



**HAL**  
open science

# Conception de métaheuristiques d'optimisation pour la segmentation d'images : application aux images IRM du cerveau et aux images de tomographie par émission de positons

Ahmed Nasreddine Benaichouche

► **To cite this version:**

Ahmed Nasreddine Benaichouche. Conception de métaheuristiques d'optimisation pour la segmentation d'images : application aux images IRM du cerveau et aux images de tomographie par émission de positons. Analyse numérique [cs.NA]. Université Paris-Est, 2014. Français. NNT : 2014PEST1106 . tel-01143778

**HAL Id: tel-01143778**

**<https://theses.hal.science/tel-01143778>**

Submitted on 20 Apr 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# THÈSE

Pour l'obtention du grade de

## DOCTEUR DE L'UNIVERSITÉ PARIS-EST

Diplôme National - Arrêté du 7 août 2006

ÉCOLE DOCTORALE MATHÉMATIQUES ET STIC (MSTIC, E.D. 532)

DOMAINE DE RECHERCHE : SIGNAL, IMAGE, ET AUTOMATIQUE (CNU 27, 61).

Présentée par

**Ahmed Nasreddine Benaichouche**

---

---

**Conception de métaheuristiques d'optimisation pour la  
segmentation d'images. Application aux images IRM du  
cerveau et aux images de tomographie par émission de  
positons**

---

Directeur de thèse : **Pr. Patrick Siarry**

Soutenue le 10 décembre 2014  
Devant la Commission d'Examen

### JURY

|                                   |  |                    |
|-----------------------------------|--|--------------------|
| Mme. Christine Fernandez-Maloigne | Professeur à l'Université de Poitiers          | Rapporteur         |
| M. Marc Sevaux                    | Professeur à l'Université de Bretagne-Sud      | Rapporteur         |
| M. Jin-Kao Hao                    | Professeur à l'Université d'Angers             | Examineur          |
| Mme. Su Ruan                      | Professeur à l'Université de Rouen             | Examineur          |
| M. Hamouche Oulhadj               | MCF à l'Université de Paris-Est Créteil        | Co-encadrant       |
| M. Patrick Siarry                 | Professeur à l'Université de Paris-Est Créteil | Directeur de thèse |

---

# Résumé

La segmentation d'image est le processus de partitionnement d'une image numérique en régions, non chevauchées, homogènes vis-à-vis de certaines caractéristiques, telles que le niveau de gris, la texture, le mouvement, etc. Elle a des applications dans plusieurs domaines comme l'imagerie médicale, la détection d'objets, la biométrie, l'imagerie par satellite, la navigation de robot, la vidéosurveillance, etc. Le processus de segmentation représente une étape cruciale dans les systèmes de vision par ordinateur, car les caractéristiques et décisions sont extraites et prises à partir de son résultat. Les premiers algorithmes de segmentation d'image ont vu le jour dans les années 1970. Depuis, de nombreuses techniques et méthodes de segmentation ont été expérimentées pour essayer d'améliorer les résultats. Néanmoins, jusqu'à nos jours, aucun algorithme de segmentation d'image n'arrive à fournir des résultats parfaits sur une large variété d'images.

Les "métaheuristiques" sont des procédures conçues pour résoudre des problèmes d'optimisation dits difficiles. Ce sont en général des problèmes aux données incomplètes, incertaines, bruitées ou confrontés à une capacité de calcul limitée. Les métaheuristiques ont connu le succès dans une large variété de domaines. Cela découle du fait qu'elles peuvent être appliquées à tout problème pouvant être exprimé sous la forme d'un problème d'optimisation de critère(s). Ces méthodes sont, pour la plupart, inspirées de la physique (recuit simulé), de la biologie (algorithmes évolutionnaires) ou de l'éthologie (essais particuliers, colonies de fourmis).

Ces dernières années, l'introduction des métaheuristiques dans le domaine du traitement d'images a permis d'étudier la segmentation sous un angle différent, avec des résultats plus ou moins réussis. Dans le but d'apporter notre contribution et d'améliorer davantage les performances des méthodes de segmentation, nous avons proposé des algorithmes basés régions, contours et hybrides, mettant en œuvre des métaheuristiques d'optimisation dans des approches mono et multiobjectif. Les méthodes proposées ont été évaluées sur des bases de données expérimentales composées d'images synthétiques, d'images IRM simulées et d'images IRM réelles ainsi que des images de tomographie par émission de positons (TEP). Les résultats obtenus sont significatifs et prouvent l'efficacité des idées proposées.

**Mots clés :** *segmentation d'image, classification, logique floue, post-segmentation, métaheuristiques, optimisation, optimisation multiobjectif, optimisation par essaim particulier (OEP), colonies de fourmis, imagerie médicale, images IRM, images TEP.*

---

# Abstract

Image segmentation is the process of partitioning a digital image into homogeneous non-overlapped regions with respect to some characteristics, such as gray value, motion, texture, etc. It is used in various applications like medical imaging, objects detection, biometric system, remote sensing, robot navigation, video surveillance, etc. The success of the machine vision system depends heavily on its performance, because characteristics and decisions are extracted and taken from its result. The first image segmentation algorithms were introduced in the 70's. Since then, various techniques and methods were experimented to improve the results. Nevertheless, up till now, no method produces a perfect result for a wide variety of images.

Metaheuristics are a high level procedure designed to solve hard optimization problems. These problems are in general characterized by their incomplete, uncertain or noised data, or faced to low computing capacity. Metaheuristics have been extremely successful in a wide variety of fields and demonstrate significant results. This is due to the fact that they can applied to solve any problem which can be formulated as an optimization problem. These methods are, mainly, inspired from physics (simulated annealing), biology (evolutionary algorithms), or ethology (particle swarm optimization, ant colony optimization).

In recent years, metaheuristics are starting to be exploited to solve segmentation problems with varying degrees of success and allow to consider the problem with different perspectives. Bearing this in mind, we propose in this work three segmentation and post-segmentation approaches based on mono or multiobjective optimization metaheuristics. The proposed methods were evaluated on databases containing synthetic images, simulated MRI images, real MRI images and PET images. The obtained results show the efficiency of the proposed ideas.

