithme FCM a classé une partie de la matière blanche comme étant appartenant au liquide céphalorachidien, contrairement aux classes obtenues par FPSOFCM .

D'un point de vue visuel, les résultats obtenus par FPSOFCM sont de bonne qualité, même s'ils sont similaires à ceux de FCM pour un bruit de 5% et une inhomogénéité RF de 40%. Comme nous l'avons déjà mentionné, la valeur 40% d'inhomogénéité RF, produit des images de nature très artificielle. Sur ce type d'images, il est très difficile de comparer les performances des méthodes de segmentation.

- **Résultats sur des images IRM réelles pathologiques (coupes Axiales)**

La Figure 6.21 donne des exemples de segmentation de 3 images réelles par FCM et FPSOFCM. Dans ce cas, le nombre de classes C est égal à 5, la pathologie est considérée comme une 5 ème classe (couleur blanche). D'après cette figure, il est difficile de tirer des conclusions visuelles précises sur ce type d'images.

Les Figures 6.22 à 6.24 illustrent les résultats de la segmentation de 3 images IRM tumorales [151, 185]. Dans ce cas, le nombre de classes C est égal à 5. (Fond, LCR, MG, MB et tumeur). Sur ces figures, nous observons que la segmentation des différents tissus cérébraux ainsi que la tumeur est satisfaisante.

6.4.3.3 Evaluation quantitative de la segmentation

Dans la segmentation d'images, la validation des résultats obtenus est un problème délicat, où plusieurs approches peuvent être envisagées dans le but de fournir une évaluation quantitative et qualitative. Nous distinguerons principalement, les approches exploitant les résultats réalisés manuellement par des experts sur des images réelles, et les approches exploitant des images de synthèse pour lesquelles les références sont parfaitement connues.

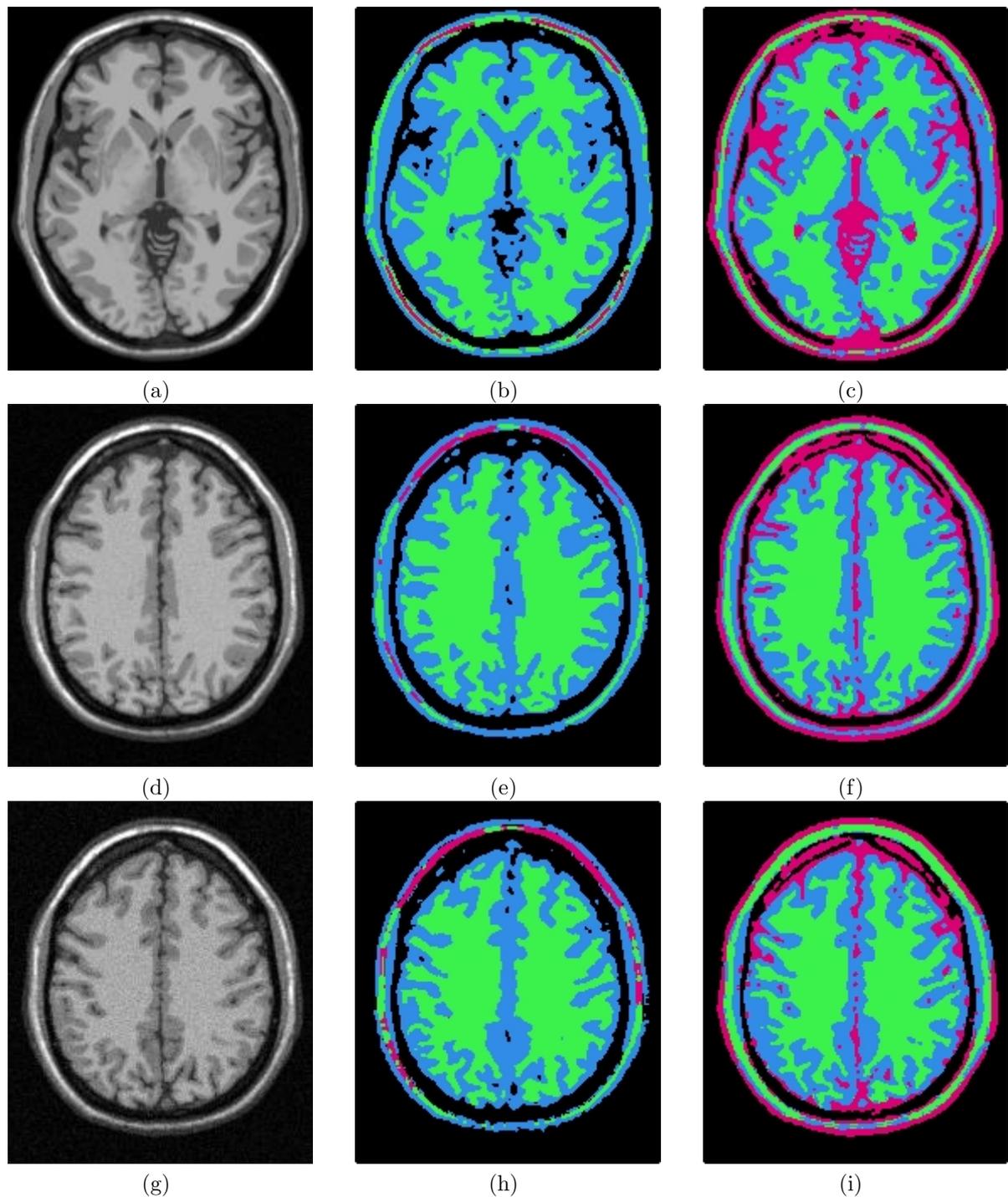


Figure 6.15: Résultats de segmentation (4 classes) sur des images pondérées en T1 (Inhomogénéité $R_f=0\%$). (a) (d) (g) : images originales pondérées en T1, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

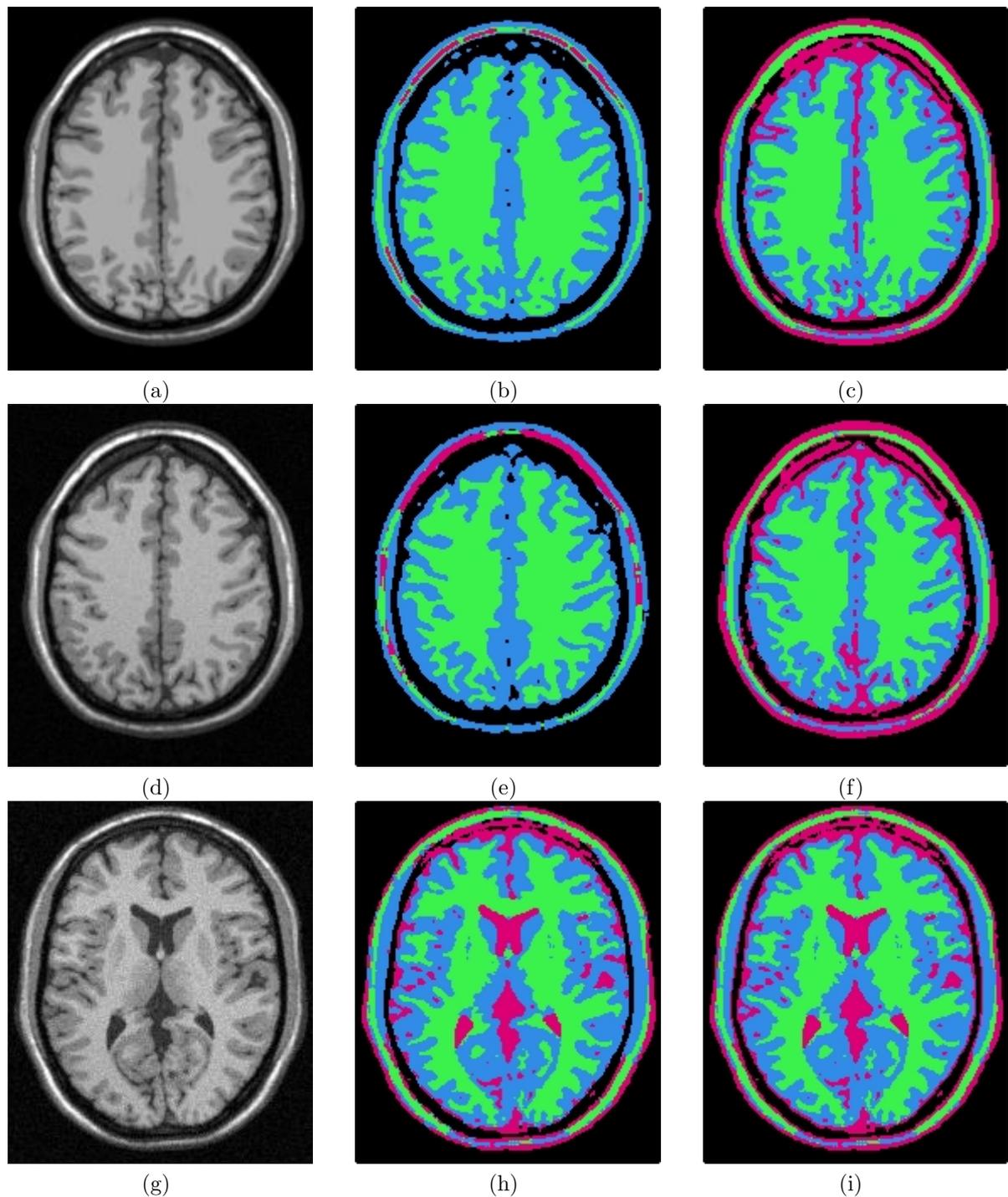


Figure 6.16: Résultats de segmentation (4 classes) sur des images pondérées en T1 (Inhomogénéité $R_f=20\%$). (a) (d) (g) : images originales pondérées en T1, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

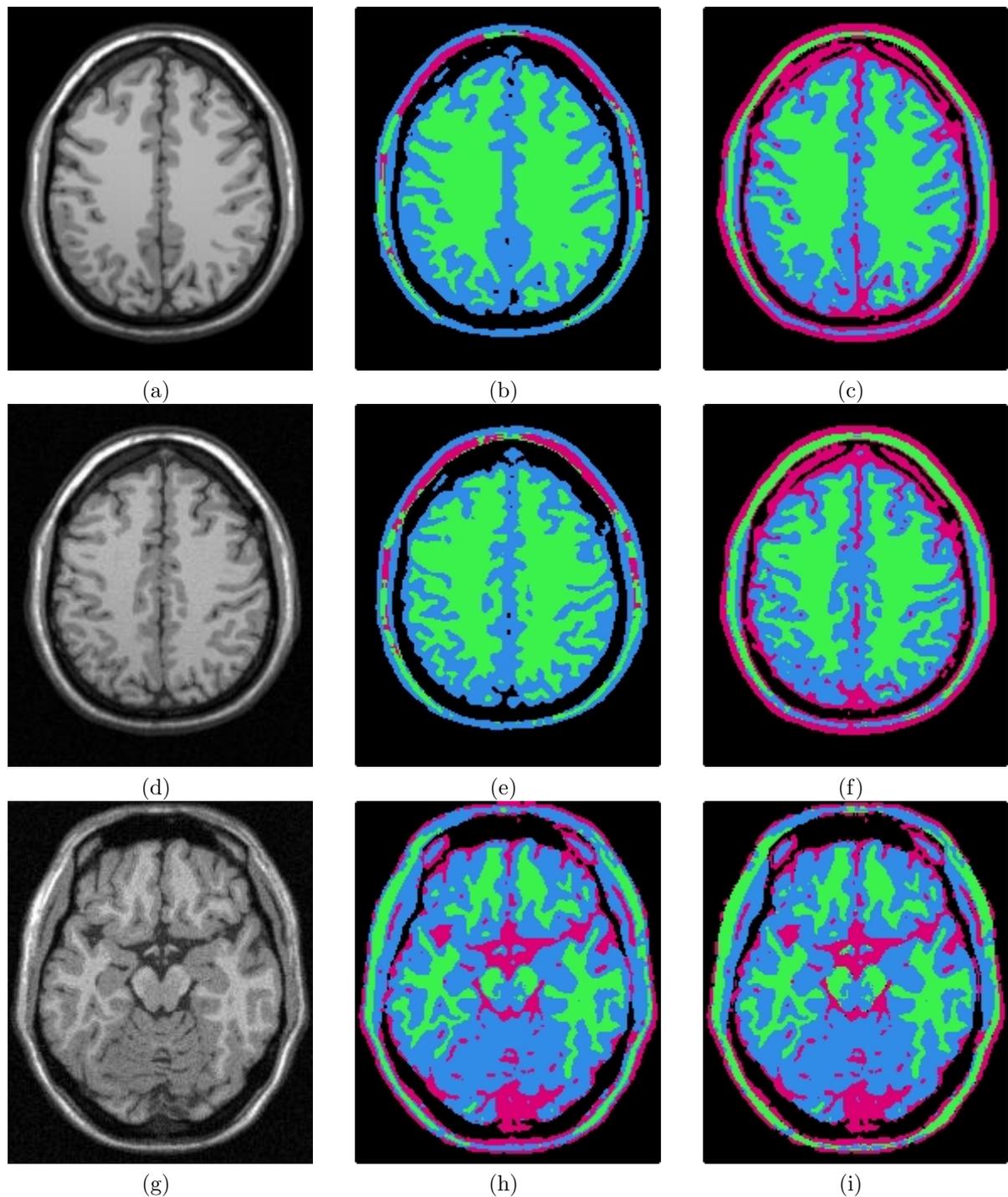


Figure 6.17: Résultats de segmentation (4 classes) sur des images pondérées en T1 (Inhomogénéité $R_f=40\%$). (a) (d) (g) : images originales pondérées en T1, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

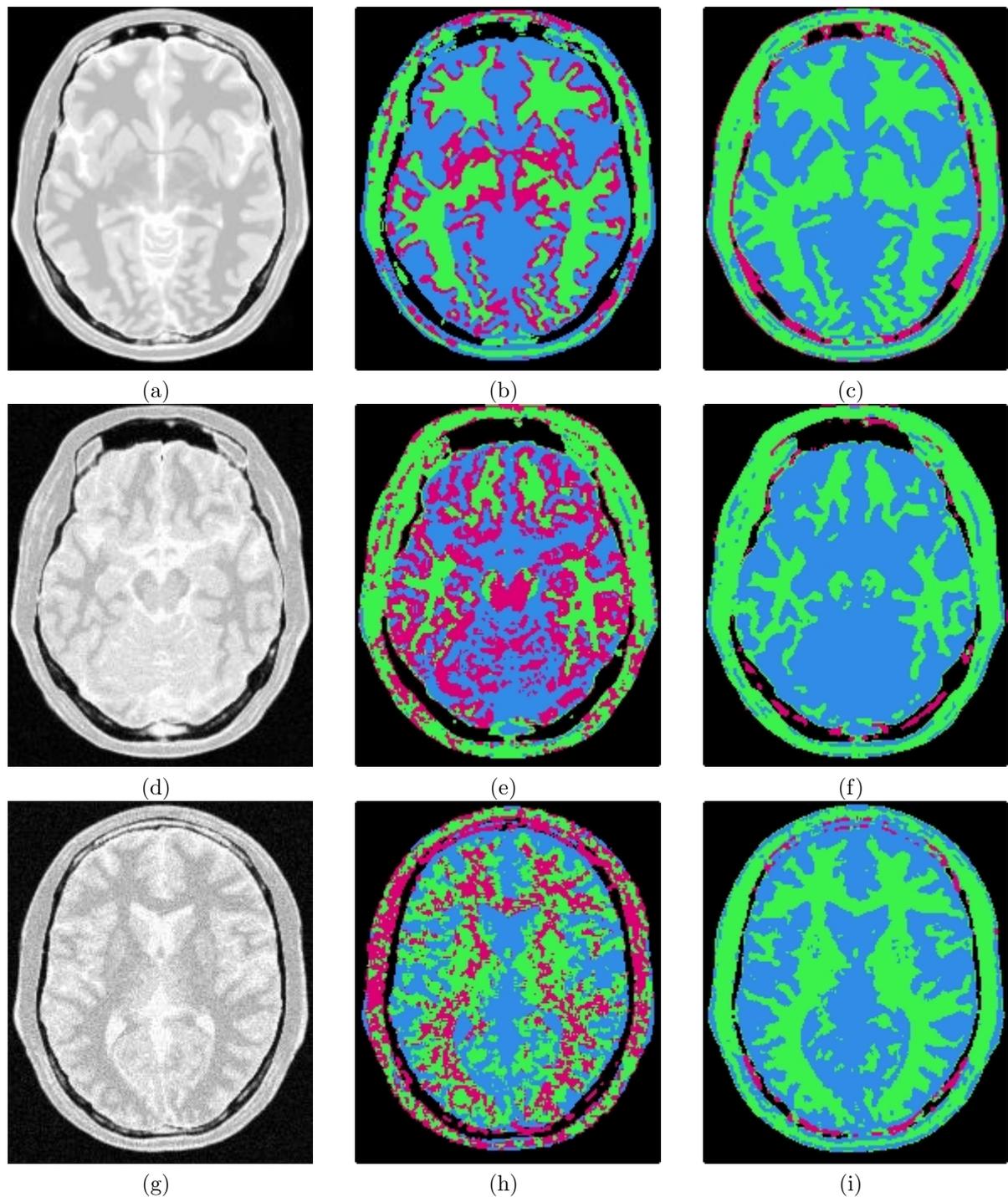


Figure 6.18: Résultats de segmentation (4 classes) sur des images pondérées en DP (Inhomogénéité $R_f=0\%$). (a) (d) (g) : images originales pondérées en DP, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

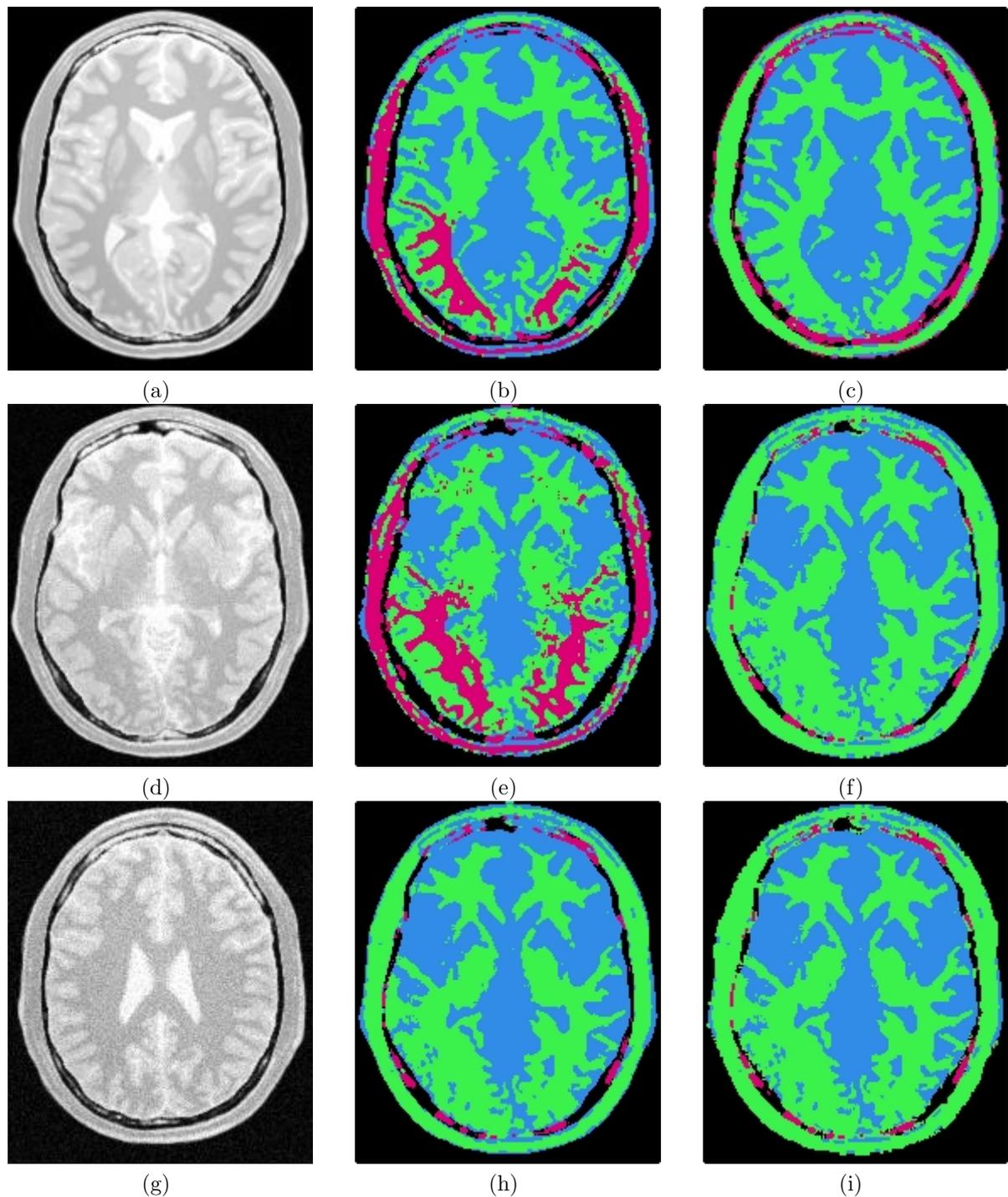


Figure 6.19: Résultats de segmentation (4 classes) sur des images pondérées en DP (Inhomogénéité $R_f=20\%$). (a) (d) (g) : images originales pondérées en DP, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

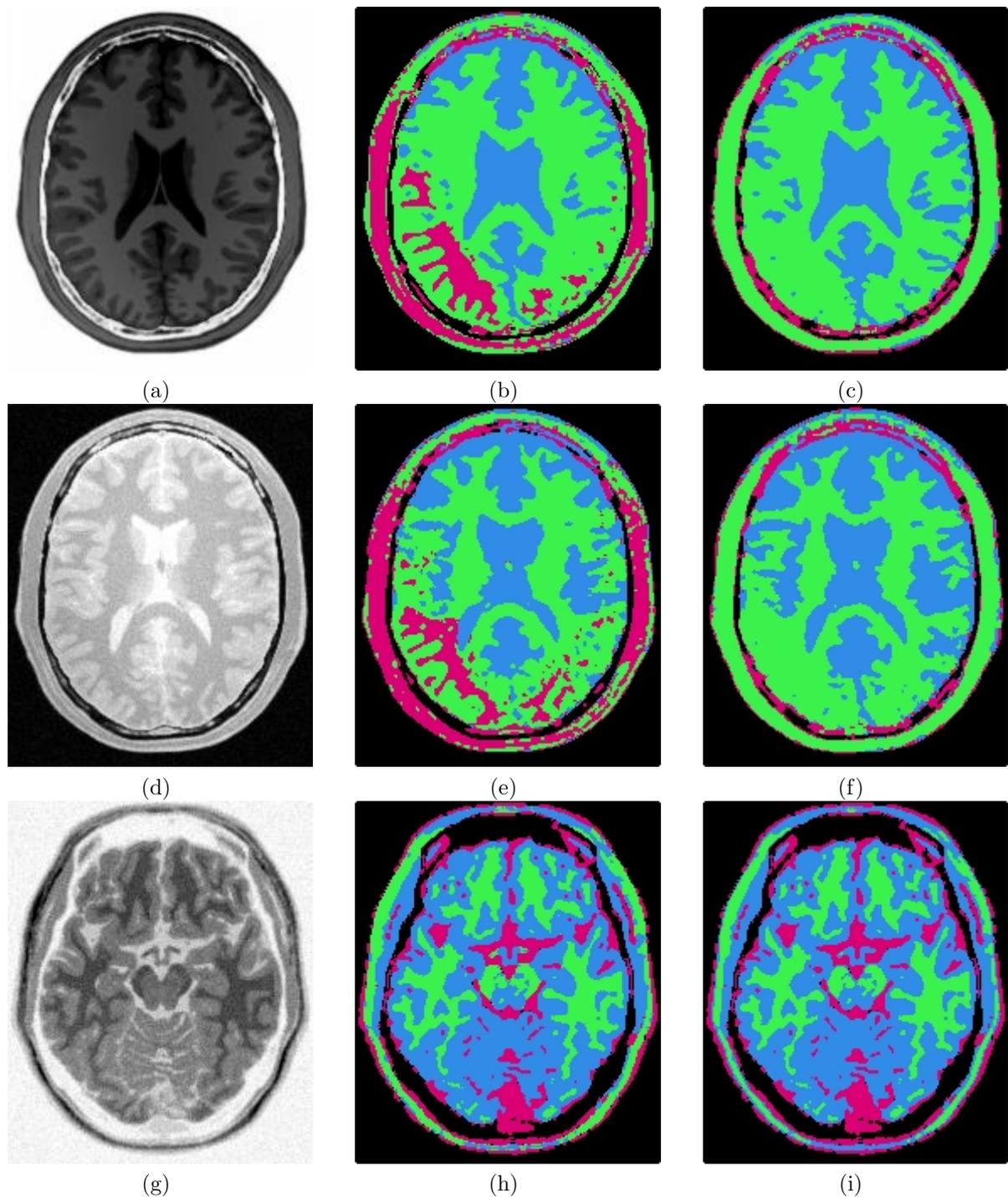


Figure 6.20: Résultats de segmentation (4 classes) sur des images pondérées en DP (Inhomogénéité $R_f=40\%$). (a) (d) (g) : images originales pondérées en DP, bruitées avec un bruit Gaussien respectivement (0% 3% 5%). (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

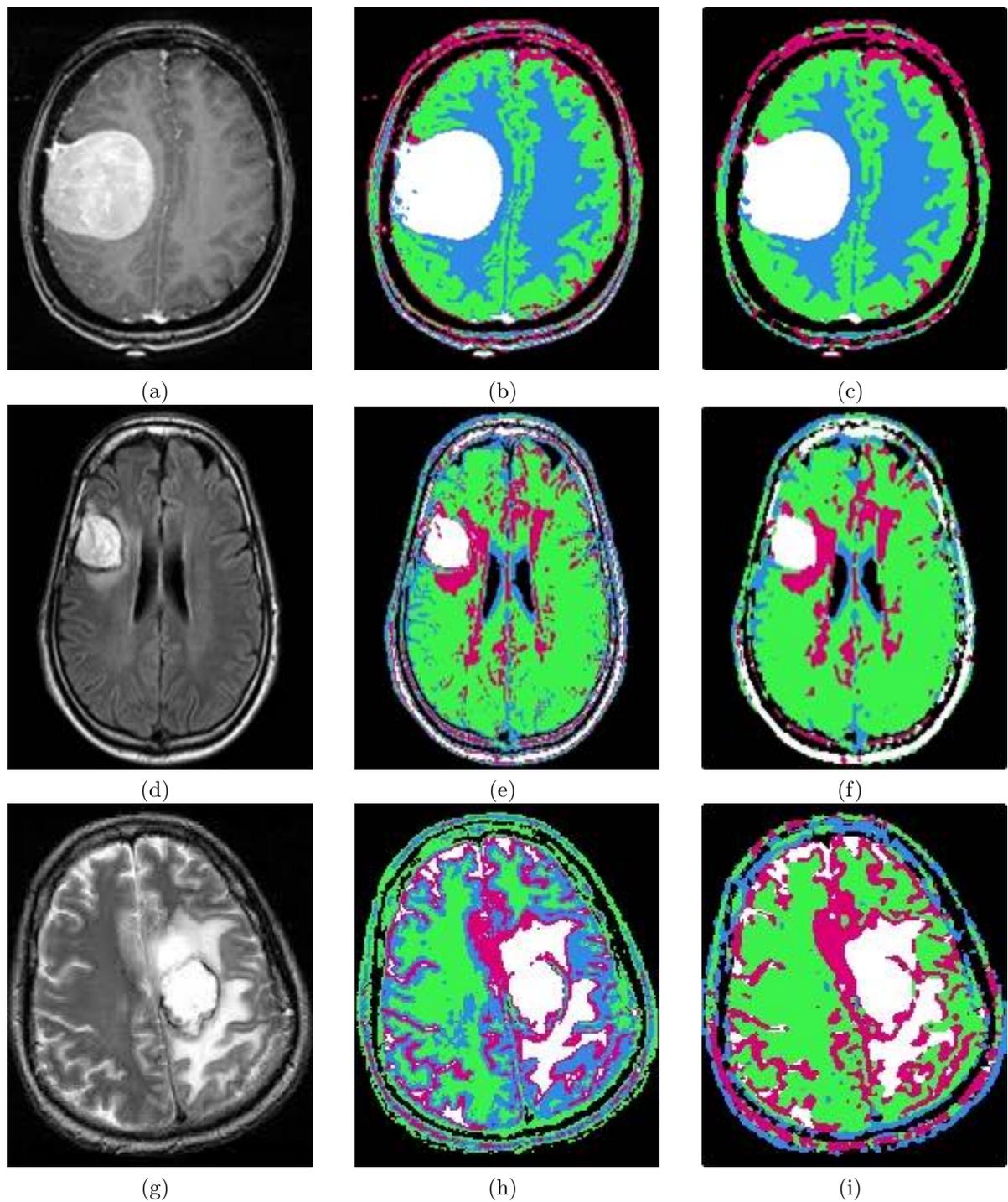


Figure 6.21: Résultats de segmentation (5 classes) sur des images réelles. (a) (d) (g) : images originales. (b) (e) (h) : résultats FCM. (c) (f) (i) : résultats FPSOFCM.

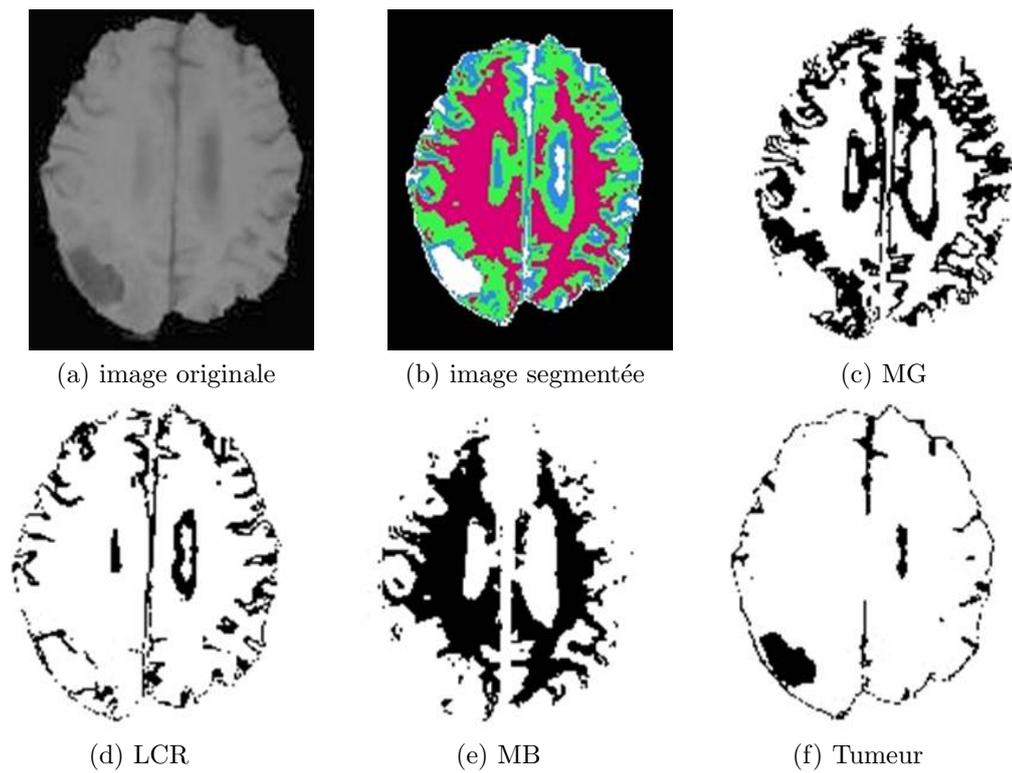


Figure 6.22: Séparation des classes segmentées image 1.

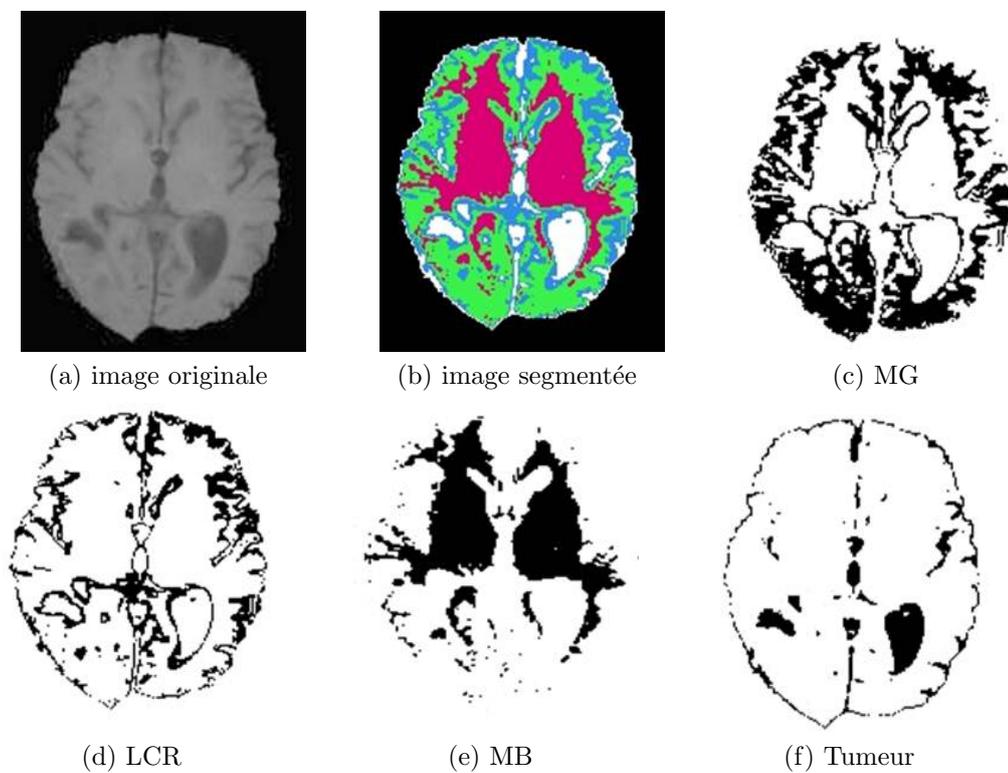


Figure 6.23: Séparation des classes segmentées image 2.

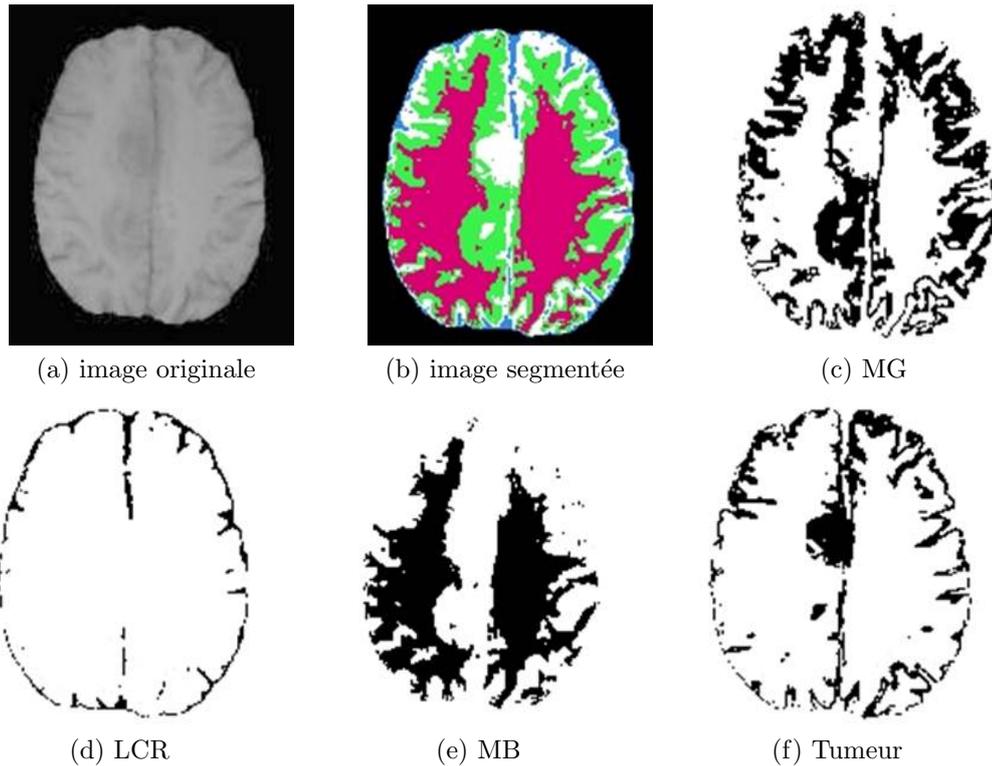


Figure 6.24: Séparation des classes segmentées image 3.

Dans cette section, nous allons comparer quantitativement les résultats de FCM et FPSOFCM avec des images de référence (*ground truth*). La comparaison est faite pour les deux principales substances cérébrales, à savoir : la matière grise (MG) et la matière blanche (MB).

On considère A, une région de l'image de référence, B une région de l'image segmentée.

Les pixels sont classés en 4 catégories :

- Vrais Positifs (VP) : le nombre de pixels présents dans A et B
- Faux Positifs (FP) : le nombre de pixels présents dans B et pas dans A
- Vrais Négatifs (VN) : le nombre de pixels qui ne sont présents ni dans A ni dans B
- Faux Négatifs (FN) : le nombre de pixels présents dans la A et pas dans B

Nous avons choisi les critères d'évaluation les plus utilisés dans le domaine de la segmentation d'images, à savoir :

1. Sensibilité (SE)

Définie comme la proportion des vrais positifs parmi l'ensemble des structures qui devraient être segmentées. Plus la sensibilité est proche de 1, moins il y a d'erreurs de détection.

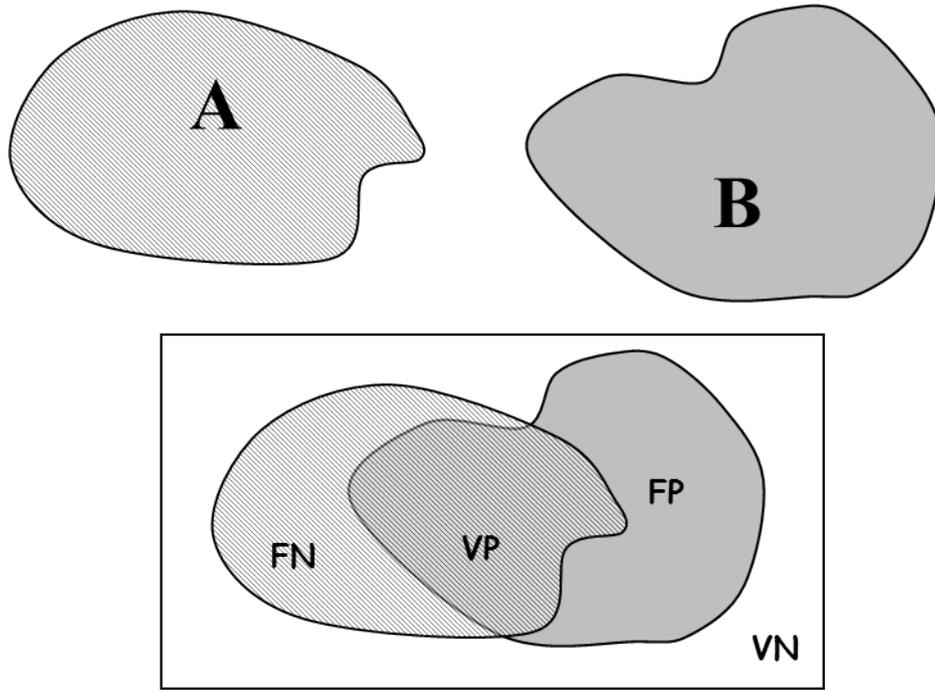


Figure 6.25: Comparaison de l'image de référence (A) à l'image segmentée (B).

$$SE = \frac{VP}{VP + FN} \quad (6.15)$$

2. Spécificité (SP)

Définie comme la proportion de vrais négatifs parmi l'ensemble des structures qui ne devraient pas être segmentées. Plus la spécificité est proche de 1, moins il y a de faux positifs.

$$SP = \frac{VN}{VN + FP} \quad (6.16)$$

3. Recouvrement (Jaccard coefficient)

Défini comme la mesure de recouvrement entre l'image segmentée et l'image de référence. Plus le recouvrement est proche de 1, moins il y a de faux positifs et de faux négatifs.

$$RE = \frac{VP}{VP + FP + FN} \quad (6.17)$$

4. Similarité (SI)

Tout comme le recouvrement, la similarité correspond à la mesure de recouvrement entre l'image segmentée et l'image de référence. La similarité vaut 1, si les deux segmentations (A et B) sont identiques et 0 lorsqu'il n'y a pas de similarité.

$$SI = \frac{2VP}{2VP + FP + FN} \quad (6.18)$$

5. Erreur (ERR)

C'est l'erreur de classification des pixels. Dans le cas idéal, elle vaut 0.

$$ERR = \frac{FP + FN}{N} \quad (6.19)$$

Les Tableaux 6.5 et 6.6 présentent les résultats quantitatifs de segmentation de la matière grise et la matière blanche, obtenus par FCM et FPSOFCM sur les images IRM simulées de l'échantillon 5 (Tableau C.5). Les meilleurs résultats, sont indiqués en gras.