**Keywords :** *Image segmentation, data clustering, fuzzy logic, post-segmentation, metaheuristics, optimization, multiobjective optimization, particle swarm optimization, ant colony, medical imaging, MRI images, PET images.*

---

# Table des matières

|   |          |
|---|----------|
| <b>Introduction générale</b>  | <b>1</b> |
| <b>I État de l’art : segmentation d’image et métaheuristiques d’optimisation</b>                                  | <b>5</b> |
| I.1 Introduction . . . . .  | 5        |
| I.2 Les métaheuristiques d’optimisation . . . . .   | 5        |
| I.2.1 Les métaheuristiques à solution unique . . . . .  | 6        |
| I.2.1.1 La méthode de descente . . . . .  | 6        |
| I.2.1.2 La méthode du recuit simulé . . . . .   | 7        |
| I.2.1.3 La méthode de recherche tabou . . . . .   | 8        |
| I.2.2 Les métaheuristiques à population de solutions . . . . .  | 10       |
| I.2.2.1 Les algorithmes évolutionnaires . . . . .   | 10       |
| I.2.2.2 Algorithmes d’intelligence en essaim . . . . .  | 12       |
| I.2.2.2.1 La méthode des essais particuliers . . . . .  | 12       |
| I.2.2.2.2 La méthode des colonies de fourmis . . . . .  | 13       |
| I.2.3 L’optimisation multiobjectif . . . . .  | 14       |
| I.2.3.1 Méthodes d’agrégation . . . . .   | 14       |
| I.2.3.2 Méthodes basées sur l’approche de Pareto . . . . .  | 16       |
| I.2.3.3 Méthodes non Pareto et sans agrégation . . . . .  | 17       |
| I.3 Les méthodes de segmentation d’image et leurs améliorations par les métaheuristiques d’optimisation . . . . . | 17       |
| I.3.1 Méthodes de segmentation par seuillage . . . . .  | 18       |
| I.3.1.1 Améliorations par les métaheuristiques d’optimisation . . . . .   | 19       |

|            |   |           |
|------------|---|-----------|
| I.3.2      | Méthodes de segmentation par croissance de régions . . . . .  | 20        |
| I.3.2.1    | Améliorations par les métaheuristiques d’optimisation . . . . .   | 20        |
| I.3.3      | Méthodes de segmentation par chaînes de Markov . . . . .  | 20        |
| I.3.3.1    | Améliorations par les métaheuristiques d’optimisation . . . . .   | 22        |
| I.3.4      | Méthodes de segmentation par modèles déformables . . . . .  | 22        |
| I.3.4.1    | Améliorations par les métaheuristiques d’optimisation . . . . .   | 23        |
| I.3.5      | Méthodes de segmentation par classification . . . . .   | 23        |
| I.3.5.1    | Améliorations par les métaheuristiques d’optimisation . . . . .   | 25        |
| I.4        | Conclusion . . . . .  | 26        |
| <b>II</b>  | <b>Contribution à la segmentation d’image par classification floue optimisée par essai particulaire mono-objectif</b> | <b>27</b> |
| II.1       | Introduction . . . . .  | 27        |
| II.2       | Segmentation d’image via l’algorithme des c-moyennes floues . . . . .   | 28        |
| II.3       | Contribution à l’amélioration de l’algorithme des c-moyennes floues . . . . .   | 30        |
| II.3.1     | Représentation et initialisation . . . . .  | 30        |
| II.3.2     | Processus de segmentation . . . . .   | 33        |
| II.3.3     | Post-segmentation . . . . .   | 36        |
| II.4       | Résultats expérimentaux . . . . .   | 39        |
| II.4.1     | Résultats sur des images synthétiques . . . . .   | 42        |
| II.4.2     | Résultats sur des images IRM simulées . . . . .   | 43        |
| II.5       | Conclusion . . . . .  | 44        |
| <b>III</b> | <b>Contribution à la segmentation d’image par classification floue optimisée par essai particulaire multiobjectif</b> | <b>51</b> |
| III.1      | Introduction . . . . .  | 51        |
| III.2      | Optimisation multiobjectif par essai particulaire (MOPSO) . . . . .   | 52        |
| III.2.1    | L’optimisation multiobjectif . . . . .  | 52        |
| III.2.2    | Optimisation multiobjectif par essai particulaire . . . . .   | 53        |
| III.3      | Contribution à la segmentation d’image par classification multiobjectif . . . . .                                     | 55        |
| III.3.1    | Représentation et initialisation . . . . .  | 55        |
| III.3.2    | Critères de segmentation . . . . .  | 55        |
| III.3.3    | Processus d’optimisation . . . . .  | 57        |
| III.3.4    | Combinaison des solutions du front de Pareto . . . . .  | 57        |
| III.3.4.1  | Détection des pixels potentiellement mal classés . . . . .  | 57        |

---

|  |           |
|--|-----------|
| III.3.4.2 Affectation des pixels potentiellement mal classés . . . . .               | 58        |
| III.4 Résultats expérimentaux . . . . .  | 59        |
| III.4.1 Résultats sur des images synthétiques . . . . .                              | 60        |
| III.4.2 Résultats sur des images IRM simulées . . . . .                              | 61        |
| III.4.3 Résultats sur des images IRM réelles . . . . .                               | 62        |
| III.5 Conclusion . . . . .   | 63        |
| <b>IV Post-segmentation d'image via la métaheuristique des colonies de fourmis</b>   | <b>73</b> |
| IV.1 Introduction . . . . .  | 73        |
| IV.2 Optimisation par colonies de fourmis . . . . .                                  | 74        |
| IV.2.1 Système de fourmis "Ant System" (AS) . . . . .                                | 76        |
| IV.2.2 Système de colonie de fourmis "Ant Colony System" (ACS) . . . . .             | 77        |
| IV.3 Contribution à la segmentation d'image par colonie de fourmis . . . . .         | 78        |
| IV.3.1 Représentation et initialisation . . . . .                                    | 78        |
| IV.3.2 Processus de construction . . . . .   | 78        |
| IV.3.3 Mises à jour . . . . .  | 79        |
| IV.4 Résultats expérimentaux . . . . .   | 82        |
| IV.5 Conclusion . . . . .  | 83        |
| <b>V Application aux images de type "Tomographie par Émission de Positons" (TEP)</b> | <b>86</b> |
| V.1 Introduction . . . . .   | 86        |
| V.2 La tomographie par émission de positons . . . . .                                | 86        |
| V.2.1 Principe de l'imagerie TEP . . . . .   | 87        |
| V.2.2 Isotopes radioactifs . . . . .   | 88        |
| V.2.3 Radiotraceurs . . . . .  | 89        |
| V.2.4 Système de détection . . . . .   | 89        |
| V.3 Caractéristiques des images TEP . . . . .  | 90        |
| V.3.1 Le bruit . . . . .   | 90        |
| V.3.2 La résolution spatiale . . . . .   | 93        |
| V.3.3 L'effet de volume partiel . . . . .  | 93        |
| V.4 Résultats expérimentaux . . . . .  | 94        |
| V.4.1 Résultats sur des images de fantômes . . . . .                                 | 94        |
| V.4.2 Résultats sur les examens cliniques . . . . .                                  | 96        |
| V.4.2.1 Données . . . . .  | 96        |

---

Table des matières

---

|  |                                  |            |
|--|----------------------------------|------------|
| V.4.2.2                                    | Problématique générale . . . . . | 96         |
| V.4.2.3                                    | Résultats . . . . .              | 97         |
| V.5  | Conclusion . . . . .             | 97         |
| <b>Conclusion générale et perspectives</b> |                                  | <b>105</b> |
| <b>Références bibliographiques</b>         |                                  | <b>108</b> |