D'après ces tableaux, nous pouvons constater que l'algorithme proposé FPSOFCM obtient une sensibilité élevée pour la matière grise, ce qui signifie que cette matière est bien caractérisée comme étant une matière grise. Pour la matière blanche, la sensibilité est faible avec FPSOFCM pour 4 images (images 2, 3, 4, 8), mais la spécificité est élevée. En fait, il est difficile de juger la qualité de la segmentation, si ces deux indices quantitatifs (sensibilité, spécificité) sont contradictoires. Pour les autres indices (RE, SI et ERR), on remarque que notre approche FPSOFCM a donné de bons résultats pour toutes les images segmentées. Le premier point que l'on peut remarquer, est la différence forte entre les taux de recouvrement (RE) obtenus par FCM et FPSOFCM sur les images (2, 3 et 6). Un second point à mentionner, concerne le taux faible des pixels mal classés (ERR) et la similarité élevée, obtenus par FPSOFCM pour toutes les images de test. Ces résultats montrent que FPSOFCM est plus robuste vis-à-vis le taux de bruit et d'inhomogénéité RF que l'algorithme standard FCM.

Pour bien visualiser les taux de recouvrement(RE) ainsi l'erreur (ERR) de FCM et FPSOFCM, nous avons transformé les valeurs numériques en histogrammes (Figures 6.26 à 6.29).

6.4.3.4 Comparaison de FPSOFCM avec FPSO

Afin d'évaluer les performances de notre approche avec d'autres méthodes existantes dans la littérature. Les résultats de segmentation obtenus par FPSOFCM, ont été comparés à ceux fournis par la variante FPSO proposée dans [179].

Les résultats moyens des indices (PC, MPC, PE, XB) après 10 exécutions indépendantes de FPSOFCM et FPSO [179], sont décrits dans le Tableau 6.7.

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.789	0.836	51.226	0.677	0.175
	FPSOFCM	0.927	0.918	72.948	0.844	0.080
MB	FCM	1.000	0.903	73.297	0.846	0.077
	FPSOFCM	0.977	0.915	74.120	0.851	0.072

(a) image 1

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.802	0.869	49.713	0.664	0.143
	FPSOFCM	0.942	0.940	73.642	0.848	0.059
MB	FCM	0.994	0.924	75.816	0.862	0.062
	FPSOFCM	0.942	0.950	78.172	0.877	0.052

(b) image 2

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.793	0.882	51.076	0.676	0.134
	FPSOFCM	0.937	0.935	71.820	0.836	0.065
MB	FCM	0.996	0.919	74.871	0.856	0.065
	FPSOFCM	0.974	0.945	79.398	0.885	0.049

(c) image 3

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.882	0.878	68.785	0.815	0.121
	FPSOFCM	0.908	0.867	69.381	0.819	0.121
MB	FCM	0.950	0.930	56.034	0.718	0.068
	FPSOFCM	0.947	0.939	59.062	0.743	0.060

(d) image 4

Tableau 6.5: Résultats quantitatifs de segmentation.

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.447	0.877	32.640	0.492	0.229
	FPSOFCM	0.922	0.828	60.591	0.755	0.149
MB	FCM	0.817	0.856	52.215	0.686	0.152
	FPSOFCM	0.989	0.826	58.815	0.741	0.140

(a) image 5

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.501	0.853	34.427	0.512	0.232
	FPSOFCM	0.843	0.815	53.510	0.697	0.178
MB	FCM	0.444	0.902	31.528	0.479	0.187
	FPSOFCM	0.980	0.789	52.071	0.685	0.174

(b) image 6

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.543	0.673	20.383	0.339	0.348
	FPSOFCM	0.623	0.874	37.947	0.550	0.167
MB	FCM	0.816	0.784	48.893	0.657	0.208
	FPSOFCM	0.979	0.749	55.074	0.710	0.195

(c) image 7

		SE	SP	RE(%)	SI	ERR
MG	FCM	0.873	0.877	67.790	0.808	0.124
	FPSOFCM	0.896	0.890	71.351	0.833	0.108
MB	FCM	0.943	0.925	57.289	0.728	0.073
	FPSOFCM	0.939	0.934	59.732	0.748	0.066

(d) image 8

Tableau 6.6: Résultats quantitatifs de segmentation.

Tableau 6.7: Comparaison des performances de FPSOFCM et FPSO [179].

Bruit	3%		5%		
	RF	FPSO	FPSOFCM	FPSO	FPSOFCM
0%	<i>PC</i>	0.852393	0.859673	0.839129	0.852231
		(0.007037)	(0.000004)	(0.012236)	(0.000004)
	<i>MPC</i>	0.803191	0.812897	0.785505	0.802975
		(0.009383)	(0.000005)	(0.016314)	(0.000006)
<i>PE</i>	0.290696	0.274941	0.315613	0.289826	
	(0.013818)	(0.000007)	(0.022977)	(0.000008)	
<i>XB</i>	0.063255	0.052187	0.072418	0.053955	
	(0.009046)	(0.000003)	(0.023322)	(0.000004)	
20%	<i>PC</i>	0.860577	0.877007	0.826756	0.840574
		(0.007705)	(0.000000)	(0.012167)	(0.000006)
	<i>MPC</i>	0.814103	0.836010	0.769008	0.787432
		(0.010273)	(0.000001)	(0.016223)	(0.000007)
<i>PE</i>	0.271125	0.239581	0.337334	0.311252	
	(0.014082)	(0.000001)	(0.022356)	(0.000011)	
<i>XB</i>	0.062880	0.051877	0.082125	0.055119	
	(0.011396)	(0.000001)	(0.031562)	(0.000005)	
40%	<i>PC</i>	0.863840	0.873514	0.834471	0.840085
		(0.016877)	(0.000004)	(0.006332)	(0.000005)
	<i>MPC</i>	0.818453	0.831352	0.779295	0.786780
		(0.022503)	(0.000005)	(0.008442)	(0.000007)
<i>PE</i>	0.269730	0.248942	0.323134	0.311292	
	(0.031892)	(0.000008)	(0.012311)	(0.000008)	
<i>XB</i>	0.056612	0.039534	0.061926	0.053458	
	(0.035061)	(0.000003)	(0.006750)	(0.000003)	

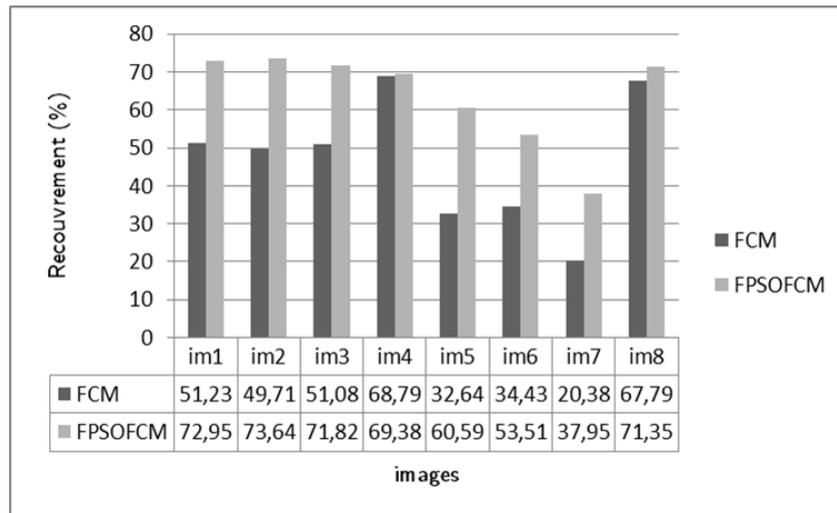


Figure 6.26: Taux de recouvrement de la matière grise.

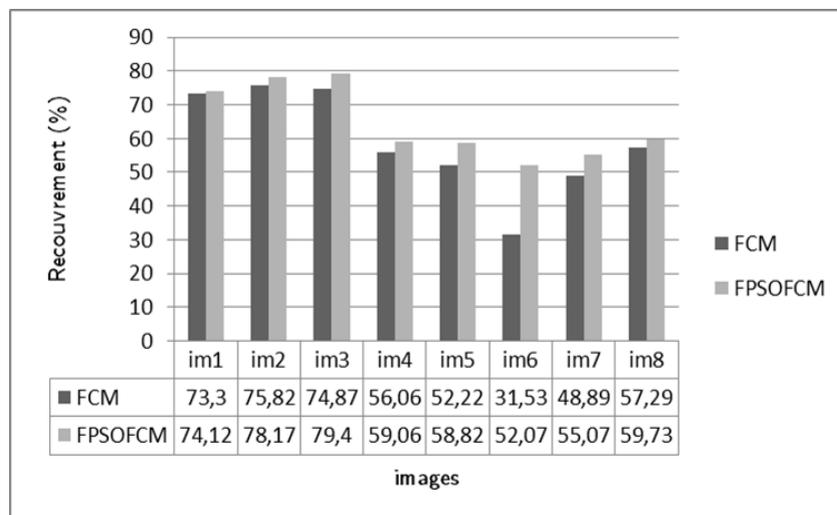


Figure 6.27: Taux de recouvrement de la matière blanche.

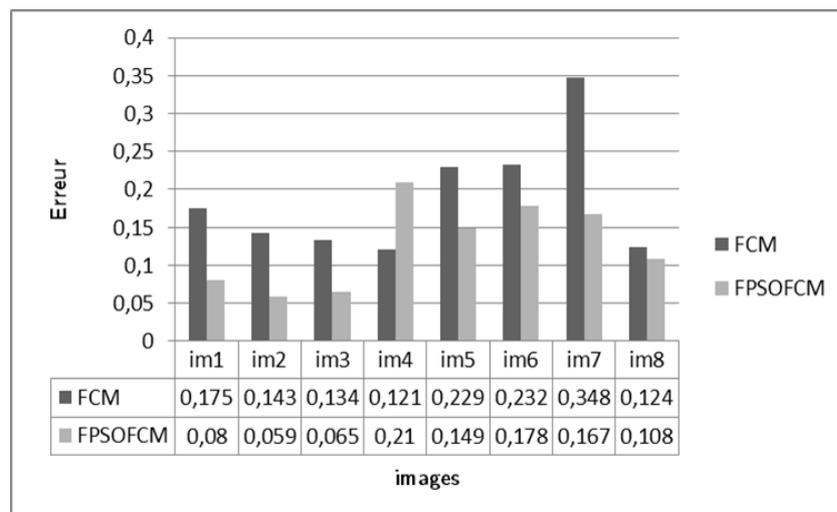


Figure 6.28: Erreur de classification de la matière grise.

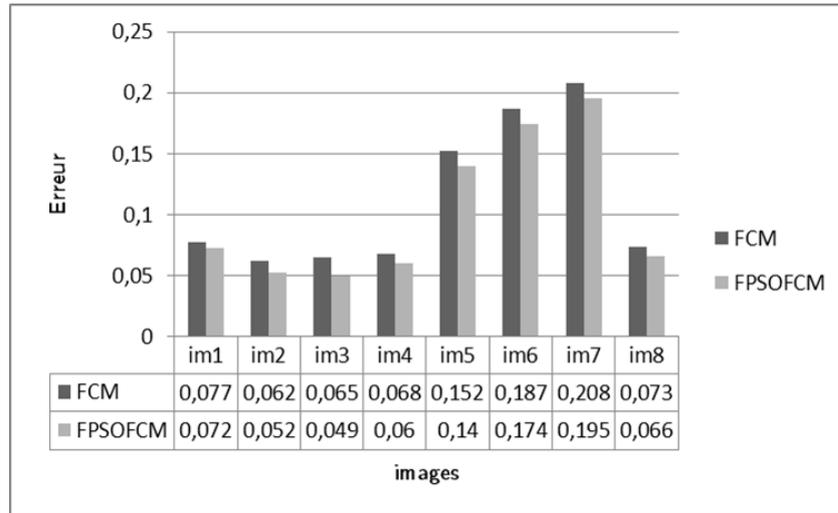


Figure 6.29: Erreur de classification de la matière blanche.

Les deux algorithmes sont exécutés avec les mêmes paramètres utilisés par H. Izakian et A. Abraham dans [179], $c_1 = c_2 = 2$, $m = 2$, $w = 0.9 \rightarrow 0.1$, nombre de particule $P = 10$. Les meilleurs résultats sont en gras. L'écart type (entre parenthèses) est calculé pour comparer la stabilité des deux algorithmes. Les images de test utilisées concernent l'échantillon 6 (Tableau C.6).

Les résultats du Tableau 6.7 donnent l'avantage à notre algorithme FPSOFCM par rapport à FPSO [179], pour toutes les valeurs de bruit (3%, 5%) et d'inhomogénéité (20%, 40%). En effet, l'initialisation ajustée des centres de classes de FCM a permis d'obtenir des valeurs élevées de PC, MPC et des valeurs faibles pour PE et XB. Donc, le FPSOFCM a généré des classes compactes et clairement séparées.

Les Figures 6.30 à 6.35 représentent des histogrammes comparatifs pour le taux de recouvrement, la similarité et l'erreur de classification. Ces figures montrent bien les performances de notre approche FPSOFCM par rapport à FPSO. En effet, le taux de recouvrement est amélioré jusqu'à 9% (image 4) pour les deux matières. La similarité est aussi améliorée, ce qui prouve que les classes obtenues par FPSOFCM se rapprochent des classes de références avec une erreur de classification (ERR) faible. Ces résultats illustrent également l'efficacité et la robustesse de FPSOFCM vis-à-vis le taux de bruit et d'inhomogénéité RF.

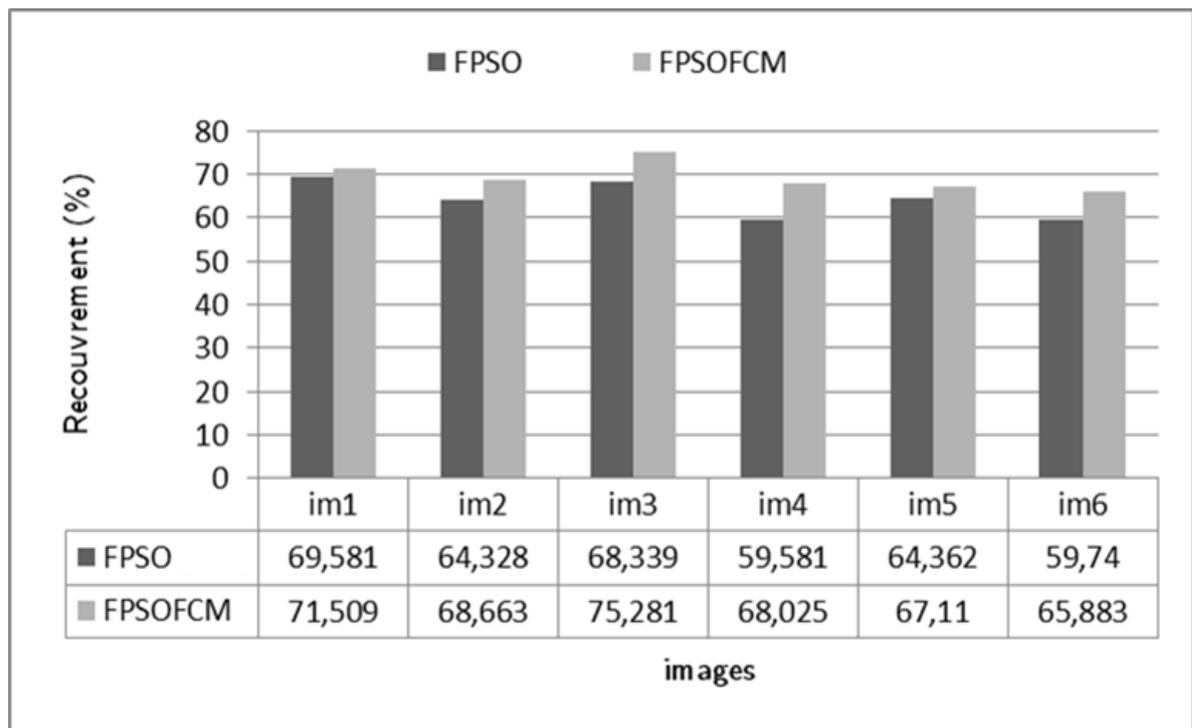


Figure 6.30: Taux de recouvrement de la matière grise.

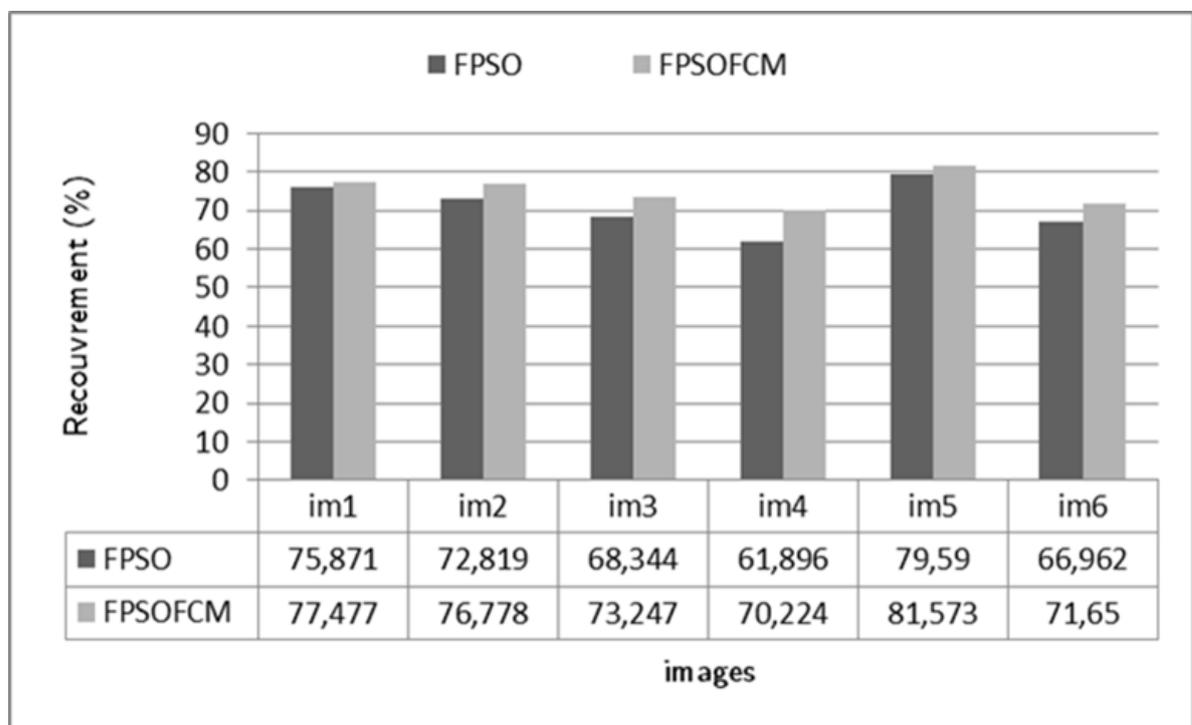


Figure 6.31: Taux de recouvrement de la matière blanche.

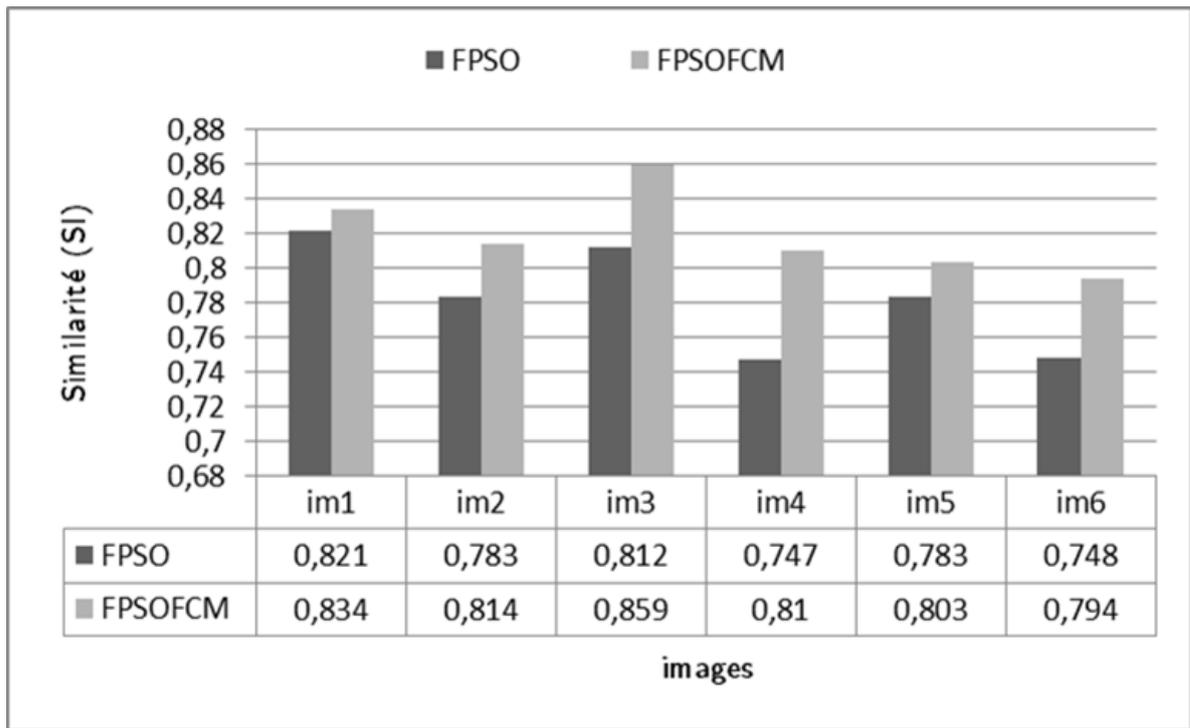


Figure 6.32: Indice de similarité de la matière grise.

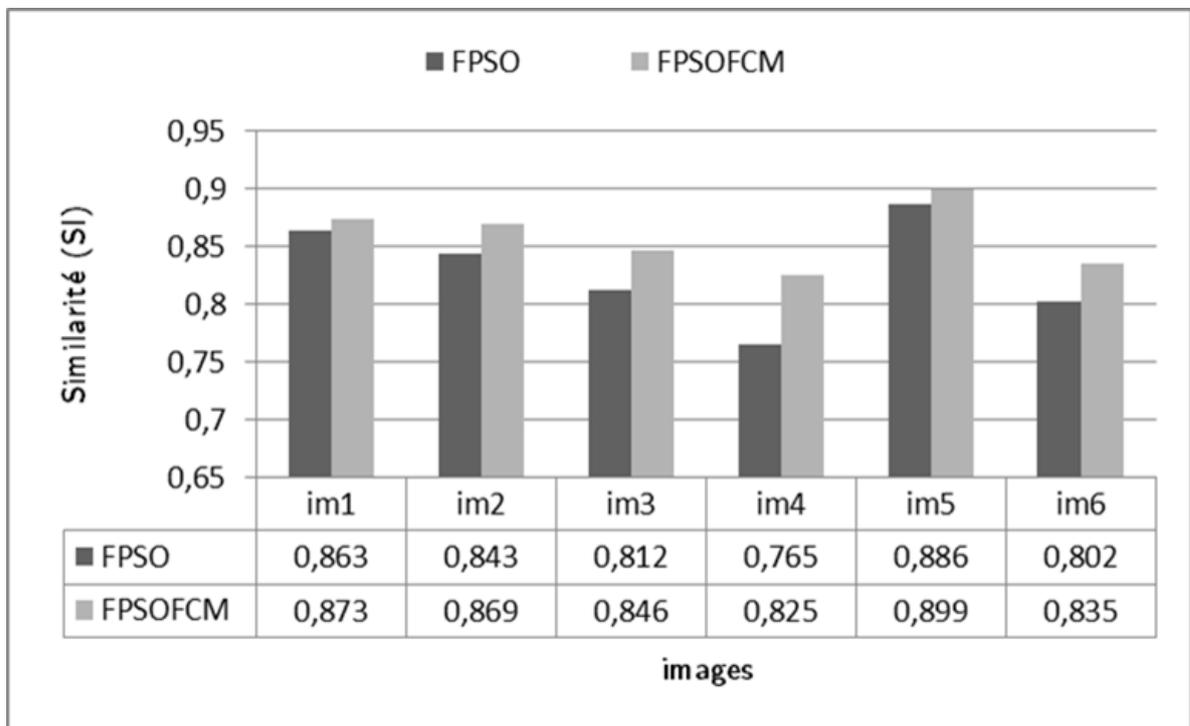


Figure 6.33: Indice de similarité de la matière blanche.

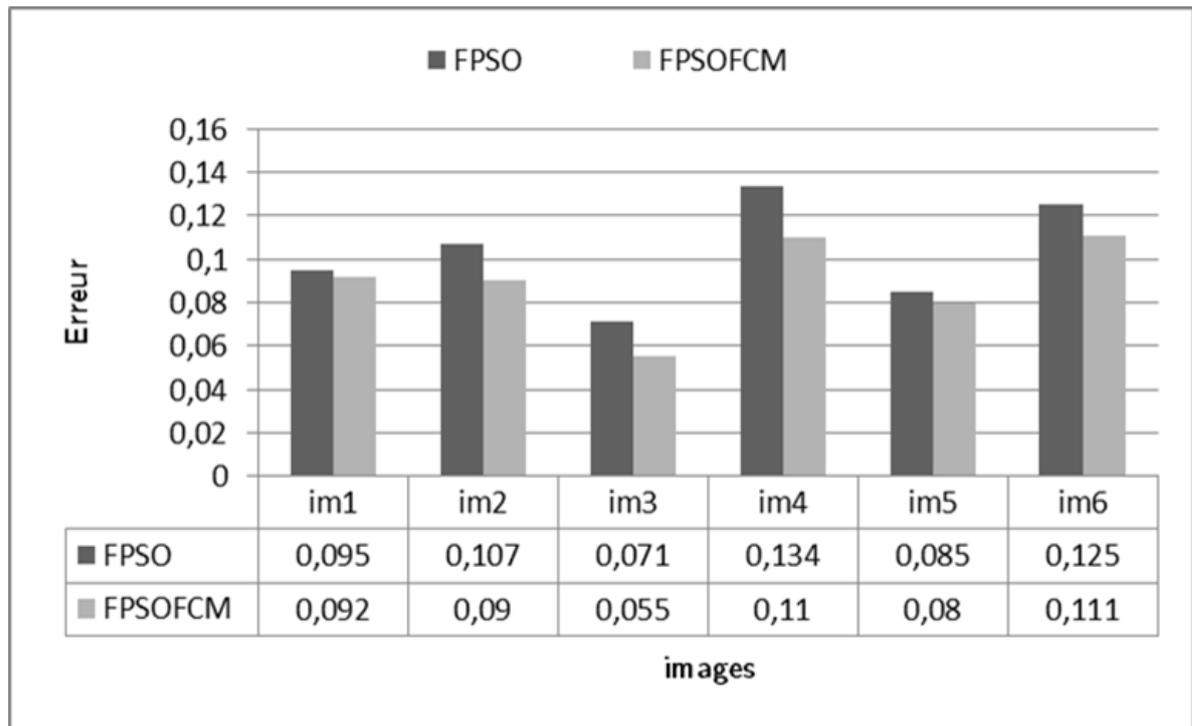


Figure 6.34: Erreur de classification de la matière grise.

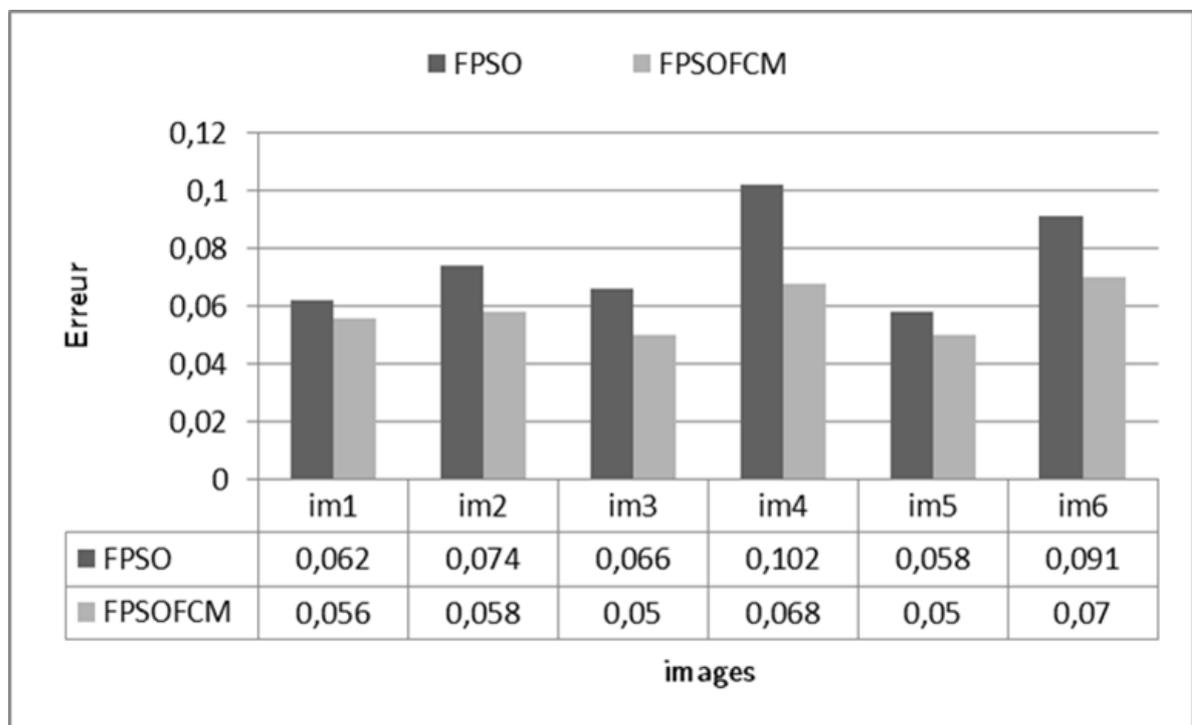


Figure 6.35: Erreur de classification de la matière blanche.

6.5 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle méthode de segmentation d'images FPSOFCM, basée sur l'hybridation de FCM avec une variante de la métaheuristique PSO. L'hybridation des deux algorithmes est exploitée afin de pallier au problème du blocage dans des minima locaux. Pour cela, nous avons défini une nouvelle fonction objectif qui combine des indices de validité flous, permettant de chercher la meilleure configuration initiale des centres de classes de l'algorithme FCM. Nous avons justifié ensuite, le choix des principaux paramètres régissant cette approche.

Dans la partie expérimentale, nous avons exposé les résultats sur 6 échantillons d'images IRM simulées et réelles. Dans un premier temps, les résultats de FCM et FPSOFCM sont présentés selon plusieurs critères, à savoir : PC, MPC, PE et XB. Puis, en second temps, les résultats visuels de la segmentation ont été illustrés, où nous avons constaté que contrairement à l'algorithme FCM, FPSOFCM a caractérisé correctement les différents tissus cérébraux pour la plupart des images de test. Enfin, une étude comparative de FPSOFCM [144] et FPSO [179] est présentée.

Les résultats obtenus par FPSOFCM sur l'ensemble des images de test, sont assez encourageants et comparés de façon favorable avec ceux d'autres méthodes existantes. On peut conclure que l'approche proposée FPSOFCM est robuste pour tout type de modalité, même en présence de bruit, ce qui nous permet de confirmer l'intérêt majeur des métaheuristicques à améliorer la qualité de la segmentation.

Conclusion générale et perspectives

Au cours de cette thèse, nous nous sommes intéressés à l'exploitation des métaheuristiques bio-inspirées dans le domaine de la segmentation d'images IRM cérébrales. L'objectif recherché était de reformuler le problème de la segmentation en un problème d'optimisation, en vue d'améliorer les résultats et aider le médecin à caractériser les différents tissus cérébraux sains et pathologiques (tumeurs).

Après avoir effectué une revue bibliographique sur les différentes méthodes de segmentation et décelé les avantages et les inconvénients de chacune, nous avons constaté l'importance et la difficulté du problème de la segmentation d'images IRM cérébrales. Deux raisons expliquent cette difficulté : d'une part, la diversité des approches qui continuent à être proposées dans la littérature, sans arriver à une méthode générique, applicable à une large variété d'images, et d'autre part, les problèmes dus non seulement aux artefacts spécifiques à l'IRM (bruit, l'effet du volume partiel, etc.), mais aussi, à l'anatomie complexe du cerveau humain. De ce fait, nous nous sommes focalisés sur la mise en œuvre d'une approche hybride, qui rend la segmentation plus efficace et plus robuste.

La principale contribution de ce travail de thèse porte sur l'amélioration de l'algorithme de segmentation floue FCM. Ce dernier a été largement utilisé à cause de son efficacité à manipuler des données bruitées et imprécises. Néanmoins, son principal inconvénient est lié à la forte dépendance aux positions initiales des centres de classes, ce qui conduit au blocage dans des minima locaux. Afin de pallier cet inconvénient, nous avons proposé une hybridation de FCM avec l'algorithme bio-inspiré FPSO. Cet algorithme stochastique qui s'inspire à l'origine du comportement social des animaux évoluant en essaim, est caractérisé par sa simplicité et son adaptabilité pour résoudre des problèmes d'optimisation continue. La méthode proposée se déroule sur deux étapes. Premièrement, une nouvelle fonction objectif basée sur des indices de validité flous est définie. Dans

cette étape, les particules de l'essaim parcourent l'espace de recherche afin de trouver les meilleurs centres de classes qui augmentent à la fois l'homogénéité intra classe, et l'hétérogénéité inter classes. Deuxièmement, les centres de classes obtenus par FPSO, seront utilisés comme configuration initiale de l'algorithme FCM.

L'algorithme de segmentation proposé FPSOFCM a été testé et évalué sur une base de 60 images IRM, comportant : des images simulées affectées de plusieurs types et taux de bruit et d'inhomogénéité RF, et des images réelles pathologiques. Ces évaluations ont été produites par comparaison avec les images de références. Les résultats visuels de la segmentation ont été illustrés, où nous avons constaté que contrairement à l'algorithme FCM, FPSOFCM a caractérisé correctement les différents tissus cérébraux pour la majorité des images de test. Nous avons comparé notre approche avec l'algorithme standard FCM et l'algorithme FPSO présenté dans [179]. Les différentes expérimentations ont mis en évidence l'efficacité et la supériorité de FPSOFCM par rapport à FCM et FPSO sur la plupart des critères d'évaluation. Toutes ces performances, nous ont permis de confirmer que l'utilisation des algorithmes bio-inspirés, permet de pallier aux problèmes rencontrés dans les méthodes classiques et améliore considérablement la qualité de la segmentation.

A partir de ce travail, de nombreuses perspectives pourraient être envisagées dans le cadre de futurs travaux, on peut citer :

- L'incorporation de l'information spatiale dans la fonction objectif pour rendre l'algorithme plus robuste aux niveaux élevés de bruit.
- Réfléchir sur l'intégration de plusieurs sources d'informations dans le cadre de la fusion des données.
- Exploiter d'autres techniques bio-inspirées et d'autres modes d'hybridation.
- Développer un système capable de donner un diagnostic à travers les résultats de segmentation générés par notre approche.

Enfin, une perspective intéressante reste l'adaptation des algorithmes bio-inspirés à la segmentation de séquences d'images IRM. Ce qui consistera à résoudre un problème d'optimisation dynamique.

Annexe A

Résultats sur des coupes coronales

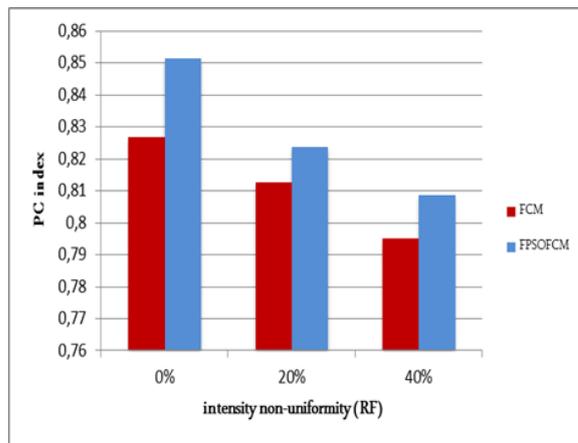
Dans cette Annexe, les images pondérées en T1 correspondent à l'échantillon 3 et celles pondérées en DP, à l'échantillon 4.

Tableau A.1: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en T1.

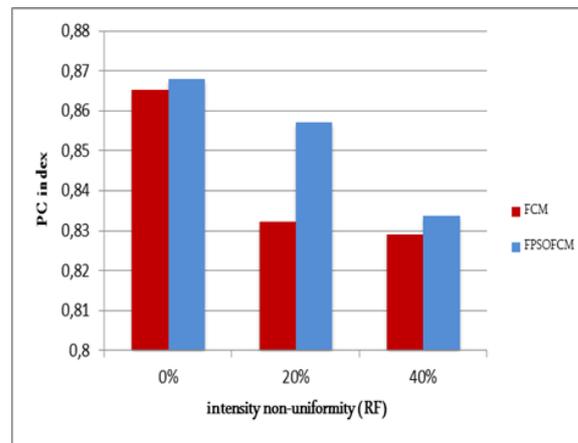
Bruit	0%		3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM	
0%	<i>PC</i>	0.826677 (0.037701)	0.851364 (0.000004)	0.825431 (0.017618)	0.831305 (0.000012)	0.797310 (0.005118)	0.799864 (0.000014)
	<i>MPC</i>	0.768903 (0.050268)	0.801818 (0.000006)	0.767241 (0.023490)	0.775073 (0.000016)	0.729747 (0.006824)	0.733152 (0.000019)
	<i>PE</i>	0.337510 (0.068489)	0.292664 (0.000008)	0.340808 (0.031823)	0.330198 (0.000022)	0.392229 (0.011039)	0.386718 (0.000026)
	<i>XB</i>	0.069045 (0.013740)	0.060047 (0.000003)	0.064028 (0.003143)	0.062979 (0.000008)	0.085267 (0.004343)	0.087431 (0.000018)
20%	<i>PC</i>	0.812681 (0.021763)	0.823560 (0.000011)	0.802658 (0.018877)	0.815014 (0.000013)	0.782115 (0.019714)	0.791978 (0.000032)
	<i>MPC</i>	0.750241 (0.029018)	0.764747 (0.000015)	0.736877 (0.025169)	0.753353 (0.000017)	0.709486 (0.026285)	0.722637 (0.000043)
	<i>PE</i>	0.364874 (0.038440)	0.345658 (0.000022)	0.383961 (0.034650)	0.361279 (0.000023)	0.417802 (0.033586)	0.400994 (0.000053)
	<i>XB</i>	0.065054 (0.005590)	0.062258 (0.000009)	0.077623 (0.000772)	0.078127 (0.000011)	0.082669 (0.009884)	0.077762 (0.000029)
40%	<i>PC</i>	0.795198 (0.020703)	0.808737 (0.000019)	0.812724 (0.000020)	0.812740 (0.000024)	0.797313 (0.000034)	0.797313 (0.000034)
	<i>MPC</i>	0.726931 (0.027604)	0.744983 (0.000026)	0.750299 (0.000026)	0.750320 (0.000032)	0.729751 (0.000045)	0.729751 (0.000045)
	<i>PE</i>	0.396145 (0.035166)	0.373148 (0.000033)	0.361958 (0.000033)	0.361932 (0.000040)	0.390572 (0.000056)	0.390572 (0.000056)
	<i>XB</i>	0.071131 (0.005537)	0.067504 (0.000010)	0.060447 (0.000017)	0.060460 (0.000020)	0.065678 (0.000039)	0.065678 (0.000039)

Tableau A.2: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en DP.