---

# Introduction générale

La vue représente un sens prépondérant chez l'être humain et fournit la plupart des informations nécessaires à nos interactions avec notre environnement. Elle représente à elle seule 60 à 90% de nos apports sensoriels. À partir de la vue, un être humain peut apprendre, reconnaître, comprendre, planifier, prendre des décisions et avoir même des émotions (en regardant des œuvres d'art). Ainsi, l'image représente l'une des sources les plus riches en informations. À cause des diverses possibilités d'actions qu'offrent ces informations, dans une multitude de domaines, il y a eu un fort engouement pour la recherche dans le domaine de la vision par ordinateur, en particulier depuis l'avènement des images numériques. Jusqu'à nos jours, ce domaine ne cesse de susciter de l'intérêt et reste l'un des domaines les plus étudiés. Il existe principalement deux niveaux de traitement automatique d'images : un traitement bas niveau consacré à l'acquisition, la compression, la segmentation, la restauration et l'amélioration, et un traitement haut niveau visant la reconnaissance et l'interprétation de l'image.

L'imagerie médicale est l'un des domaines qui profitent le mieux des techniques de traitement et d'analyse d'image. Elle permet l'acquisition d'images d'attributs inaccessibles autrement. Ces attributs du corps humain peuvent être anatomiques, physiologiques ou métaboliques. Les applications fondamentales de l'imagerie médicale sont : l'aide au diagnostic, l'aide à l'intervention, l'aide au suivi thérapeutique, l'amélioration des connaissances du corps humain et de son fonctionnement, la sécurité et l'archéologie.

L'une des opérations les plus importantes du traitement automatique des images est la segmentation d'image. Elle est présente dans la quasi-totalité des systèmes de vision par ordinateur. Son efficacité affecte directement les performances de ces derniers, car elle représente le

pivot entre les traitements bas niveau et les traitements haut niveau. À cause de son importance et ses difficultés, l'étude de la segmentation d'images continue à s'amplifier et à susciter beaucoup d'intérêt depuis les années 1970. Son objectif est d'étiqueter les différents objets présents dans une image selon certains critères, qui dépendent de l'application visée. Principalement, les critères utilisés sont : l'homogénéité des niveaux de gris, de la couleur, de la texture ou du mouvement. Du fait de la complexité du problème de la segmentation (démonstré problème mal posé) et de la grande variabilité des images, un très grand nombre de méthodes de segmentation automatique des images sont et continuent à être proposées dans la littérature. Certains chercheurs ironisent même, en disant que l'on se rapproche d'une méthode par image. Ainsi, il est évident que l'on est loin de la méthode idéale permettant de segmenter une large variété d'images.

Dans cette thèse, nous nous intéressons à la segmentation d'images au moyen de métaheuristiques d'optimisation. Ces algorithmes sont des stratégies générales visant la résolution d'une large variété de problèmes d'optimisation difficiles, souvent non solubles au moyen de méthodes exactes. Ces problèmes sont généralement des problèmes aux données incomplètes, incertaines ou bruitées ou des problèmes confrontés à une capacité de calcul limitée. La force des métaheuristiques réside dans leur caractère stochastique leur permettant d'exploiter des espaces de recherche très vastes et leur acceptation de solutions dégradées leur permettant de s'affranchir du problème de blocage dans les minima locaux. Ces méthodes sont, pour la plupart, inspirées de la physique (recuit simulé), de la biologie (algorithmes évolutionnaires) ou de l'éthologie (essaims particuliers, colonies de fourmis, etc.).

Les méthodes de segmentation sont souvent basées sur la minimisation d'un critère ou l'estimation de paramètres (modèles). Lorsqu'elles sont confrontées à des cas très difficiles, ces méthodes font appel soit à des hypothèses simplificatrices sur les modèles (ex. : Gaussienne) soit à des critères de segmentation et des attributs de faible complexité. Contrairement à ces méthodes, dites classiques, les métaheuristiques présentent l'avantage de traiter le problème de la segmentation d'image dans sa globalité, grâce à leur capacité à trouver des solutions sous-optimales proches des optima globaux dans des temps de calcul raisonnables. Ainsi, le problème de la segmentation peut être traité sous un angle différent, qui ne nécessite pas le recours à des simplifications a priori.

Cette thèse a été préparée au sein du Laboratoire Images, Signaux et Systèmes Intelligents (LiSSi, E.A. 3956) de l'Université de Paris-Est Créteil (UPEC). Ce travail de recherche a été

proposé et effectué sous la direction du Pr Patrick Siarry et co-encadré par M. Hamouche Oulhadj dans le groupe SIMO (Signal, IMage et Optimisation) du LISSI. Cette équipe s'intéresse, d'une part, au développement de méthodes d'imagerie médicale et d'analyse d'images médicales, et d'autre part, au développement de méthodes d'optimisation.

L'objectif de cette thèse est d'améliorer les performances des méthodes de segmentation d'images grâce à l'apport des métaheuristiques d'optimisation. Nous nous sommes intéressés principalement à l'optimisation de la méthode de segmentation par classification floue. Les améliorations apportées à cette approche nous ont conduits à développer deux nouvelles méthodes de segmentation. La première méthode vise à améliorer le critère de segmentation de la méthode des C-moyennes floues, la convergence de l'algorithme et la qualité des résultats de la segmentation, grâce à l'intégration d'un bloc de post-segmentation. La deuxième méthode proposée est une approche de segmentation multiobjectif, dont l'objectif est d'exploiter simultanément deux critères complémentaires de segmentation : un critère flou basé région et un deuxième critère basé contours. De plus, nous avons proposé une nouvelle technique de sélection de la solution finale de segmentation basée sur la combinaison des solutions du front de Pareto fournies par l'approche multiobjectif. Enfin, nous avons conçu une nouvelle méthode de post-segmentation qui met en œuvre l'algorithme des colonies de fourmis, afin de raffiner les résultats de n'importe quelle méthode de segmentation, en détectant et corrigeant les pixels mal classés.

Ce manuscrit est organisé comme suit :

Le premier chapitre présente une étude bibliographique où sont exposées les différentes métaheuristiques d'optimisation puis les différentes familles de méthodes de segmentation, ainsi que leur amélioration par ces techniques d'optimisation.

Dans le deuxième chapitre, nous présentons notre première contribution, qui est l'amélioration de la segmentation par classification floue grâce à la métaheuristique d'optimisation par essaim particulaire, mise en œuvre dans une approche mono-objectif.

Dans le troisième chapitre, nous exposons une extension de la première contribution vers une approche multiobjectif, où un critère complémentaire basé contour est introduit.