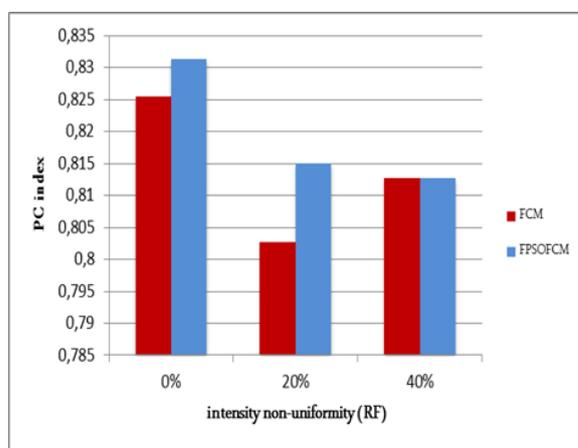
Bruit	0%		3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	FCM	FPSOFCM	
0%	<i>PC</i>	0.865185 (0.013084)	0.868035 (0.011436)	0.836635 (0.028025)	0.859519 (0.000004)	0.819298 (0.017309)	0.815821 (0.016969)
	<i>MPC</i>	0.820247 (0.017445)	0.824046 (0.015249)	0.782180 (0.037367)	0.812692 (0.000005)	0.759064 (0.023078)	0.754428 (0.022625)
	<i>PE</i>	0.252387 (0.025170)	0.246902 (0.021998)	0.301442 (0.051232)	0.259608 (0.000011)	0.336888 (0.030784)	0.343071 (0.030178)
	<i>XB</i>	0.095692 (0.044323)	0.086019 (0.038686)	0.144675 (0.077122)	0.081704 (0.000002)	0.110462 (0.029684)	0.116422 (0.029102)
20%	<i>PC</i>	0.832303 (0.024762)	0.857066 (0.000005)	0.830752 (0.022778)	0.840709 (0.014871)	0.822960 (0.015512)	0.826343 (0.013540)
	<i>MPC</i>	0.776404 (0.033016)	0.809421 (0.000007)	0.774336 (0.030371)	0.787612 (0.019828)	0.763947 (0.020682)	0.768457 (0.018053)
	<i>PE</i>	0.310281 (0.047585)	0.262694 (0.000012)	0.312424 (0.041014)	0.294499 (0.026784)	0.329289 (0.028304)	0.323119 (0.024708)
	<i>XB</i>	0.170753 (0.079783)	0.090970 (0.000002)	0.114347 (0.040418)	0.096724 (0.026512)	0.109495 (0.024647)	0.104075 (0.021424)
40%	<i>PC</i>	0.829141 (0.011184)	0.833710 (0.009134)	0.833423 (0.009082)	0.837395 (0.005944)	0.802005 (0.000002)	0.802006 (0.000000)
	<i>MPC</i>	0.772188 (0.014912)	0.778280 (0.012178)	0.777897 (0.012109)	0.783193 (0.007925)	0.736007 (0.000003)	0.736009 (0.000000)
	<i>PE</i>	0.314139 (0.023520)	0.304530 (0.019208)	0.307556 (0.018511)	0.299457 (0.012117)	0.370098 (0.000004)	0.370096 (0.000000)
	<i>XB</i>	0.111678 (0.026408)	0.100894 (0.021562)	0.103706 (0.019394)	0.095242 (0.012727)	0.105638 (0.000019)	0.105628 (0.000001)



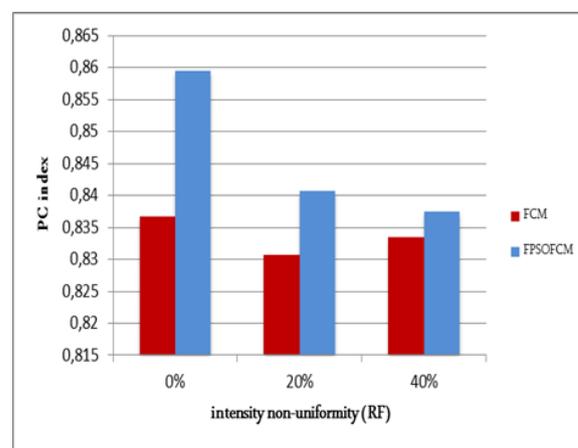
(a) Bruit= 0%



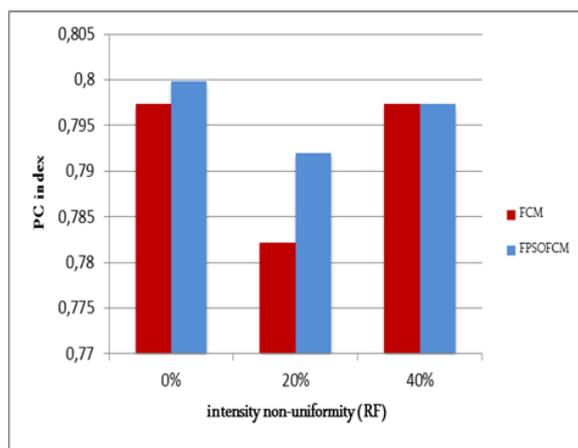
(a) Bruit= 0%



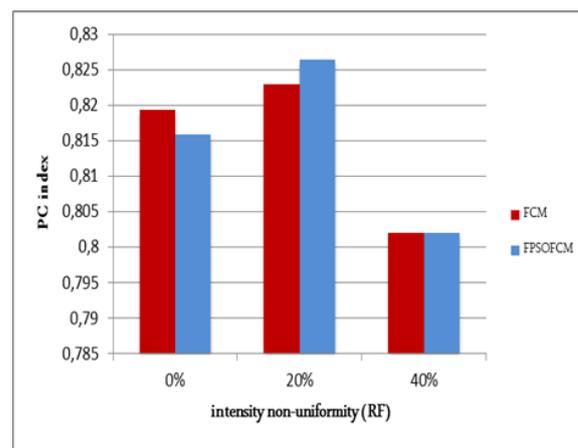
(b) Bruit= 3%



(b) Bruit= 3%



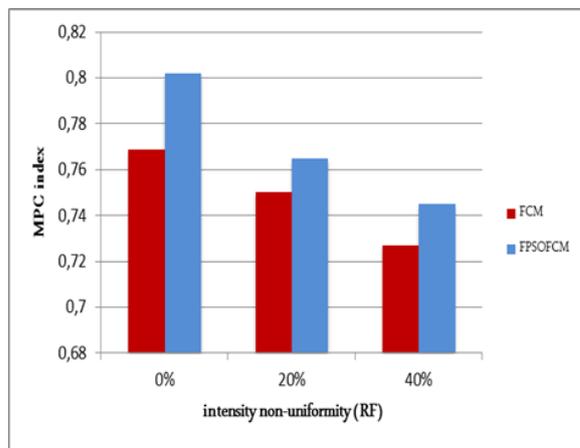
(c) Bruit= 5%



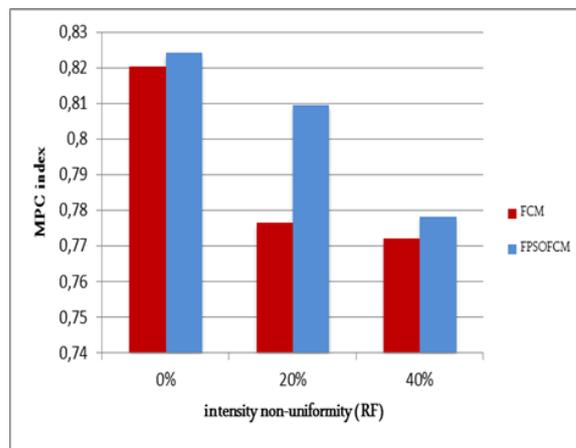
(c) Bruit= 5%

Figure A.1: Comparaison de l'indice PC sur des images simulées pondérées en T1.

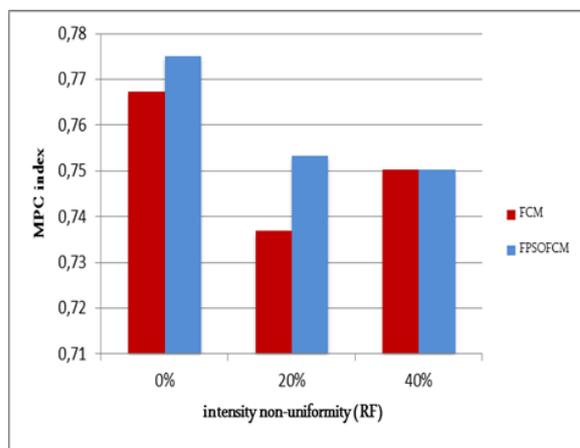
Figure A.2: Comparaison de l'indice PC sur des images simulées pondérées en DP.



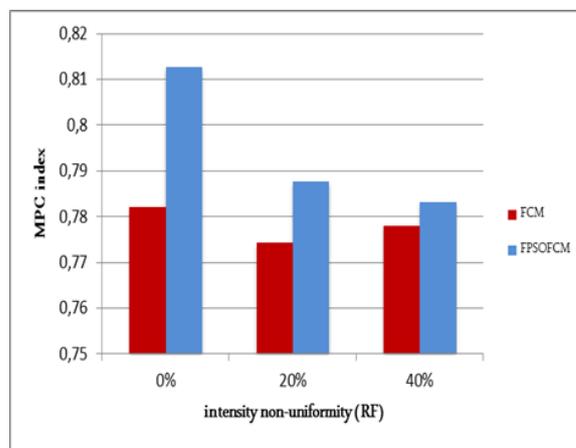
(a) Bruit= 0%



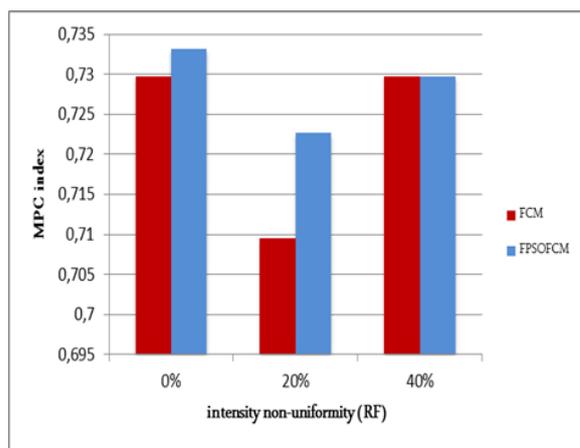
(a) Bruit= 0%



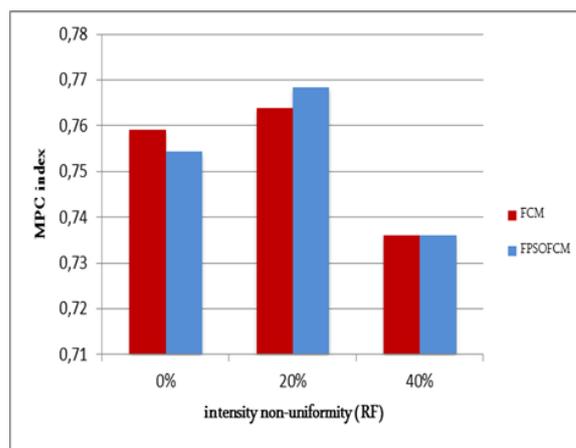
(b) Bruit= 3%



(b) Bruit= 3%



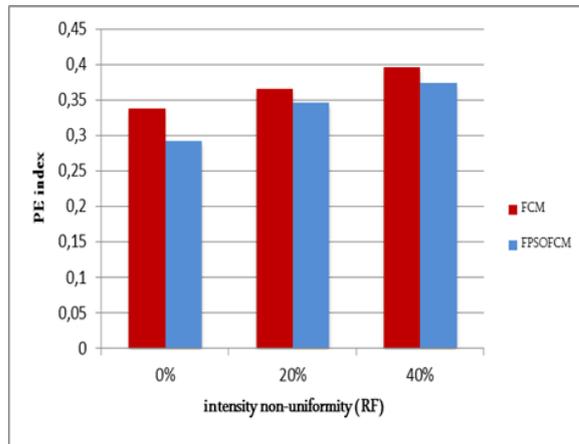
(c) Bruit= 5%



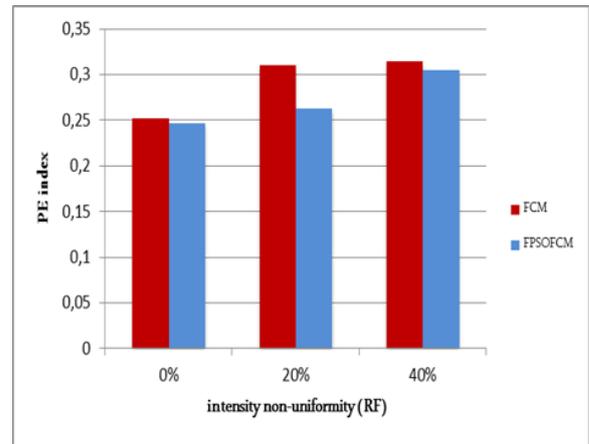
(c) Bruit= 5%

Figure A.3: Comparaison de l'indice MPC sur des images simulées pondérées en T1.

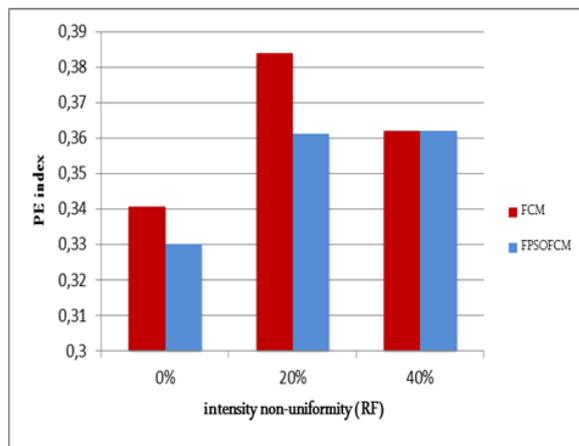
Figure A.4: Comparaison de l'indice MPC sur des images simulées pondérées en DP.



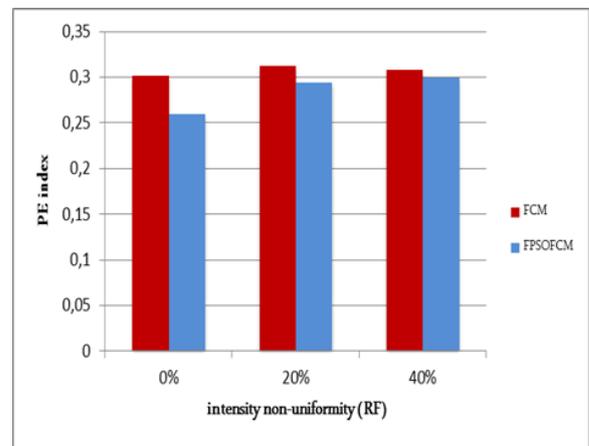
(a) Bruit= 0%



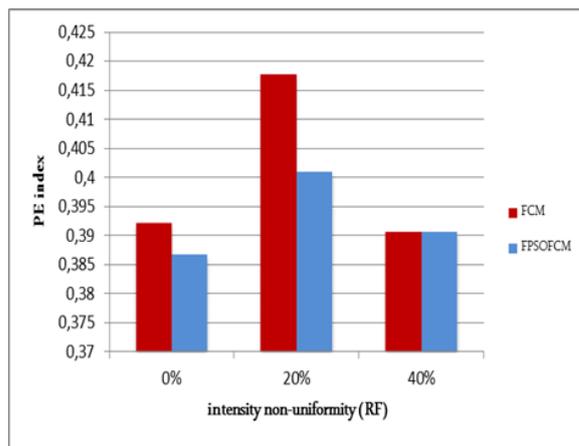
(a) Bruit= 0%



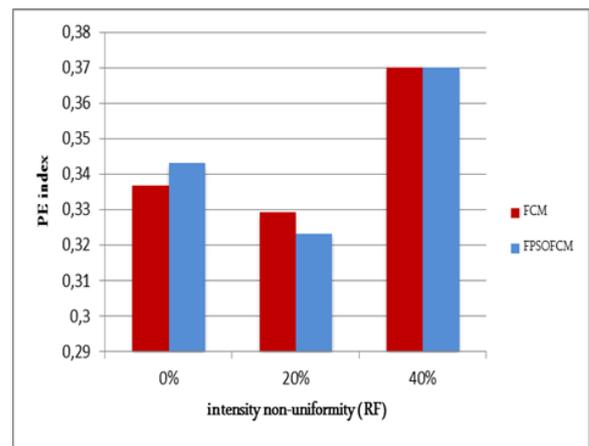
(b) Bruit= 3%



(b) Bruit= 3%



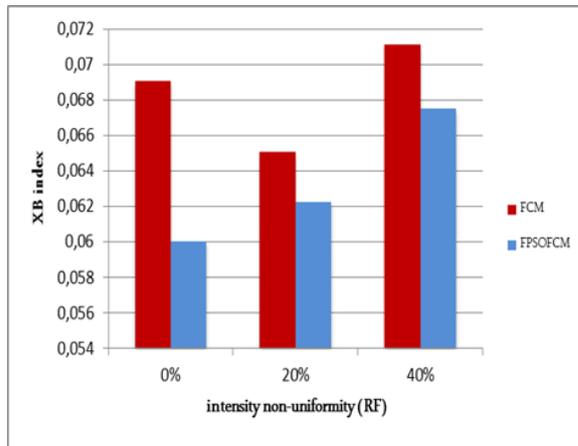
(c) Bruit= 5%



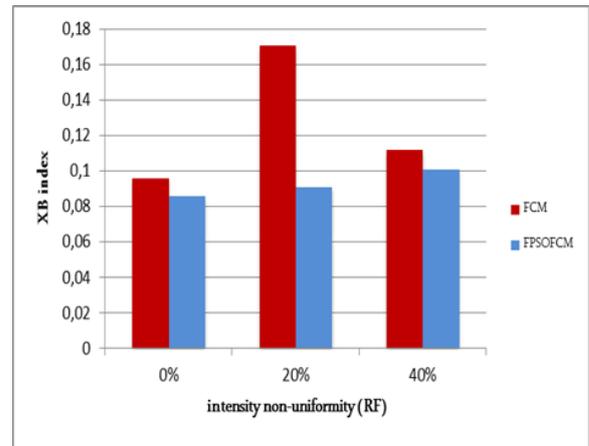
(c) Bruit= 5%

Figure A.5: Comparaison de l'indice PE sur des images simulées pondérées en T1.

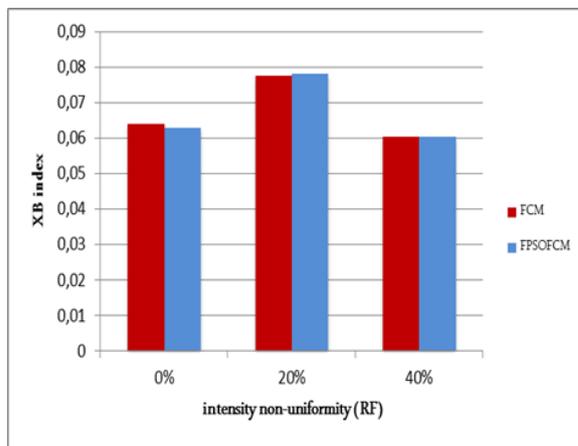
Figure A.6: Comparaison de l'indice PE sur des images simulées pondérées en DP.