Le quatrième chapitre est dédié à la présentation d'une nouvelle méthode de post-segmentation basée sur l'algorithme des colonies de fourmis.

Dans le cinquième chapitre, nous nous penchons sur l'application des différentes méthodes proposées pour traiter le problème de la segmentation des images de type "Tomographie par Émission de Positons" (TEP).

Enfin, ce manuscrit s'achève par une conclusion générale, où nous récapitulons nos contributions et proposons des perspectives, sur la base des travaux effectués.

---

---

# Chapitre I

---

## État de l'art : segmentation d'image et métaheuristiques d'optimisation

### I.1 Introduction

Les métaheuristiques d'optimisation et la segmentation d'image font partie des domaines de recherche les plus actifs. Ces dernières années, plusieurs travaux ont créé le pont entre ces deux champs de recherche. En effet, le problème de segmentation d'image est formalisé comme un problème d'optimisation combinatoire. D'où l'utilisation des métaheuristiques. De plus, l'efficacité des métaheuristiques pour résoudre des problèmes de grande taille a rendu possibles le traitement de plus grandes quantités d'informations, de caractéristiques d'image et de critères de segmentation.

Dans ce chapitre, sans prétendre l'exhaustivité, nous présentons les principales métaheuristiques d'optimisation en premier lieu (section I.2). Ensuite, nous présentons les importantes familles de méthodes de segmentation, ainsi que leur amélioration à l'aide des métaheuristiques d'optimisation (section I.3).

### I.2 Les métaheuristiques d'optimisation

Les *métaheuristiques* sont des procédures de haut niveau conçues pour résoudre des problèmes d'optimisation dits difficiles. Ce sont en général des problèmes aux données incomplètes, incertaines, bruitées ou confrontés à une capacité de calcul limitée. Les métaheuristiques ont rencontré le succès dans une large variété de domaines. Cela découle du fait qu'elles peuvent être appliquées à tout problème pouvant être exprimé sous la forme d'un problème d'optimisation de critère(s). Ces méthodes sont, pour la plupart, inspirées de la physique (recuit simulé), de la bio-

logie (algorithmes évolutionnaires) ou de l'éthologie (essais particuliers, colonies de fourmis). Dans ce qui suit, nous présentons brièvement les principales métaheuristiques d'optimisation. Nous les classons en deux catégories : métaheuristiques à solution unique et métaheuristiques à population de solutions.

### I.2.1 Les métaheuristiques à solution unique

Les méthodes à solution unique, plus connues sous le nom de *méthodes de recherche locale* ou encore *méthodes de trajectoire*, sont basées sur l'évolution d'une seule solution dans l'espace de recherche. Typiquement, les méthodes de recherche locale démarrent d'une solution unique, puis à chaque itération, la solution courante est déplacée dans un voisinage local en espérant améliorer la fonction objectif. Les méthodes à solution unique englobent principalement la méthode de descente, la recherche locale itérée, la recherche à voisinage variable, la méthode du recuit simulé et la recherche tabou. Ces méthodes diffèrent essentiellement dans leur manière d'exploiter le voisinage local, i.e, choix des candidats et critères de déplacement. Dans cette section, nous présentons les méthodes les plus utilisées dans les problèmes de segmentation d'image, notamment la méthode de descente, le recuit simulé et la recherche tabou.

#### I.2.1.1 La méthode de descente

La méthode de descente représente la technique la plus intuitive et la plus simple dans le domaine de l'optimisation. Appelée aussi *hill climbing* (par référence à un cas de maximisation de la fonction objectif), cette méthode fonctionne selon le principe suivant : en démarrant d'une solution aléatoire, à chaque itération la solution courante se déplace vers une meilleure solution présente dans son voisinage jusqu'à ce qu'aucune amélioration ne soit plus possible. Le choix des solutions candidates dans le voisinage peut se faire de deux manières, soit évaluer la *fitness* de la fonction objectif pour toutes les solutions présentes dans le voisinage puis se déplacer vers la meilleure, soit choisir une solution aléatoire dans le voisinage, puis se déplacer vers cette solution si elle améliore le critère, ou sinon choisir une nouvelle solution. La procédure générale de cette méthode est présentée dans l'Algorithme I.1.

L'inconvénient majeur de la méthode de descente est qu'elle se trouve piégée dans le premier optimum local rencontré. L'amélioration possible de cette méthode consiste à lancer la méthode plusieurs fois en démarrant de différentes solutions initiales, ou dans un schéma plus complexe de réitération, comme *la recherche itérée* [Lourenço et al., 2001]. L'inconvénient majeur de cette méthode peut, dans certains cas, s'avérer un atout considérable (ex. l'espace de recherche est

---

### Algorithme I.1 Algorithme de descente

---

**Nécessite :** La fonction objectif  $f$

- 1: Générer une solution aléatoire  $S$
  - 2: Calculer la *fitness*  $f(S)$  associée à la solution initiale  $S$
  - 3: Initialiser la solution optimale :  $S_{opt} \leftarrow S$
  - 4: **Tant que** la condition d'arrêt n'est pas vérifiée **Faire**
  - 5:     Générer la liste des solutions dans le voisinage de la solution courante
  - 6:     Trouver la meilleure solution  $S'$  parmi les solutions voisines
  - 7:     **Si**  $f(S') < f(S)$  **Alors**
  - 8:          $S \leftarrow S'$
  - 9:          $S_{opt} \leftarrow S$
  - 10:    **Fin Si**
  - 11: **Fin tant que**
  - 12: **Retourner :** La solution optimale  $S_{opt}$
- 

convexe). De plus, la méthode de descente peut être utilisée comme première approche dans un problème d'optimisation pour avoir une idée sur les optima locaux du problème.

#### I.2.1.2 La méthode du recuit simulé

La méthode du recuit simulé (en anglais "Simulated Annealing" (SA)) est une métaheuristique d'optimisation inspirée de la technique de simulation de Metropolis [Metropolis et al., 1953] en mécanique statistique basée sur la distribution de Boltzmann. Le processus de recuit en métallurgie consiste à appliquer des cycles de chauffage et de refroidissement contrôlés à un matériau, afin de réorganiser sa structure cristallographique. Le but de cette opération est d'obtenir un matériau homogène de bonne qualité. À l'inverse de la technique de la trempe qui donne naissance à un état métastable, ce procédé permet d'éviter cet état caractéristique des minima locaux de l'énergie.

La transposition de la technique du recuit physique à la métaheuristique d'optimisation du recuit simulé est basée sur les analogies suivantes : la fonction objectif à optimiser est assimilée à l'énergie du matériau et la température est représentée par un paramètre de contrôle définissant le schéma de refroidissement.

L'algorithme d'optimisation du recuit simulé a été proposé pour la première fois par Kirkpatrick *et al.* [Kirkpatrick et al., 1983], et indépendamment par Černý [Černý, 1985]. Le principe de cet algorithme est le suivant : une configuration (solution) initiale est générée aléatoirement

(ou par une heuristique), puis à chaque itération la solution courante est déplacée de façon aléatoire dans son voisinage local, générant ainsi une nouvelle solution. Si cette nouvelle solution est meilleure que la précédente, elle est retenue. Sinon elle est acceptée avec une certaine probabilité. Cette étape est répétée tant que l'équilibre thermodynamique n'est pas atteint. Une fois l'équilibre atteint, la température est diminuée jusqu'à un nouvel équilibre. Et ainsi de suite, jusqu'à une condition d'arrêt (nombre d'itérations maximal atteint ou "système figé").

La méthode du recuit simulé est présentée dans l'Algorithme I.2. Au début de la recherche, la température est élevée, autorisant ainsi des dégradations importantes de la fonction objectif. Au fil des itérations, la température diminue, ce qui réduit la probabilité d'acceptation des dégradations de la *fitness*. Quand la température tend vers 0, seules les améliorations sont acceptées. Ce schéma se traduit par une exploration (diversification) de l'espace de recherche, en début de recherche, et une exploitation (intensification) des régions prometteuses, en fin de recherche.

### I.2.1.3 La méthode de recherche tabou

La méthode de recherche tabou (en anglais "Tabu Search" (TS)) a été proposée pour la première fois par Glover en 1986 [Glover, 1986]. Les principales idées de la recherche tabou utilisées actuellement sont présentées dans les références [Glover, 1989, 1990]. L'objectif de cette méthode est d'introduire un peu d'intelligence dans le processus de recherche, afin de ne pas se vouer totalement au hasard, au risque de faire des cycles menant toujours vers le même optimum local. En effet, la méthode de recherche tabou possède une mémoire où sont stockées les leçons tirées du passé. Par ailleurs, cette méthode, dans sa version originale, est dépourvue du caractère stochastique.

Plus concrètement, la méthode TS utilise une mémoire où sont enregistrés les derniers déplacements effectués (ou leurs attributs). L'ensemble de ces déplacements, qu'il est interdit d'effectuer en sens inverse, est regroupé dans la "liste taboue". Cela permet d'éviter le phénomène de cyclage et dirige la recherche vers de nouvelles régions non visitées encore. De même que toutes les méthodes de recherche locale, la recherche tabou nécessite un déplacement de la solution courante vers une nouvelle solution présente dans l'ensemble de ses voisines. À l'inverse de la méthode SA qui choisit aléatoirement une seule solution dans son voisinage, la méthode TS explore généralement l'ensemble de son voisinage avant de choisir une direction. Ainsi, après évaluation de la fonction objectif pour toutes les solutions voisines, la meilleure solution non

---

### Algorithme I.2 Algorithme du recuit simulé

---

**Nécessite :** La fonction objectif  $f$ , la température maximale  $T_{max}$ , la température minimale  $T_{min}$  et la fonction de diminution de la température  $abaisser(T)$ .

- 1: Initialiser la température :  $T \leftarrow T_{max}$
  - 2: Générer une solution aléatoire  $S$
  - 3: Calculer la *fitness*  $f(S)$  associée à la solution initiale  $S$
  - 4: Initialiser la solution optimale :  $S_{opt} \leftarrow S$
  - 5: **Tant que**  $T > T_{min}$  **Faire**
  - 6:     **Tant que** l'équilibre thermodynamique n'est pas atteint **Faire**
  - 7:         Tirer une nouvelle solution  $S'$  dans le voisinage de  $S$
  - 8:         Calculer la variation d'énergie :  $\Delta f \leftarrow f(S') - f(S)$
  - 9:         **Si**  $\Delta f \leq 0$  **Alors**
  - 10:             Accepter la nouvelle solution :  $S \leftarrow S'$
  - 11:         **Sinon**
  - 12:             **Si**  $\exp\left(-\frac{\Delta f}{T}\right) > x$  aléatoire  $\in [0, 1]$  **Alors**
  - 13:                 Accepter la nouvelle solution :  $S \leftarrow S'$
  - 14:             **Fin Si**
  - 15:         **Fin Si**
  - 16:         **Si**  $f(S') < f(S)$  **Alors**
  - 17:             Mettre à jour la solution optimale :  $S_{opt} \leftarrow S'$
  - 18:         **Fin Si**
  - 19:     Abaisser la température  $T$  :  $T \leftarrow abaisser(T)$
  - 20:     **Fin tant que**
  - 21: **Fin tant que**
  - 22: **Retourner :** La solution optimale  $S_{opt}$
- 

taboue remplace la solution actuelle. Si un optimum local est atteint, la meilleure solution présente dans le voisinage est choisie, même si elle n'améliore pas la solution courante ou si elle est présente dans la liste taboue. Ce mécanisme appelé *mécanisme d'aspiration* permet d'améliorer la solution actuelle et/ou de visiter de nouvelles régions qui ne peuvent être atteintes qu'en passant par des chemins déjà visités.

La taille de la mémoire réservée à la liste des déplacements tabous agit comme un paramètre de compromis entre diversification et intensification. Une mémoire de taille relativement faible favorise l'intensification, car pour un nombre restreint de mouvements bannis, la recherche aura vocation à visiter souvent les mêmes solutions. En revanche, si la taille de la mémoire augmente, la diversification est favorisée, dans la mesure où de nouvelles régions ont une plus forte probabilité d'être visitées. Un algorithme générique de la méthode de recherche tabou est donné dans l'Algorithme I.3.

---

### Algorithme I.3 Algorithme de la recherche tabou

---

**Nécessite :** La fonction objectif  $f$  et la taille de la liste taboue

- 1: Générer une solution aléatoire  $S$
  - 2: Calculer la *fitness*  $f(S)$  associée à la solution initiale  $S$
  - 3: Initialiser la solution optimale :  $S_{opt} \leftarrow S$
  - 4: **Tant que** la condition d'arrêt n'est pas vérifiée **Faire**
  - 5:     Générer la liste des candidats non tabous par opération de voisinage
  - 6:     Trouver la meilleure solution  $S'$  parmi les candidats
  - 7:     **Si**  $f(S') < f(S)$  **Alors**
  - 8:          $S \leftarrow S'$
  - 9:          $S_{opt} \leftarrow S$
  - 10:    **Fin Si**
  - 11:    Mettre à jour la liste taboue
  - 12: **Fin tant que**
  - 13: **Retourner :** La solution optimale  $S_{opt}$
- 

## I.2.2 Les métaheuristiques à population de solutions

À l'inverse des méthodes de recherche à solution unique, les métaheuristiques à population de solutions sont des méthodes qui font évoluer simultanément un ensemble d'individus (solutions) dans l'espace de recherche, où chacun profite de l'expérience du groupe, de manière directe ou indirecte. Ces méthodes sont principalement inspirées du vivant. On peut distinguer deux catégories de métaheuristiques à population : les algorithmes évolutionnaires inspirés de la théorie de l'évolution de C. Darwin [Charles, 1859] et les algorithmes d'intelligence en essaim inspirés de l'éthologie<sup>1</sup> et de la biologie.