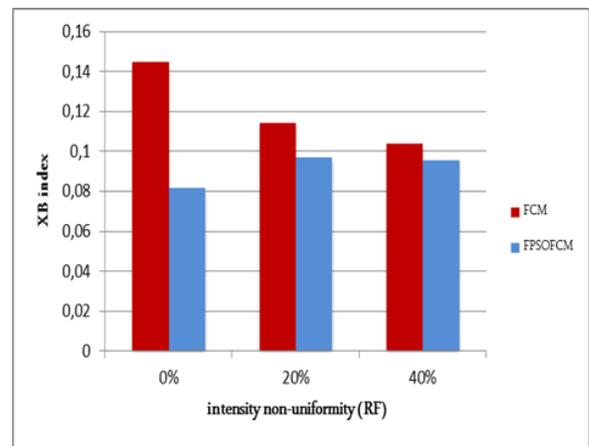
(a) Bruit= 0%



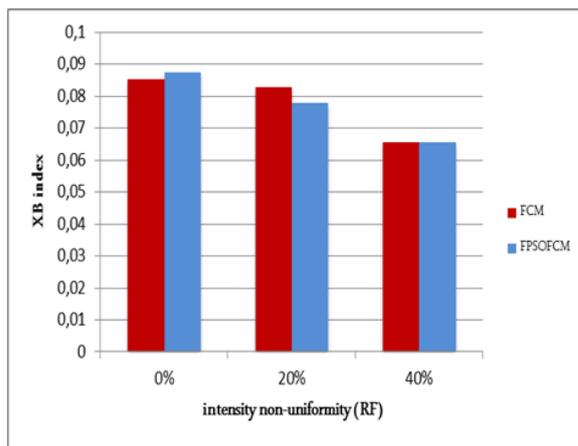
(a) Bruit= 0%



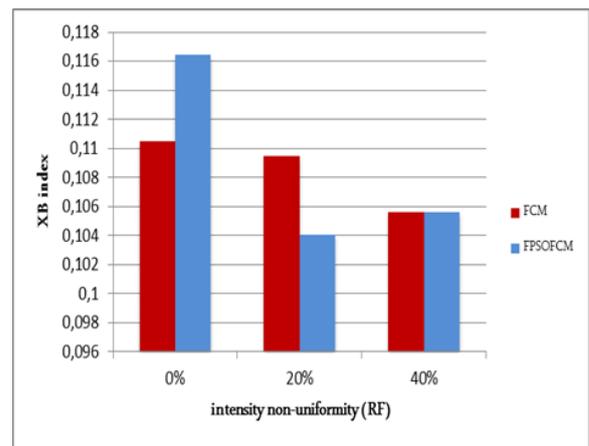
(b) Bruit= 3%



(b) Bruit= 3%



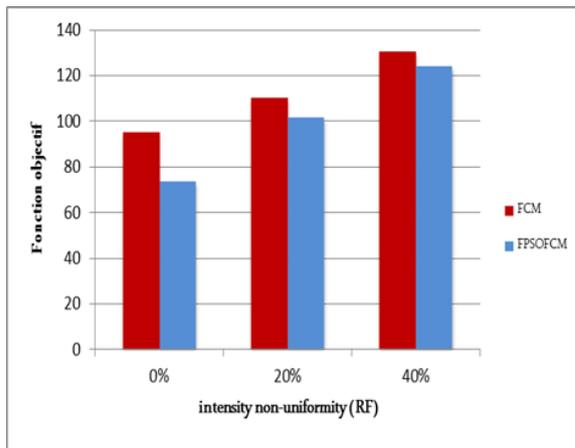
(c) Bruit= 5%



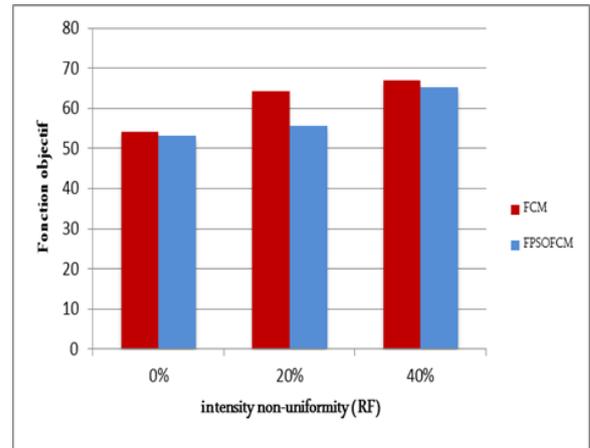
(c) Bruit= 5%

Figure A.7: Comparaison de l'indice XB sur des images simulées pondérées en T1.

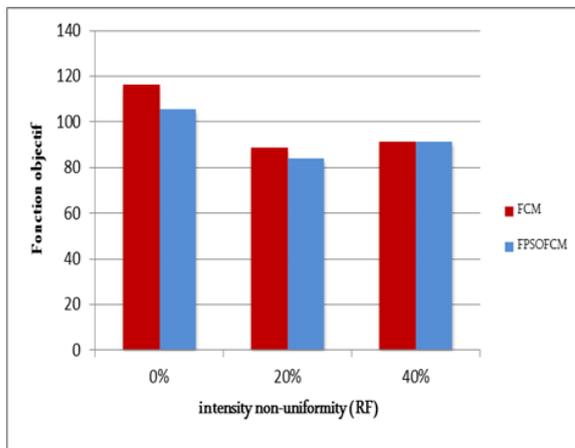
Figure A.8: Comparaison de l'indice XB sur des images simulées pondérées en DP.



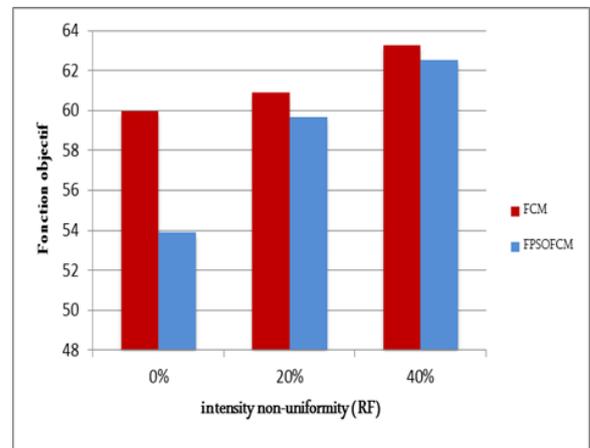
(a) Bruit= 0%



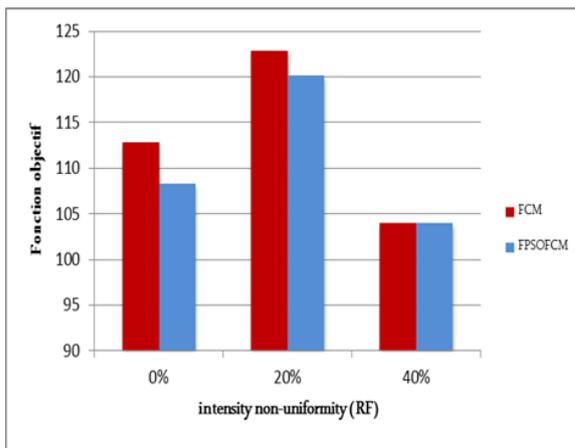
(a) Bruit= 0%



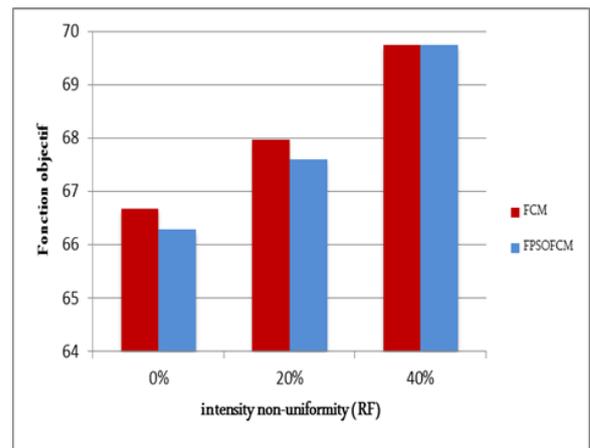
(b) Bruit= 3%



(b) Bruit= 3%



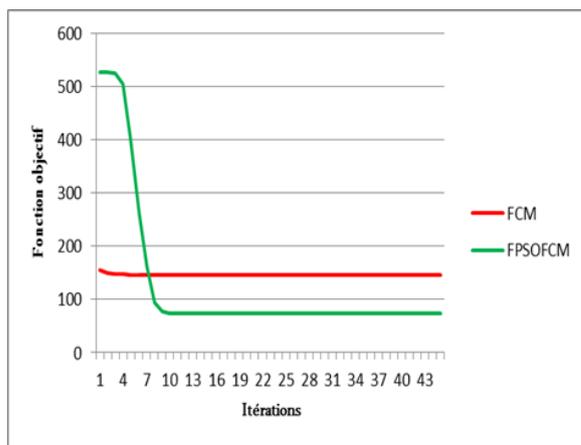
(c) Bruit= 5%



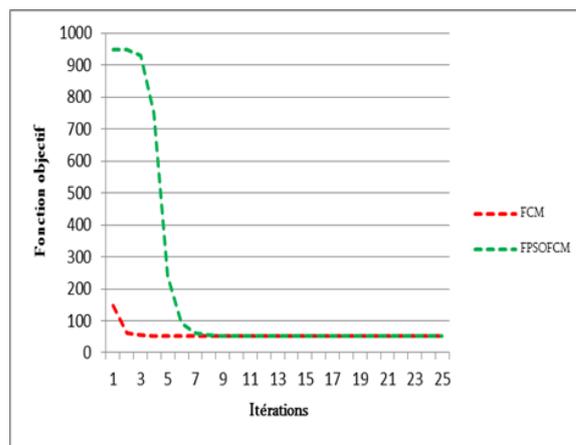
(c) Bruit= 5%

Figure A.9: Comparaison de la fonction objectif moyenne sur des images simulées pondérées en T1.

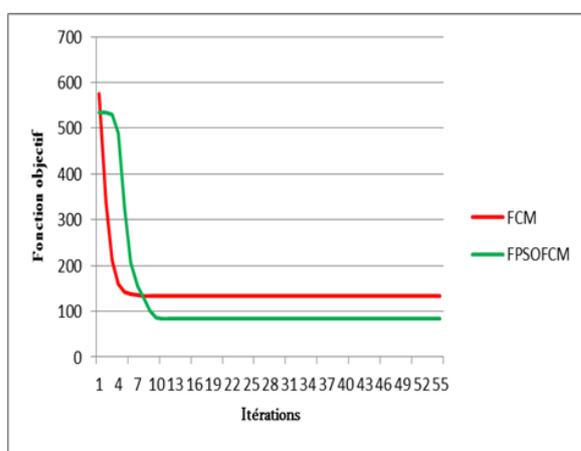
Figure A.10: Comparaison de la fonction objectif moyenne sur des images simulées pondérées en DP.



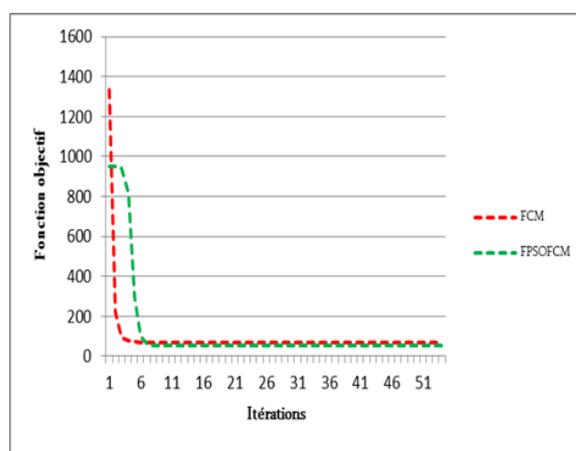
(a) Bruit= 0%



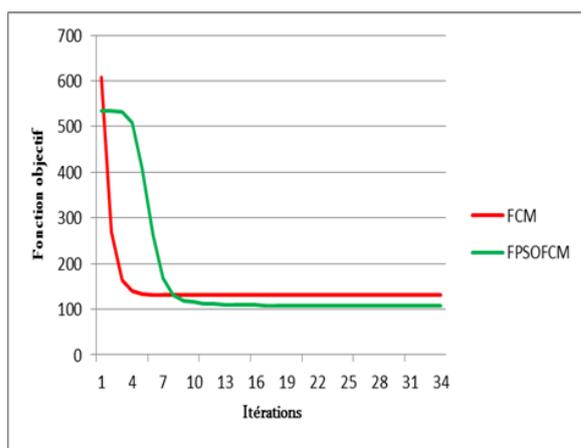
(a) Bruit= 0%



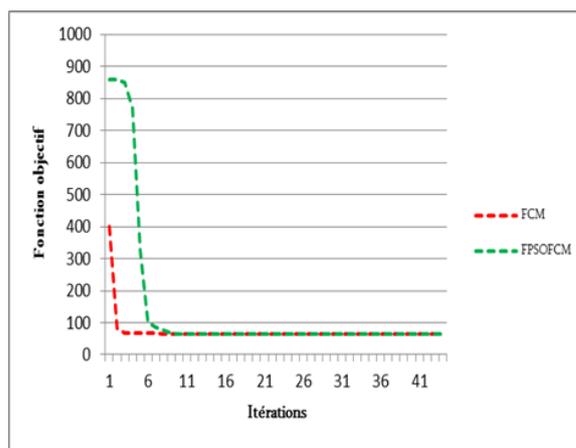
(b) Bruit= 3%



(b) Bruit= 3%



(c) Bruit= 5%



(c) Bruit= 5%

Figure A.11: Convergence de FCM et FPSOFM sur des images simulées pondérées en T1.

Figure A.12: Convergence de FCM et FPSOFM sur des images simulées pondérées en DP.

Annexe B

Résultats sur images bruitées (salt & pepper)

Tableau B.1: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en T1.

Bruit	3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	
0%	RF				
	<i>PC</i>	0.849686 (0.014765)	0.859358 (0.000006)	0.842174 (0.011062)	0.853239 (0.000009)
	<i>MPC</i>	0.799582 (0.019686)	0.812478 (0.000008)	0.789566 (0.014749)	0.804318 (0.000012)
	<i>PE</i>	0.290417 (0.028310)	0.271872 (0.000012)	0.304407 (0.020881)	0.283521 (0.000019)
20%	<i>XB</i>	0.080789 (0.001633)	0.079715 (0.000005)	0.087276 (0.005915)	0.093190 (0.000006)
	<i>PC</i>	0.868429 (0.011490)	0.882505 (0.000004)	0.874248 (0.006287)	0.877391 (0.000006)
	<i>MPC</i>	0.824572 (0.015320)	0.843340 (0.000005)	0.832331 (0.008382)	0.836522 (0.000008)
	<i>PE</i>	0.254878 (0.021951)	0.227987 (0.000008)	0.244337 (0.011530)	0.238573 (0.000013)
40%	<i>XB</i>	0.064281 (0.005749)	0.071318 (0.000005)	0.077296 (0.009310)	0.081951 (0.000007)
	<i>PC</i>	0.865132 (0.008341)	0.870588 (0.000008)	0.861377 (0.003687)	0.863793 (0.000011)
	<i>MPC</i>	0.820176 (0.011121)	0.827450 (0.000011)	0.815169 (0.004916)	0.818391 (0.000014)
	<i>PE</i>	0.261890 (0.016364)	0.251188 (0.000016)	0.268580 (0.007448)	0.263699 (0.000020)
<i>XB</i>	0.059095 (0.003267)	0.061238 (0.000007)	0.064539 (0.007497)	0.069445 (0.000007)	

Tableau B.2: Comparaison des performances de FCM et FPSOFCM sur des images simulées pondérées en DP.

Bruit	3%		5%		
	FCM	FPSOFCM	FCM	FPSOFCM	
0%	RF				
	<i>PC</i>	0.876602 (0.008462)	0.879423 (0.000003)	0.873348 (0.007937)	0.877319 (0.000002)
	<i>MPC</i>	0.835470 (0.011283)	0.839231 (0.000004)	0.831131 (0.010583)	0.836425 (0.000003)
	<i>PE</i>	0.230685 (0.017062)	0.224996 (0.000009)	0.237489 (0.016896)	0.229035 (0.000009)
20%	<i>XB</i>	0.089953 (0.040513)	0.076448 (0.000001)	0.098260 (0.045922)	0.075299 (0.000001)
	<i>PC</i>	0.856747 (0.018031)	0.862763 (0.000004)	0.856745 (0.011510)	0.860584 0.000006
	<i>MPC</i>	0.808996 (0.024042)	0.817017 (0.000005)	0.808993 (0.015347)	0.814112 (0.000008)
	<i>PE</i>	0.264819 (0.031883)	0.254179 (0.000010)	0.265127 (0.021639)	0.257908 (0.000015)
40%	<i>XB</i>	0.087835 (0.031240)	0.077420 (0.000002)	0.083191 (0.022724)	0.075616 (0.000001)
	<i>PC</i>	0.862652 (0.011953)	0.866636 0.000001	0.865296 (0.011956)	0.869281 (0.000003)
	<i>MPC</i>	0.816869 (0.015937)	0.822181 (0.000002)	0.820394 (0.015941)	0.825709 (0.000003)
	<i>PE</i>	0.261308 (0.020367)	0.254521 (0.000005)	0.257111 (0.020224)	0.250369 (0.000007)
<i>XB</i>	0.069335 (0.029592)	0.059471 (0.000002)	0.065832 (0.031381)	0.055371 (0.000001)	

Annexe C

Description des images de test

Tableau C.1: Echantillon d'images N° 1.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	T1	0%	0%	Axiale	71
Image 2	T1	3%	0%	Axiale	105
Image 3	T1	5%	0%	Axiale	110
Image 4	T1	0%	20%	Axiale	105
Image 5	T1	3%	20%	Axiale	111
Image 6	T1	5%	20%	Axiale	82
Image 7	T1	0%	40%	Axiale	110
Image 8	T1	3%	40%	Axiale	115
Image 9	T1	5%	40%	Axiale	56

Tableau C.2: Echantillon d'images N° 2.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	DP	0%	0%	Axiale	67
Image 2	DP	3%	0%	Axiale	56
Image 3	DP	5%	0%	Axiale	81
Image 4	DP	0%	20%	Axiale	80
Image 5	DP	3%	20%	Axiale	70
Image 6	DP	5%	20%	Axiale	98
Image 7	DP	0%	40%	Axiale	93
Image 8	DP	3%	40%	Axiale	87
Image 9	DP	5%	40%	Axiale	57

Tableau C.3: Echantillon d'images N° 3.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	T1	0%	0%	Coronale	84
Image 2	T1	3%	0%	Coronale	117
Image 3	T1	5%	0%	Coronale	121
Image 4	T1	0%	20%	Coronale	120
Image 5	T1	3%	20%	Coronale	121
Image 6	T1	5%	20%	Coronale	125
Image 7	T1	0%	40%	Coronale	126
Image 8	T1	3%	40%	Coronale	115
Image 9	T1	5%	40%	Coronale	119

Tableau C.4: Echantillon d'images N° 4.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	DP	0%	0%	Coronale	116
Image 2	DP	3%	0%	Coronale	74
Image 3	DP	5%	0%	Coronale	81
Image 4	DP	0%	20%	Coronale	74
Image 5	DP	3%	20%	Coronale	75
Image 6	DP	5%	20%	Coronale	69
Image 7	DP	0%	40%	Coronale	60
Image 8	DP	3%	40%	Coronale	58
Image 9	DP	5%	40%	Coronale	92

Tableau C.5: Echantillon d'images N° 5.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	T1	0%	0%	Axiale	71
Image 2	T1	5%	0%	Axiale	110
Image 3	T1	0%	20%	Axiale	105
Image 4	T1	5%	40%	Axiale	56
Image 5	DP	0%	0%	Axiale	67
Image 6	DP	5%	0%	Axiale	81
Image 7	DP	0%	40%	Axiale	93
Image 8	DP	5%	40%	Axiale	57

Tableau C.6: Echantillon d'images N° 6.

	Modalité	Bruit	inhomogénéité RF	Coupe	N° coupe
Image 1	T1	3%	0%	Axiale	65
Image 2	T1	5%	0%	Axiale	84
Image 3	T1	3%	20%	Axiale	118
Image 4	T1	5%	20%	Axiale	61
Image 5	T1	3%	40%	Axiale	97
Image 6	T1	5%	40%	Axiale	79

Bibliographie

- [1] Yves Morin, Jean-Pierre Wainsten, and Véra Lemaire. *Larousse médical*. Larousse, Paris, 2006.
- [2] P. C. Lauterbur. Image formation by induced local interactions : Examples employing nuclear magnetic resonance. *Nature*, 242(5394) :190–191, Mar 1973.
- [3] L.J. Erasmus, D. Hurter, M. Naude, H.G. Kritzinger, and S. Acho. A short overview of MRI artefacts. *South African Journal of Radiology*, 8(2), 2004.
- [4] Benoît Scherrer. *Segmentation des tissus et structures sur les IRM cérébrales : agents markoviens locaux coopératifs et formulation bayésienne*. PhD thesis, Grenoble Institute of Technology, France, 2008.
- [5] R.M. Dubois M.S. Cohen and M.M. Zeineh. Rapid and effective correction of rf inhomogeneity for high field magnetic resonance imaging. *Human Brain Mapping*, 10 :204–211, 2000.
- [6] H.D. Cheng, X.H. Jiang, Y. Sun, and Jingli Wang. Color image segmentation : advances and prospects. *Pattern Recognition*, 34(12) :2259 – 2281, 2001.
- [7] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering : A review. *ACM Comput. Surv.*, 31(3) :264–323, September 1999.
- [8] Steven L. Horowitz and Theodosios Pavlidis. Picture segmentation by a tree traversal algorithm. *J. ACM*, 23(2) :368–388, April 1976.
- [9] Steven W. Zucker. Region growing : Childhood and adolescence. *Computer Graphics and Image Processing*, 5(3) :382 – 399, 1976.
- [10] Marcel Bosc. *Contribution à la détection de changements dans des séquences IRM 3D multimodales*. PhD thesis, Louis Pasteur University, Strasbourg, Alsace, France, 2003.
- [11] Hichem Talbi. *Algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images*. PhD thesis, Université Abdelhamid Mehri Constantine 2, 2009.
- [12] Jérémy Lecoœur and Christian Barillot. Segmentation d'images cérébrales : État de l'art. Research Report RR-6306, INRIA, 2007.
- [13] John Canny. Finding edges and lines in images. Technical report, Cambridge, MA, USA, 1983.
- [14] R. Deriche. Optimal edge detection using recursive filtering. In *International Conference on Computer Vision*, 1987.
- [15] J. Shen and S. Castan. An optimal linear operator for edge detection. In *International Conference on Vision and Pattern Recognition*, pages 321–331, 1986.
- [16] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes : Active contour models. *International Journal of Computer Vision*, 1(4) :321–331, Jan 1988.