### I.2.2.1 Les algorithmes évolutionnaires

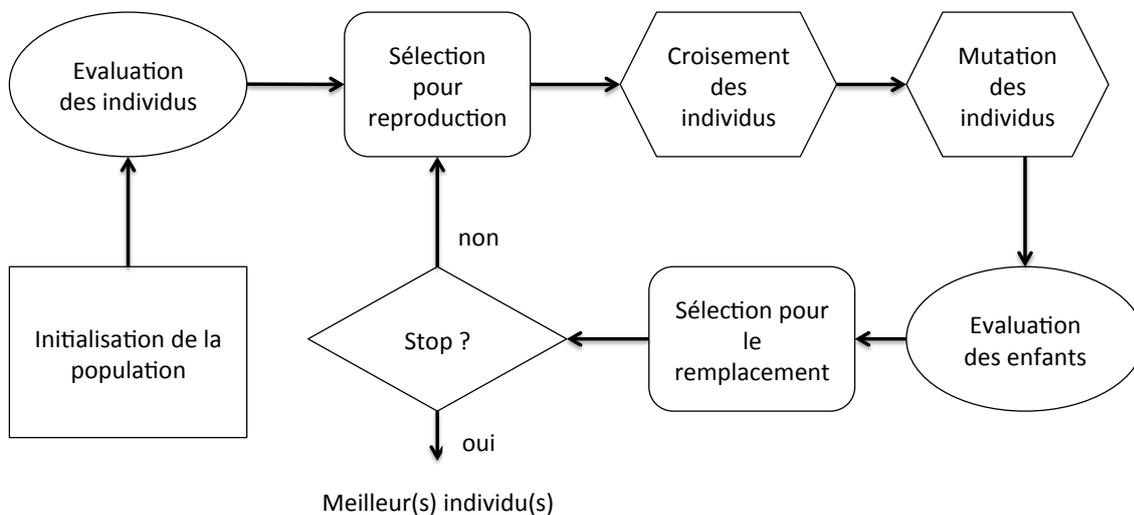
En 1859, C. Darwin a présenté une théorie basée sur le principe de sélection naturelle pour expliquer l'évolution des êtres vivants. La sélection naturelle est définie par son auteur comme suit : « J'ai donné à ce principe, en vertu duquel une variation si insignifiante qu'elle soit se conserve et se perpétue, si elle est utile, le nom de sélection naturelle ». Cette théorie repose sur trois principes : le principe de variation, qui est défini comme la différence entre les individus d'une population : aussi infime soit-elle, cette différence est nécessaire pour que la sélection naturelle opère. Le second principe est celui d'adaptation, qui se traduit par des variations des caractères des individus. Ces variations (mutations) donnent un avantage sélectif aux individus les plus adaptés à leur environnement. Le dernier principe est le principe d'hérédité, qui néces-

---

1. Étude du comportement des espèces animales. (Larousse)

site que les caractères soient transmissibles à la descendance.

Les algorithmes évolutionnaires ont commencé à faire leur apparition pour résoudre les problèmes d'ingénierie dès les années 1950, mais n'ont connu le succès qu'à la parution de ces quatre approches indépendantes : la stratégie d'évolution [Eigen, 1973], la programmation évolutionnaire [Fogel et al., 1966], les algorithmes génétiques [Holland, 1975] et la programmation génétique [Koza, 1992]. Toutes ces approches sont basées sur les mêmes principes de déroulement. La figure I.1 représente la procédure générale d'un algorithme évolutionnaire. Dans ce qui suit, nous nous contentons de décrire brièvement les algorithmes génétiques, car ils représentent indéniablement la méthode la plus utilisée des algorithmes évolutionnaires et des métaheuristiques de manière générale.



**Figure I.1** – Procédure générale d'un algorithme évolutionnaire (source : [Dréo et al., 2003]).

Les algorithmes génétiques ont été proposés initialement par J. Holland [Holland, 1975], mais ils n'ont connu la gloire qu'après 14 ans, avec la parution de l'ouvrage de référence de D.E. Goldberg [Goldberg, 1989]. Dans les algorithmes génétiques, les individus (population), sujets de l'évolution, sont un ensemble de solutions du problème à résoudre. À chaque itération, la population courante (parents) donne naissance à une nouvelle génération (enfants), en utilisant

des opérateurs de sélection et de variation. Les opérateurs de sélection permettent à un individu d'être choisi pour la reproduction (croisement) ou le remplacement. Généralement, plus un individu est performant, plus il a de chances d'être sélectionné. Les opérateurs de variation peuvent être classés en deux catégories : opérateurs de croisement et opérateurs de mutation. Les opérateurs de croisement, en combinant plusieurs parents (deux en général), créent un ou plusieurs enfants. Les opérateurs de mutation, en apportant des modifications à un seul individu, génèrent un nouvel individu.

### I.2.2.2 Algorithmes d'intelligence en essaim

Les comportements collectifs de certaines espèces, telles que les fourmis, les abeilles, les poissons, les oiseaux, etc. pour résoudre certains problèmes très complexes en effectuant de simples tâches individuelles, ont inspiré plusieurs chercheurs pour transposer ces comportements en algorithmes d'optimisation. Ces derniers sont nés de la modélisation mathématique des phénomènes éthologiques et biologiques [Bonabeau et al., 1999]. Les algorithmes d'intelligence en essaim qui ont connu le plus de succès sont l'optimisation par essaim particulaire et les algorithmes de colonies de fourmis, que nous décrivons dans les sections ci-dessous.

#### I.2.2.2.1 La méthode des essaims particulaires

L'optimisation par essaim particulaire (OEP) (*Particle Swarm Optimization* (PSO), en anglais) est une métaheuristique d'optimisation proposée pour la première fois par J. Kennedy et R. Eberhart en 1995 [Kennedy & Eberhart, 1995] en s'inspirant des modélisations statistiques développées par [Reynolds, 1987] et [Heppner & Grenander, 1990], qui permettent de simuler le déplacement de volées d'oiseaux et de bancs de poissons.

Dans l'algorithme d'essaim particulaire, la recherche s'effectue par une population d'individus appelés *particules*. Chaque particule *survolant* l'espace de recherche en quête de l'optimum global est considérée comme solution potentielle du problème. Afin de définir sa direction de vol, une particule se base sur deux types d'informations : une information tirée de sa propre expérience et une information tirée de l'expérience de l'essaim. Le mouvement des particules est régi par les équations suivantes (Eqs. I.1 et I.2) :

$$V^{(k+1)} = w.V^{(k)} + c_1.rand_1.(Pbest^{(k)} - X^{(k)}) + c_2.rand_2.(Gbest^{(k)} - X^{(k)}) \quad (I.1)$$

$$X^{(k+1)} = X^{(k)} + V^{(k+1)} \quad (\text{I.2})$$

où :

- $X$  : est la position des particules,
- $V$  : est la vitesse des particules,
- $w$  : est le paramètre d'inertie,
- $Pbest$  : est la meilleure position personnelle,
- $Gbest$  : est la meilleure solution de l'essaim,
- $rand_1, rand_2$  : sont des variables aléatoires entre 0 et 1,
- $c_1, c_2$  : sont des constantes positives,
- $k$  : est la variable d'itération.

Les trois termes de l'équation de vitesse peuvent être traduits comme suit :

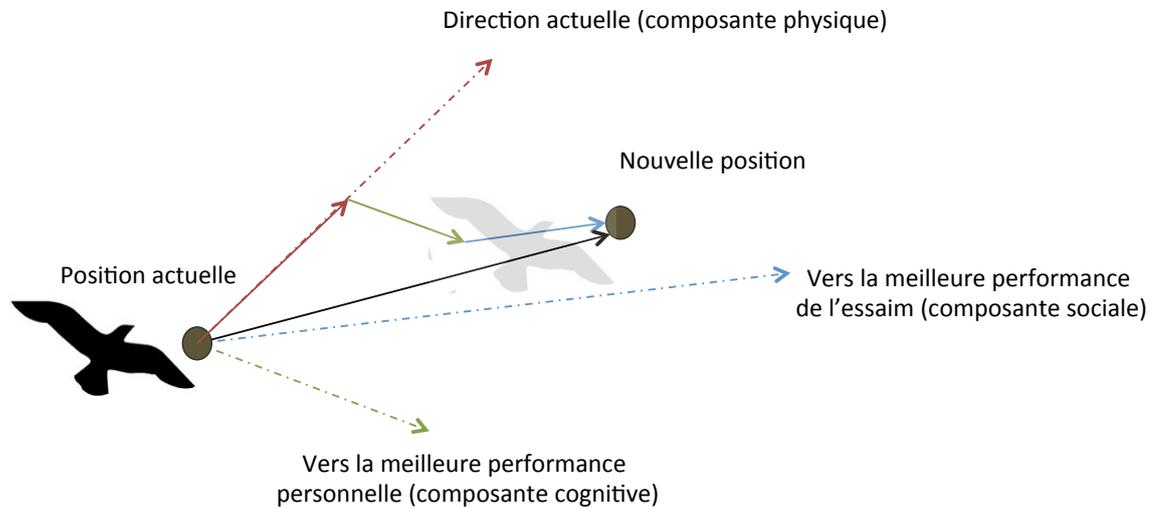
1.  $w.V^{(k)}$  : représente une composante physique d'inertie, qui incite chaque particule à suivre sa direction courante de déplacement,
2.  $c_1.rand_1 \cdot (Pbest^{(k)} - X^{(k)})$  : représente une composante cognitive, qui incite la particule à revenir vers le meilleur site qu'elle a déjà visité,
3.  $c_2.rand_2 \cdot (Gbest^{(k)} - X^{(k)})$  : représente une composante sociale, qui incite la particule à se diriger vers le meilleur site trouvé par ses congénères.

La figure [I.2](#) illustre un exemple de direction de vol d'une particule dans un espace de recherche.

La procédure générale des algorithmes d'optimisation par essaim particulaire est présentée dans l'Algorithme [I.4](#). Plusieurs améliorations de l'OEP ont été proposées dans la littérature, pour plus de détails, se référer à [[Clerc, 2010](#); [Siarry \(Ed.\), 2014](#)].

### I.2.2.2 La méthode des colonies de fourmis

Les algorithmes de colonies de fourmis sont des métaheuristiques d'optimisation inspirées du comportement collectif des fourmis dans leur processus de recherche de nourriture et d'optimisation du chemin entre leur nid et la source de nourriture trouvée. Une description détaillée de ces algorithmes est donnée dans la section [IV.2](#).



**Figure I.2** – Algorithme OEP : exemple de déplacement d'une particule dans l'espace de recherche.

### I.2.3 L'optimisation multiobjectif

Les méthodes d'optimisation exposées précédemment permettent d'optimiser un seul critère. Cependant, il arrive parfois qu'on ait besoin d'optimiser plusieurs objectifs simultanément. Ces objectifs sont souvent contradictoires, i.e. ils n'atteignent pas leur optimum pour la même solution. On parle dans ce cas d'*optimisation multiobjectif* ou d'*optimisation multicritère*. L'exemple le plus illustratif de l'optimisation multiobjectif est le problème de minimisation du coût de production d'un produit tout en maximisant sa qualité. L'optimisation multiobjectif a été amorcée au XIX<sup>ème</sup> siècle par [Edgeworth, 1885; Pareto, 1896] pour faire face aux problèmes économiques et industriels. La formalisation d'un problème multiobjectif est donnée dans la section III.2.

Il existe principalement trois approches pour résoudre un problème d'optimisation multiobjectif : approche d'agrégation, approche Pareto et approche non Pareto sans agrégation. Ces approches sont exposées dans les sections suivantes.

#### I.2.3.1 Méthodes d'agrégation

Les méthodes dites d'agrégation sont des techniques de transformation d'un problème d'optimisation multiobjectif en un problème d'optimisation mono-objectif. La manière la plus simple

### Algorithme I.4 OEP

---

**Nécessite :** La fonction objectif  $f$ , la taille de la population et les paramètres de l'équation (I.1).

- 1: Générer pour chaque particule une position et une vitesse initiale
  - 2: Evaluer pour chaque particule la *fitness* de la fonction objectif
  - 3: Affecter à la meilleure solution personnelle de chaque particule sa position initiale
  - 4: Déterminer la meilleure solution globale
  - 5: **Tant que** la condition d'arrêt n'est pas vérifiée **Faire**
  - 6:     Déplacer les particules selon les équations (I.1) et (I.2)
  - 7:     Evaluer la fonction objectif pour les nouvelles positions des particules
  - 8:     Mettre à jour les meilleures solutions personnelles des particules et la meilleure solution globale
  - 9: **Fin tant que**
  - 10: **Retourner :** La meilleure solution globale
- 

et la plus utilisée pour agréger plusieurs critères est d'utiliser une moyenne pondérée (Eq. I.3) :

$$F(x) = \sum_{i=1}^n w_i f_i(x) \quad (\text{I.3})$$

où :