- [17] Juan C. Moreno, V.B. Surya Prasath, Hugo Proença, and K. Palaniappan. Fast and globally convex multiphase active contours for brain MRI segmentation. *Computer Vision and Image Understanding*, 125 :237 – 250, 2014.
- [18] Ahmad Mostaar, Mohammad Houshyari, and Saeedeh Badiéyan. A novel active contour model for MRI brain segmentation used in radiotherapy treatment planning. *Electron Physician*, 8(5) :2443–2451, May 2016.
- [19] Anne-Sophie Capelle-Laizé. *Three dimensional multi-echoes MR segmentation for the detection of brain tumours by evidence theory*. PhD thesis, Université de Poitiers, December 2003.
- [20] Pablo Mesejo, Andrea Valsecchi, Linda Marrakchi-Kacem, Stefano Cagnoni, and Sergio Damas. Biomedical image segmentation using geometric deformable models and metaheuristics. *Computerized Medical Imaging and Graphics*, 43 :167 – 178, 2015.
- [21] G. Evelin Sujji, Y.V.S. Lakshmi, and G. Wiselin Jiji. MRI brain image segmentation based on thresholding. *International Journal of Advanced Computer Research*, 3(1) :97–101, 2013.
- [22] Mohammad Hamed Mozaffari and Won-Sook Lee. Multilevel thresholding segmentation of T2 weighted brain MRI images using convergent heterogeneous particle swarm optimization. *CoRR*, abs/1605.04806, 2016.
- [23] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1) :62–66, Jan 1979.
- [24] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, and Image Processing*, 29(3) :273 – 285, 1985.
- [25] Ali El-Zaart and Ali A.Ghosn. MRI images thresholding for alzheimer detection. *International Journal of Computer Science and Information Technology (CS & IT)*, 2013.
- [26] N. Mohd Saad, S. A R Abu-Bakar, Sobri Muda, M. Mokji, and A. R. Abdullah. Automated region growing for segmentation of brain lesion in diffusion-weighted mri. In *Lecture Notes in Engineering and Computer Science*, volume 1, pages 674–677, 2012.
- [27] k Dvorak, p. Bartusek and E Gescheidtova. Automatic segmentation of multi-contrast MRI using statistical region merging. In *Progress in Electromagnetics Research Symposium*, pages 1865–1869, 2014.
- [28] S. L. Horowitz and T. Pavlidis. Picture Segmentation by a directed split-and-merge procedure. In *Proceedings of the 2nd International Joint Conference on Pattern Recognition, Copenhagen, Denmark*, pages 424–433, 1974.
- [29] I.N. Manousakas, P.E. Undrill, G.G. Cameron, and T.W. Redpath. Split-and-merge segmentation of magnetic resonance medical images : Performance evaluation and extension to three dimensions. *Computers and Biomedical Research*, 31(6) :393 – 412, 1998.
- [30] Ioannis Marras, Nikolaos Nikolaidis, and Ioannis Pitas. 3d geometric split-merge segmentation of brain MRI datasets. *Computers in Biology and Medicine*, 48 :119 – 132, 2014.
- [31] Jean Serra. *Image Analysis and Mathematical Morphology*. Academic Press, Inc., Orlando, FL, USA, 1983.

- [32] H. Digabel and C Lantuejoul. Iterative algorithms. In *Proceedings of the 2nd European Symposium Quantitative Analysis of Microstructures in Material Science, Biology and Medicine*, pages 85–89, 1978.
- [33] M. E. Brummer, R. M. Mersereau, R. L. Eisner, and R. R. J. Lewine. Automatic detection of brain contours in MRI data sets. *IEEE Transactions on Medical Imaging*, 12(2) :153–166, Jun 1993.
- [34] L. Gui, R. Lisowski, T. Faundez, P. S. Hüppi, F. Lazeyras, and M. Kocher. Automatic segmentation of newborn brain MRI using mathematical morphology. In *IEEE International Symposium on Biomedical Imaging : From Nano to Macro*, pages 2026–2030, March 2011.
- [35] Barra.V. *Segmentation floue des tissus cérébraux en IRM 3D : une approche possibiliste versus autres méthodes*. PhD thesis, Universite Blaise Pascal, 1999.
- [36] J. L. Marroquin, B. C. Vemuri, S. Botello, E. Calderon, and A. Fernandez-Bouzas. An accurate and efficient bayesian method for automatic segmentation of brain MRI. *IEEE Transactions on Medical Imaging*, 21(8) :934–945, Aug 2002.
- [37] Ali Ahmadvand and Peyman Kabiri. Multispectral MRI image segmentation using markov random field model. *Signal, Image and Video Processing*, 10(2) :251–258, Feb 2016.
- [38] N. E. A. Khalid and P. N. M. M. Haniff. MRI brain abnormalities segmentation using k-nearest neighbors (k-nn). *Int. J. Comput. Sci. Eng*, 3(2) :980, 2011.
- [39] S. Pereira, A. Pinto, V. Alves, and C. A. Silva. Brain tumor segmentation using convolutional neural networks in MRI images. *IEEE Transactions on Medical Imaging*, 35(5) :1240–1251, May 2016.
- [40] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1 : Statistics*, pages 281–297, Berkeley, Calif., 1967.
- [41] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.
- [42] R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2) :98–110, May 1993.
- [43] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of The Royal Statistical Society, Series B*, 39(1) :1–38, 1977.
- [44] Dorin Comaniciu and Peter Meer. Mean shift : A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5) :603–619, May 2002.
- [45] G. Ball and D. Hall. Isodata : A novel method of data analysis and pattern classification. Technical report, Stanford Research Institute, Menlo Park, 1965.
- [46] Amir Nakib. *Design of new metaheuristics for image segmentation. Application on biomedical imaging*. PhD thesis, Université Paris XII Val de Marne, December 2007.
- [47] K. Sinha and G. R. Sinha. Efficient segmentation methods for tumor detection in MRI images. In *Electrical, Electronics and Computer Science (SCEECS), IEEE Students' Conference on*, pages 1–6, March 2014.
- [48] Mothd Belal Al-Daoud. A new algorithm for cluster initialization. In *The Second World Enformatika Conference, WEC'05, Istanbul, Turkey, CDROM*, pages 74–76, 2005.

- [49] M. Emre Celebi, Hassan A. Kingravi, and Patricio A. Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications*, 40(1) :200 – 210, 2013.
- [50] J. C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3) :32–57, 1973.
- [51] A.N. Benaichouche, H. Oulhadj, and P. Siarry. Improved spatial fuzzy c-means clustering for image segmentation using PSO initialization, mahalanobis distance and post-segmentation correction. *Digital Signal Processing*, 23(5) :1390 – 1400, 2013.
- [52] N. N. Gopal and M. Karnan. Diagnose brain tumor through MRI using image processing clustering algorithms such as fuzzy c means along with intelligent optimization techniques. In *IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, Dec 2010.
- [53] S. Bharathi and P. Venkatesan. A survey on image segmentation by fuzzy c means clustering techniques. In *2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pages 604–608, July 2016.
- [54] Semchedine Moussa, Toumi Lyazid, and Moussaoui Abdelouahab. Nouvelle méthode optimale de classification des données : application à la segmentation des IRM cérébrales. In *2nd International Conference : E-Medical Systems. Sfax- Tunisia*, October 2008.
- [55] Lofti A. Zadeh. Fuzzy sets as a basis for a theory of probability. *Fuzzy Sets and Systems*, 1 :3–28, 1978.
- [56] Francesco Masulli and Andrea Schenone. A fuzzy clustering based segmentation system as support to diagnosis in medical imaging. *Artificial Intelligence in Medicine*, 16(2) :129 – 147, 1999.
- [57] A. Rajendran and R. Dhanasekaran. Enhanced possibilistic fuzzy c-means algorithm for normal and pathological brain tissue segmentation on magnetic resonance brain image. *Arabian Journal for Science and Engineering*, 38(9) :2375–2388, Sep 2013.
- [58] Hassan Khotanlou and Mahlagha Afrasiabi. Segmentation of multiple sclerosis lesions in brain mr images using spatially constrained possibilistic fuzzy c-means classification. *J Med Signals Sens*, 1(3) :149–155, Jul 2011.
- [59] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek. A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems*, 13(4) :517–530, Aug 2005.
- [60] Zhenping Xie, Shitong Wang, and F. L. Chung. An enhanced possibilistic c-means clustering algorithm epcm. *Soft Computing*, 12(6) :593–611, Apr 2008.
- [61] Sami Hlioui. *Étude d’une machine synchrone à double excitation : contribution à la mise en place d’une plate-forme de logiciels en vue d’un dimensionnement optimal*. PhD thesis, Université de Technologie de Belfort-Montbeliard, 2008.
- [62] Lemouari Ali. introduction aux méthaheuristiques. *Support de Cours Université de Jijel Faculté des Sciences Exactes et Informatique*, 2014.
- [63] Abbas El Dor. *Improvement of particle swarm optimization algorithms : applications in image segmentation and electronics*. PhD thesis, Université Paris-Est, December 2012.
- [64] M. Avriel. *Nonlinear Programming : Analysis and Methods*. Prentice-Hall, Englewood Cliffs, NJ, 1976.

- [65] Dimitri P. Bertsekas. Projected newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization*, 20(2) :221–246, 1982.
- [66] George B. Dantzig and Philip Wolfe. The decomposition algorithm for linear programs. *Econometrica*, 29(4) :767–778, 1961.
- [67] Jason Brownlee. *Clever Algorithms : Nature-Inspired Programming Recipes*. Lulu.com, 1st edition, 2011.
- [68] Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5) :533 – 549, 1986.
- [69] Christian Blum and Andrea Roli. Metaheuristics in combinatorial optimization : Overview and conceptual comparison. *ACM Comput. Surv.*, 35(3) :268–308, September 2003.
- [70] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598) :671–680, 1983.
- [71] John H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, 1975. second edition, 1992.
- [72] Dervis Karaboga and Bahriye Basturk. A powerful and efficient algorithm for numerical function optimization : artificial bee colony (ABC) algorithm. *J. Global Optimization*, 39(3) :459–471, 2007.
- [73] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948, Nov 1995.
- [74] Andrés Pérez-Urbe. Systèmes bio-inspirés ou intelligence artificielle bio-inspirée. *Support de cours. Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud*, 2012.
- [75] John Von Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, USA, 1966.
- [76] Rafał Henryk Kartaszyński and Paweł Mikołajczak. *MRI Brain Segmentation Using Cellular Automaton Approach*, pages 17–24. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [77] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6) :1087–1092, 1953.
- [78] Yann Cooren. *Improvement of an adaptive algorithm of Particulate Swarm Optimization : application in medical engineering and electronics*. PhD thesis, Université Paris-Est, November 2008.
- [79] Neeraj Sharma, Amit K. Ray, Shiru Sharma, K.K. Shukla, Lalit Aggarwal, and Satyajit Pradhan. Segmentation of medical images using simulated annealing based fuzzy c means algorithm. *International Journal of Biomedical Engineering and Technology*, 2(3) :260–278, 2009.
- [80] Baptiste Autin. Les métaheuristiques en optimisation combinatoire. *Mémoire présenté en vue d'obtenir l'examen probatoire en informatique, Conservatoire National des Arts et Métiers Paris*, 2006.
- [81] Dario Floreano and Claudio Mattiussi. *Bio-Inspired Artificial Intelligence : Theories, Methods, and Technologies*. The MIT Press, 2008.

- [82] S. N. Sivanandam and S. N. Deepa. *Introduction to Genetic Algorithms*. Springer Publishing Company, Incorporated, 1st edition, 2007.
- [83] B. Bhanu, S. Lee, and J. Ming. Adaptive image segmentation using a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(12) :1543–1567, Dec 1995.
- [84] Y. A. Ghassabeh, N. Forghani, M. Forouzanfar, and M. Teshnehlab. MRI fuzzy segmentation of brain tissue using IFCM algorithm with genetic algorithm optimization. In *IEEE/ACS International Conference on Computer Systems and Applications*, pages 665–668, May 2007.
- [85] Marco Dorigo. *Optimization, Learning and Natural Algorithms*. PhD thesis, Politecnico di Milano, Italy, 1992.
- [86] Xiao-Nian Wang, Yuan-Jing Feng, and Zu-Ren Feng. Ant colony optimization for image segmentation. In *International Conference on Machine Learning and Cybernetics*, volume 9, pages 5355–5360, Aug 2005.
- [87] Na Shi Xumin Liu, Xiaojun Wang and Cailing Li. Image segmentation algorithm based on improved ant colony algorithm. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 7(3) :433–442, 2014.
- [88] Rainer Storn and Kenneth Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4) :341–359, Dec 1997.
- [89] Julien Lepagnot. *Design of metaheuristics for dynamic optimization : application to the analysis of MRI image sequences*. PhD thesis, Université Paris-Est, December 2011.
- [90] M. G. H. Omran, A. P. Engelbrecht, and A. Salman. Differential evolution methods for unsupervised image classification. In *IEEE Congress on Evolutionary Computation*, volume 2, pages 966–973 Vol. 2, Sept 2005.
- [91] Zhenkui Pei, Yanli Zhao, and Zhen Liu. Image segmentation based on differential evolution algorithm. In *International Conference on Image Analysis and Signal Processing*, pages 48–51, April 2009.
- [92] Soham Sarkar, Gyana Ranjan Patra, Swagatam Das, and Sheli Sinha Chaudhuri. *Fuzzy Clustering of Image Pixels with a Fitness-Based Adaptive Differential Evolution*, pages 179–188. Springer International Publishing, Cham, 2013.
- [93] Craig W. Reynolds. Flocks, herds and schools : A distributed behavioral model. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '87*, pages 25–34, New York, NY, USA, 1987. ACM.
- [94] Dipankar DasGupta. *Artificial Immune Systems and Their Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st edition, 1998.
- [95] Deneche abdelhakim. Approches bio-inspirees pour la reconnaissance de formes. *mémoire de magister université Mentouri de Constantine*, 2006.
- [96] Erik Cuevas, Valentín Osuna-Enciso, Daniel Zaldívar, and Marco Pérez-Cisneros. *A Novel Multi-threshold Segmentation Approach Based on Artificial Immune System Optimization*, pages 309–317. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [97] S. Valarmathy, Vanitha, and N. Suthanthira. Feature selection optimization using artificial immune system algorithm for identifying dementia in MRI images. *Journal of Medical Imaging and Health Informatics*, 7(1), 2017.

- [98] Zong Woo Geem, Joong Hoon Kim, and G.V. Loganathan. A new heuristic optimization algorithm : Harmony search. *Simulation*, 76(2) :60–68, 2001.
- [99] Z. Shao X. Li and J. Qian. An optimizing method based on autonomous animates : fish-swarm algorithm. *Systems Engineering Theory and Practice*, 22 :32–38, 2002.
- [100] Mehdi Neshat, Ghodrat Sepidnam, Mehdi Sargolzaei, and Adel Najaran Toosi. Artificial fish swarm algorithm : A survey of the state-of-the-art, hybridization, combinatorial and indicative applications. *Artif. Intell. Rev.*, 42(4) :965–997, December 2014.
- [101] Mingyan Jiang, Nikos E. Mastorakis, Dongfeng Yuan, and Miguel Angel Lagunas. *Image Segmentation with Improved Artificial Fish Swarm Algorithm*, pages 133–138. Springer US, Boston, MA, 2009.
- [102] K. M. Passino. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems*, 22(3) :52–67, Jun 2002.
- [103] P.D. Sathya and R. Kayalvizhi. Modified bacterial foraging algorithm based multilevel thresholding for image segmentation. *Engineering Applications of Artificial Intelligence*, 24(4) :595 – 615, 2011.
- [104] Hongwei Mo and Yujing Yin. Image segmentation based on bacterial foraging and FCM algorithm. *Int. J. Swarm. Intell. Res.*, 2(3) :16–28, July 2011.
- [105] S. Binitha and S. S. Sathya. A survey of bio inspired optimization algorithms. *International Journal of Soft Computing and Engineering*, 2(2) :137–151, 2012.
- [106] Salima Ouadfel and Souham Meshoul. Handling fuzzy image clustering with a modified abc algorithm. *International Journal Intelligent System and Application*, pages 65 – 74, 2012.
- [107] Esmat Rashedi, Hossein Nezamabadi-pour, and Saeid Saryazdi. Gsa : A gravitational search algorithm. *Information Sciences*, 179(13) :2232 – 2248, 2009.
- [108] Dan Stefanoiu. optimisation inspirée par la nature. *Présentation École d’été franco-roumaine. Université polytechnique Bucarest*, 2014.
- [109] Esmat Rashedi and Hossein Nezamabadi-pour. A stochastic gravitational approach to feature based color image segmentation. *Engineering Applications of Artificial Intelligence*, 26(4) :1322 – 1332, 2013.
- [110] Genyun Sun and Aizhu Zhang. A hybrid genetic algorithm and gravitational search algorithm for image segmentation using multilevel thresholding. In *IbPRIA*, 2013.
- [111] Xin-She Yang. *Firefly Algorithms for Multimodal Optimization*, pages 169–178. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [112] T. Hassanzadeh, H. Vojodi, and A. M. E. Moghadam. An image segmentation approach based on maximum variance intra-cluster method and firefly algorithm. In *Seventh International Conference on Natural Computation*, volume 3, pages 1817–1821, July 2011.
- [113] Parisut Jitpakdee, Pakinee Aimmanee, and Bunyarit Uyyanonvara. Image clustering using fuzzy-based firefly algorithm. In *the International Conference on Information and Communication Technology for Embedded Systems (ICICTES2014), Ayutthaya, Thailand*, 2014.
- [114] Xin-She Yang. *A New Metaheuristic Bat-Inspired Algorithm*, pages 65–74. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

- [115] Adis Alihodzic and Milan Tuba. Improved bat algorithm applied to multilevel image thresholding. *Scientific World Journal*, pages 1–16, 2014.
- [116] Gagandeep Kaur and Naveen Kumari. Segmentation of brain tumor using bat optimization to distinguish wm, gm and csf. *International Journal of Science and Research (IJSR)*, 5 :1446– 1449, 2016.
- [117] Hamed Shah Hosseini. Problem solving by intelligent water drops. In *IEEE Congress on Evolutionary Computation*, pages 3226–3231, Sept 2007.
- [118] X. S. Yang and Suash Deb. Cuckoo search via levy flights. In *World Congress on Nature Biologically Inspired Computing (NaBIC)*, pages 210–214, Dec 2009.
- [119] A. Kaveh and N. Farhoudi. A new optimization method : Dolphin echolocation. *Advances in Engineering Software*, 59 :53 – 70, 2013.
- [120] U. Premaratne, J. Samarabandu, and T. Sidhu. A new biologically inspired optimization algorithm. In *International Conference on Industrial and Information Systems (ICIIS)*, pages 279–284, Dec 2009.
- [121] Antonio Mucherino and Onur Seref. Monkey search : a novel metaheuristic search for global optimization. *AIP Conference Proceedings*, 953(1) :162–173, 2007.
- [122] Amir Hossein Gandomi and Amir Hossein Alavi. Krill herd : A new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation*, 17(12) :4831 – 4845, 2012.
- [123] Maziar Yazdani and Fariborz Jolai. Lion optimization algorithm (loa) : A nature-inspired metaheuristic algorithm. *Journal of Computational Design and Engineering*, 3(1) :24 – 36, 2016.
- [124] Youcef Gheraibia and Abdelouahab Moussaoui. *Penguins Search Optimization Algorithm (PeSOA)*, pages 222–231. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [125] Andrew Davison. *Killer Game Programming in Java*. O’Reilly Media, Inc., 2005.
- [126] Arthur K. Kordon. *Swarm Intelligence : The Benefits of Swarms*, pages 145–174. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [127] Maurice Clerc and Siarry patrick. Une nouvelle métaheuristique pour l’optimisation difficile : la méthode des essaims particulaires. *Journal sur l’enseignement des sciences et technologies de l’information et des systèmes*, 3 :1–13, 2004.
- [128] J. Kennedy. Small worlds and mega-minds : effects of neighborhood topology on particle swarm performance. In *Proceedings of the Congress on Evolutionary Computation-CEC99*, volume 3, pages 1931–1938, 1999.
- [129] Andries P. Engelbrecht. *Computational Intelligence : An Introduction*. Wiley Publishing, 2nd edition, 2007.
- [130] Omran MGH. *Particle swarm optimization methods for pattern recognition and image processing*. PhD thesis, University of Pretoria, Pretoria, South Africa, 2004.
- [131] Y. Shi and R. Eberhart. A modified particle swarm optimizer. In *IEEE International Conference on Evolutionary Computation*, pages 69–73, 1998.
- [132] R. C. Eberhart and Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization. In *Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512)*, volume 1, pages 84–88 vol.1, 2000.

- [133] Russ Eberhart, Pat Simpson, and Roy Dobbins. *Computational Intelligence PC Tools*. Academic Press Professional, Inc., San Diego, CA, USA, 1996.
- [134] M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1) :58–73, Feb 2002.
- [135] Russell C. Eberhart and Yuhui Shi. Comparison between genetic algorithms and particle swarm optimization. In *Proceedings of the 7th International Conference on Evolutionary Programming VII*, EP '98, pages 611–616, London, UK, UK, 1998. Springer-Verlag.
- [136] D. Sedighizadeh and E Masehian. Particle swarm optimization methods, taxonomy and applications. *International Journal of Computer Theory and Engineering*, 1(5) :482–499, 2009.
- [137] Veyis Gunes. *Pattern recognition by combination, cooperation and selection of classifiers*. PhD thesis, Université de La Rochelle, October 2001.
- [138] Laurence Germond. *Three principles of cooperation for the segmentation of magnetic resonance images of the brain*. PhD thesis, Université Joseph-Fourier - Grenoble I, October 1999.
- [139] T. O. Ting, Xin-She Yang, Shi Cheng, and Kaizhu Huang. *Hybrid Metaheuristic Algorithms : Past, Present, and Future*, pages 71–83. Springer International Publishing, Cham, 2015.
- [140] Semchedine Moussa, Toumi Lyazid, and Moussaoui Abdelouahab. Nouvelle variante de l’algorithme FCM appliquée à la segmentation des IRM cérébrales. In *10th Maghrebian Conference on Information Technologies, Oran*, April 2008.
- [141] Brainweb : Simulated brain database. <http://www.bic.mni.mcgill.ca/brainweb/>, 2017.
- [142] Semchedine Moussa, Toumi Lyazid, and Moussaoui Abdelouahab. Initialisation médiane de l’algorithme FCM : Application à la segmentation des IRM cérébrales. In *Colloque International sur les Nouvelles Techniques Immuno-Cognitives dans les Réseaux Informatiques, (NTICRI), Oran*, May 2009.
- [143] Wei Pang, Kang-Ping Wang, Chun-Guang Zhou, and Long-Jiang Dong. Fuzzy discrete particle swarm optimization for solving traveling salesman problem. In *The Fourth International Conference on Computer and Information Technology*, pages 796–800, Sept 2004.
- [144] Semchedine Moussa and Moussaoui Abdelouahab. An efficient particle swarm optimization for MRI fuzzy segmentation. *Romanian Journal of Information Science and Technology*, 20(3) :271 – 285, 2017.
- [145] S.R. Kannan, S. Ramathilagam, R. Devi, and E. Hines. Strong fuzzy c-means in medical image data analysis. *Journal of Systems and Software*, 85(11) :2425 – 2438, 2012.
- [146] Liang Bai, Jiye Liang, and Chuangyin Dang. An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data. *Knowledge-Based Systems*, 24(6) :785 – 795, 2011.
- [147] Ze-Xuan Ji, Quan-Sen Sun, and De-Shen Xia. Retracted : A framework with modified fast FCM for brain mr images segmentation. *Pattern Recognition*, 44(5) :999 – 1013, 2011.

- [148] Yogita K. Dubey and Milind M. Mushrif. FCM clustering algorithms for segmentation of brain mr images. *Adv. Fuzzy Sys.*, 2016 :8–, March 2016.
- [149] He Sanwei Wang Haijun, Zhang Wenting and Deng Yu. An image segmentation method based on cellular automata and fuzzy c-means. *Geomatics and Information Science of Wuhan University*, 35(11) :1288–1291, 2010.
- [150] C. Sompong and S. Wongthanavas. Brain tumor segmentation using cellular automata-based fuzzy c-means. In *13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–6, July 2016.
- [151] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, L. Lanczi, E. Gerstner, M. A. Weber, T. Arbel, B. B. Avants, N. Ayache, P. Buendia, D. L. Collins, N. Cordier, J. J. Corso, A. Criminisi, T. Das, H. Delingette, Ç. Demiralp, C. R. Durst, M. Dojat, S. Doyle, J. Festa, F. Forbes, E. Geremia, B. Glocker, P. Golland, X. Guo, A. Hamamci, K. M. Iftekharuddin, R. Jena, N. M. John, E. Konukoglu, D. Lashkari, J. A. Mariz, R. Meier, S. Pereira, D. Precup, S. J. Price, T. R. Raviv, S. M. S. Reza, M. Ryan, D. Sarikaya, L. Schwartz, H. C. Shin, J. Shotton, C. A. Silva, N. Sousa, N. K. Subbanna, G. Szekely, T. J. Taylor, O. M. Thomas, N. J. Tustison, G. Unal, F. Vasseur, M. Wintermark, D. H. Ye, L. Zhao, B. Zhao, D. Zikic, M. Prastawa, M. Reyes, and K. Van Leemput. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Transactions on Medical Imaging*, 34(10) :1993–2024, Oct 2015.
- [152] Zhang You-Sheng Liu Xiao-Long and Xie Ying. Image segmentation algorithm based on simulated annealing and fuzzy c-means clustering. *Journal of Engineering Graphics*, 28 :89–93, 2007.
- [153] Qing Yang, Zhi Qiang Wang, and Yan Xu. Fuzzy c-means image segmentation algorithm based on chaotic simulated annealing. In *Advanced Development in Automation, Materials and Manufacturing*, volume 624 of *Applied Mechanics and Materials*, pages 536–539. Trans Tech Publications, October 2014.
- [154] S. Saha and S. Bandyopadhyay. MRI brain image segmentation by fuzzy symmetry based genetic clustering technique. In *IEEE Congress on Evolutionary Computation*, pages 4417–4424, Sept 2007.
- [155] M.A. Balafar, Abd Rahman Ramli, M. Iqbal Saripan, Rozi Mahmud, Syahmsiah Mashohor, and Hakimeh Balafar. MRI segmentation of medical images using FCM with initialized class centers via genetic algorithm. In *International Symposium on Information Technology*, volume 4, pages 1–4, Aug 2008.
- [156] R.Venkateswar and S.Muthukumar. Genetic approach on medical image segmentation by generalized spatial fuzzy c- means algorithm. In *IEEE International Conference on Computational Intelligence and Computing Research*, pages 210–213, 2010.
- [157] Omid Jamshidi and Abdol Hamid Pilevar. Automatic segmentation of medical images using fuzzy c-means and the genetic algorithm. *Journal of Computational Medicine*, pages 1 – 7, 2013.
- [158] Yanfang Han and Pengfei Shi. An improved ant colony algorithm for fuzzy clustering in image segmentation. *Neurocomputing*, 70(4) :665 – 671, 2007.

- [159] Peng Huang, Huizhi Cao, and Shuqian Luo. An artificial ant colonies approach to medical image segmentation. *Computer Methods and Programs in Biomedicine*, 92(3) :267 – 273, 2008.
- [160] M. Karnan and T. Logheshwari. Improved implementation of brain MRI image segmentation using ant colony system. In *IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, Dec 2010.
- [161] O. M. Alia, R. Mandava, D. Ramachandram, and M. E. Aziz. A novel image segmentation algorithm based on harmony fuzzy search algorithm. In *International Conference of Soft Computing and Pattern Recognition*, pages 335–340, Dec 2009.
- [162] O. M. Alia, R. Mandava, and M. E. Aziz. A hybrid harmony search algorithm to MRI brain segmentation. In *Cognitive Informatics (ICCI), 2010 9th IEEE International Conference on*, pages 712–721, July 2010.
- [163] X. Chu, Y. Zhu, J. Shi, and J. Song. Method of image segmentation based on fuzzy c-means clustering algorithm and artificial fish swarm algorithm. In *International Conference on Intelligent Computing and Integrated Systems*, pages 254–257, Oct 2010.
- [164] R. Jagadeesan and S. N. Sivanandam. An artificial fish swarm optimized fuzzy MRI image segmentation approach for improving identification of brain tumour. *International journal on computer science and engineering (IJCSE)*, pages 607–616, 2013.
- [165] M. Taherdangkoo, M. Yazdi, and M. H. Rezvani. Segmentation of mr brain images using FCM improved by artificial bee colony (ABC) algorithm. In *Proceedings of the 10th IEEE International Conference on Information Technology and Applications in Biomedicine*, pages 1–5, Nov 2010.
- [166] Ayat Alrosan, Norita Norwawi, Wediah Ismail, and Waleed Alomoush. Artificial bee colony based fuzzy clustering algorithms for MRI image segmentation. In *International conference on Advances in Computer Science and Electronics Engineering*, pages 225–228, 2014.
- [167] Emadaldin Mozafari Majd, M. A. As’ari, U. U. Sheikh, and S. A. R. Abu-Bakar. Hybrid image segmentation using fuzzy c-means and gravitational search algorithm. In *Proc. SPIE, Fourth International Conference on Digital Image Processing (ICDIP)*, volume 8334, 2012.
- [168] Pakinee Aimmanee Parisut Jitpakdee and Bunyarit Uyyanonvara. Image clustering using fuzzy-based firefly algorithm. In *International Conference on Information and Communication Technology for Embedded Systems (ICICTES)*, 2014.
- [169] Shahnorbanun Sahran Waleed Khamees Alomoush, Siti Norul Huda Shei Abdullah and Rizuana Iqbal Hussain. Segmentation of MRI brain images using FCM improved by firefly algorithms. *Journal of Applied Sciences*, pages 66–71, 2014.
- [170] S. Das, A. Abraham, and A. Konar. Spatial information based image segmentation using a modified particle swarm optimization algorithm. In *Sixth International Conference on Intelligent Systems Design and Applications ISDA*, volume 2, pages 438–444, Oct 2006.
- [171] Wensheng Yi, Min Yao, and Zhiwei Jiang. Fuzzy particle swarm optimization clustering and its application to image clustering. In *Proceedings of the 7th IEEE Pacific Rim Conference on Multimedia*, volume 4261 of *Lecture Notes in Computer Science*, pages 459–467. Springer, 2006.

- [172] N. Forghani, M. Forouzanfar, and E. Forouzanfar. MRI fuzzy segmentation of brain tissue using IFCM algorithm with particle swarm optimization. In *22nd international symposium on Computer and information sciences*, pages 1–4, Nov 2007.
- [173] Wei Chen and Kangling Fang. A hybridized clustering approach using particle swarm optimization for image segmentation. In *International Conference on Audio, Language and Image Processing*, pages 1365–1368, July 2008.
- [174] Z. Xian-cheng. Image segmentation based on modified particle swarm optimization and fuzzy c-means clustering. In *2009 Second International Conference on Intelligent Computation Technology and Automation*, volume 1, pages 611–616, Oct 2009.
- [175] Zang Jing and Li Bo. Image segmentation using fast fuzzy c means based on particle swarm optimization. In *3rd International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, pages 370–373, Nov 2010.
- [176] Yan ling Li and Yi Shen. Fuzzy c-means cluster segmentation algorithm based on hybridized particle swarm optimization. In *IEEE Fifth International Conference on Bio-Inspired Computing : Theories and Applications (BIC-TA)*, pages 811–815, Sept 2010.
- [177] Saeed Mirghasemi, Ramesh Rayudu, and Mengjie Zhang. *A New Modification of Fuzzy C-Means via Particle Swarm Optimization for Noisy Image Segmentation*, pages 147–159. Springer International Publishing, Cham, 2016.
- [178] Matej Črepinšek, Shih-Hsi Liu, and Marjan Mernik. Exploration and exploitation in evolutionary algorithms : A survey. *ACM Comput. Surv.*, 45(3) :35 :1–35 :33, July 2013.
- [179] Hesam Izakian and Ajith Abraham. Fuzzy c-means and fuzzy swarm for fuzzy clustering problem. *Expert Systems with Applications*, 38(3) :1835 – 1838, 2011.
- [180] James C. Bezdek. Cluster validity with fuzzy sets. *Journal of Cybernetics*, 3(3) :58–73, 1973.
- [181] J.C. Bezdek. Numerical taxonomy with fuzzy sets. *Journal of Mathematical Biology*, 1(1) :57–71, 1974.
- [182] Sitao Wu and Tommy W.S. Chow. Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density. *Pattern Recognition*, 37(2) :175 – 188, 2004.
- [183] S. Madhukumar and N. Santhiyakumari. Evaluation of k-means and fuzzy c-means segmentation on mr images of brain. *The Egyptian Journal of Radiology and Nuclear Medicine*, 46(2) :475 – 479, 2015.
- [184] Xuanli Lisa Xie and G. Beni. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8) :841–847, Aug 1991.
- [185] Michael Kistler, Serena Bonaretti, Marcel Pfahrer, Roman Niklaus, and Philippe Büchler. The virtual skeleton database : An open access repository for biomedical research and collaboration. *J Med Internet Res*, 15(11) :e245, Nov 2013.
- [186] Neha Vapiwala. Brain tumor, glioblastoma multiforme, frontal lobe. *Abramson Cancer Center of the University of Pennsylvania*, 2001.
- [187] Rajesh N. Dave. Validating fuzzy partitions obtained through c-shells clustering. *Pattern Recognition Letters*, 17(6) :613 – 623, 1996.

Productions scientifiques

Reuves internationales

1. Semchedine Moussa, Moussaoui Abdelouaheb. An Efficient Particle Swarm Optimization for MRI Fuzzy Segmentation. *Romanian Journal of Information Science and Technology*, 20(3) :271-285, 2017.

Communications internationales

1. Toumi Lyazid, Semchedine Moussa, Moussaoui Abdelouaheb. Une approche robuste semi-supervisée pour la détection des tumeurs cérébrales. *3ème journées internationales sur l'informatique graphique. Constantine 2007*.
2. Toumi Lyazid, Semchedine Moussa, Moussaoui Abdelouaheb. Une approche de classification hybride PSO-Floue pour la segmentation des images IRM cérébrales. *2ème conférence internationale sur la vision artificielle. Tizi-ouzou 2007*.
3. Semchedine Moussa, Toumi Lyazid, Moussaoui Abdelouaheb. Nouvelle méthode optimale de classification des données : application à la segmentation des IRM cérébrales. *2nd International Conference : E-Medical Systems, Sfax- Tunisia, October 2008*.
4. Semchedine Moussa, Toumi Lyazid, Moussaoui Abdelouaheb. Nouvelle variante de l'algorithme FCM appliquée à la segmentation des IRM cérébrales. *In 10th Maghrebian Conference on Information Technologies, Oran 2008*.
5. Semchedine Moussa, Toumi Lyazid, Moussaoui Abdelouaheb. Initialisation médiane de l'algorithme FCM : Application à la segmentation des IRM cérébrales. *Colloque International sur les Nouvelles Techniques Immuno-Cognitives dans les Réseaux Informatiques, (NTICRI), Oran 2009*.
6. Zine El Abidine Sid, Lyazid Aoufi, Semchedine Moussa. Nouvelle approche de filtrage Non-linéaire d'images numériques. *In Proceedings of the First International Conference on Signal, Image, Vision aind their Application (SIVA). Université de 8 mai 1945. Guelma 2011*.

Communications nationales

1. Semchedine Moussa, Toumi Lyazid, Moussaoui Abdelouaheb. Ajustement de l'algorithme flou FCM, appliqué à la segmentation des IRM cérébrales. *Premier Forum National des étudiants en Informatique Centre Universitaire de BBA 2008*.
2. Semchedine Moussa, Toumi Lyazid, Moussaoui Abdelouaheb. Initialisation médiane de l'algorithme FCM : Application à la segmentation des IRM cérébrales. *Les journées de l'informatique infodays. Université Hassiba Ben Bououali, Chlef 2011*.

في هذه الأطروحة قمنا بدراسة مختلف طرق تقسيم صور الرنين المغناطيسي للدماغ باستخدام الخوارزميات المستوحاة من الطبيعة. من بين الطرق الآلية، إختارنا خوارزمي المتوسطات الضبابية الذي أثبت فعاليته و تكيفه في معالجة المعطيات الغير دقيقة و الغير مؤكدة. لكن العائق الرئيسي لهذا الخوارزمي هو تأثيره بتهيئة مراكز الأقسام التي تحسب بطريقة عشوائية و تؤدي إلى التقارب المبكر. لتحسين أداء خوارزمي المتوسطات الضبابية، إقتارنا طريقة جديدة لتهيئة مراكز الأقسام تعتمد على خوارزمية استمثال عناصر السرب. تتركز مساهمتنا على مرحلتين أساسيتين. الأولى هي إعادة صياغة مشكلة تقسيم الصور باعتبارها مشكلة التحسين عن طريق إقتراح تابع الهدف جديد، للبحث عن أفضل مراكز الأقسام. تتمثل المرحلة الثانية في إستخدام المراكز السابقة في تهيئة خوارزمي المتوسطات الضبابية. الخوارزمي المطور جرب على صور الرنين المغناطيسي للدماغ اصطناعية و حقيقية. مقارنة النتائج مع خوارزميات أخرى، أثبتت بصفة واضحة فعالية الخوارزمي المقترح.

مفاتيح: صور الرنين المغناطيسي، تقسيم صور، خوارزمي المتوسطات الضبابية، الاستمثال، الخوارزميات المستوحاة من الطبيعة، خوارزمية استمثال عناصر السرب.

Résumé

Dans le cadre de cette thèse, nous avons focalisé notre étude sur la segmentation automatique des images IRM cérébrales par les algorithmes bio-inspirés. Parmi les méthodes non supervisées présentées, nous nous sommes intéressés à l'algorithme de c-moyennes floues (FCM). Cet algorithme a été largement appliqué, au regard de son adaptation et son efficacité à traiter des données imprécises et incertaines. Cependant, son inconvénient majeur est lié à l'initialisation aléatoire des centres de classes qui provoque une convergence prématurée et bloque FCM dans des minima locaux. Afin d'améliorer davantage les performances de FCM, nous avons proposé une nouvelle méthode d'initialisation, basée sur l'algorithme d'optimisation par essaim de particules PSO. Notre contribution s'articule autour de deux étapes principales. La première consiste à reformuler le problème de la segmentation en un problème d'optimisation, par la définition d'une nouvelle fonction objectif qui cherche les meilleurs centres de classes. La deuxième étape concerne l'initialisation de l'algorithme FCM avec les centres déjà obtenus. La méthode développée a été validée sur des images IRM simulées et réelles. La comparaison des résultats de test avec d'autres méthodes de segmentation existantes, montrent clairement la supériorité de l'approche proposée.

Mots clés : IRM, segmentation d'images, FCM, optimisation, métaheuristiques bio-inspirées, OEP.

Abstract:

In this thesis, we are focusing our study on the automatic segmentation of cerebral MRI images using bio-inspired algorithms. Among the unsupervised methods presented, we were interested in fuzzy c-means algorithm (FCM). This algorithm has been widely used, in terms of its adaptation and its efficiency in dealing with imprecise and uncertain data. However, its main weakness is related to the random initialization of cluster centers which causes premature convergence and makes FCM falling into local optimal. In order to improve the performance of FCM, we propose a new initialization method, based on the particle swarm optimization algorithm PSO. Our contribution focuses on two main stages. The first consisted in reformulating the problem of segmentation as an optimization problem by defining a new objective function in order to find the best cluster centers. The second stage is to use the obtained cluster centers as the initial seed of the FCM. The developed method was tested on synthetic and real MRI images. Comparison of test results with other existing segmentation methods, demonstrates clearly the superiority of the proposed approach.

Key words: MRI, image segmentation, FCM, optimization, bio-inspired Metaheuristic, PSO.