$F(x)$  : est le nouveau critère mono-objectif à optimiser,

$f_i(x)$  : est le  $i^{\text{ème}}$  critère à optimiser,

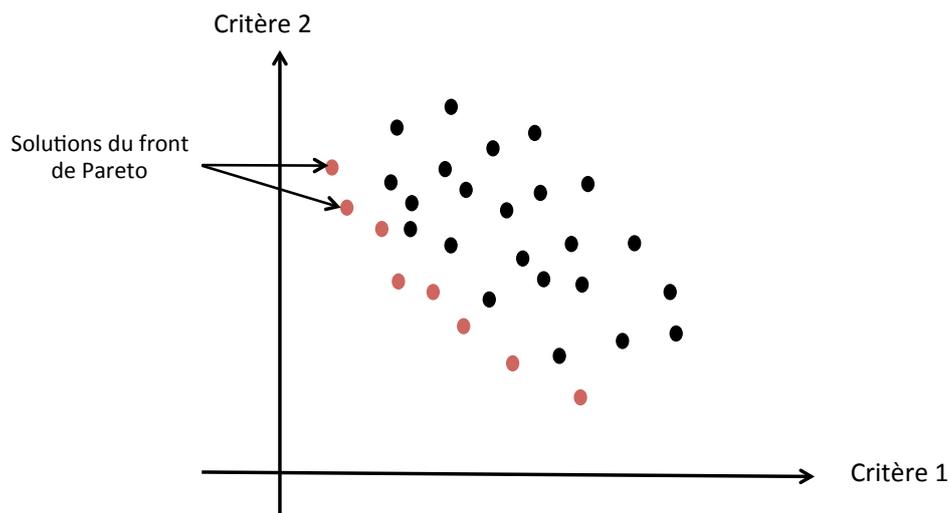
$w_i$  : est le poids associé à l' $i^{\text{ème}}$  critère  $f_i(x)$ . Ces poids sont positifs ( $w_i \geq 0$ ) et de somme unitaire ( $\sum_{i=1}^n w_i = 1$ ),

$n$  : est le nombre de critères à optimiser simultanément.

Sa simplicité de mise en œuvre et son efficacité ont rendu cette méthode largement utilisée. Son autre point fort réside dans le fait qu'elle fournit une solution unique ne nécessitant pas un opérateur de sélection. Néanmoins, la détermination des poids associés à chaque critère et l'interaction entre les différents objectifs constituent un inconvénient majeur des méthodes d'agrégation. De plus, dans le cas d'un front de Pareto non convexe, certaines solutions ne sont jamais accessibles [Ishibuchi & Murata, 1998]. En plus de la méthode d'agrégation linéaire, il existe plusieurs autres méthodes dans la littérature, les plus importantes sont : la méthode de  $\epsilon$ -contraintes [Ritzel et al., 1994], les méthodes de programmation par but (*Goal programming*) [Charnes & Cooper, 1977] et la méthode min-max [Coello Coello et al., 1995].

### I.2.3.2 Méthodes basées sur l'approche de Pareto

En optimisation mono-objectif, par simple évaluation de la fonction objectif, il est possible de comparer les qualités de diverses solutions. Cependant, en optimisation multiobjectif, il est parfois impossible de départager deux ou plusieurs solutions. Une solution peut être meilleure pour un objectif et piètre pour un autre, et inversement une solution peut s'avérer mauvaise pour le premier critère et meilleure pour le deuxième. On parle dans ce cas d'ensemble ou de front de Pareto : « Il existe un équilibre tel que l'on ne peut pas améliorer un critère sans détériorer au moins un des autres critères » [Pareto, 1896]. De manière plus concrète, le front de Pareto peut être défini comme l'ensemble des solutions, dont aucune n'est meilleure qu'une autre pour la totalité des critères. La figure I.3 illustre un exemple de front de Pareto pour un problème de minimisation de deux critères.



**Figure I.3** – Exemple de front de Pareto pour un problème de minimisation de deux critères.

Les méthodes basées sur l'approche Pareto tentent de faire converger les solutions vers l'ensemble du front de Pareto. Néanmoins, du fait de la production d'un ensemble de solutions il est nécessaire, pour ce type de méthode, de faire appel à un décideur pour retenir une seule solution finale. Plusieurs métaheuristiques ont adopté ce principe pour résoudre les problèmes d'optimisation multiobjectif. Parmi les plus avérées, on peut citer : la méthode *non-dominated sorting genetic algorithm* (NSGA) [Srinivas & Deb, 1994], la méthode NSGA-II [Deb et al., 2002], la méthode *multi objective genetic algorithm* (MOGA) [Fonseca et al., 1993], la méthode

## I.3 Les méthodes de segmentation d'image et leurs améliorations par les métaheuristiques d'optimisation

---

*multiobjective particle swarm optimization* [Coello et al., 2004] et la méthode AMOSA, basée sur le recuit simulé [Bandyopadhyay et al., 2008].

### I.2.3.3 Méthodes non Pareto et sans agrégation

Il existe un autre type de méthodes, qui ne sont basées ni sur l'agrégation ni sur le postulat de Pareto. Ces méthodes traitent principalement les objectifs séparément de manière itérative. Talbi [Talbi, 2009] classe ces méthodes en deux catégories : les approches parallèles et les approches séquentielles.

L'approche parallèle concerne principalement les métaheuristiques à population de solutions, où plusieurs sous-populations sont désignées, chacune ayant pour tâche d'optimiser un seul critère, tout en échangeant des informations entre elles. Parmi ces méthodes, on peut citer : l'algorithme génétique à évaluation vectorielle (VEGA) [Schaffer, 1985] et l'algorithme basé sur l'optimisation par essaim particulière, proposé par Parsopoulos *et al.* [Parsopoulos et al., 2004].

Dans les approches séquentielles, appelées aussi lexicographiques, la recherche est effectuée suivant un ordre préférentiel établi par un décideur. Plusieurs métaheuristiques ont été proposées dans la littérature en utilisant cette approche ; pour plus de détails, nous renvoyons le lecteur à [Collette & Siarry, 2003; Talbi, 2009].

## I.3 Les méthodes de segmentation d'image et leurs améliorations par les métaheuristiques d'optimisation

Dans cette section, nous présentons les principales approches de segmentation d'image et leurs améliorations à l'aide des métaheuristiques d'optimisation. Vu le nombre important et la diversité des méthodes de segmentation, nous n'utilisons aucune classification des méthodes de segmentation, mais nous nous contentons de donner le principe des différentes approches. En effet, du fait de cette variabilité, une classification perd son intérêt de synthétisation et c'est sans doute ce qui explique l'absence d'article consacré à un "état de l'art" des méthodes de segmentation depuis les années quatre-vingt-dix.

### I.3.1 Méthodes de segmentation par seuillage

Les méthodes de segmentation par seuillage (*thresholding*) sont des méthodes très simples, qui ont connu un grand succès dans les systèmes réels, à cause de leur facilité de mise en œuvre et leur rapidité. Ce type de méthode se base sur l'hypothèse que les différentes régions de l'image peuvent être distinguées par leurs niveaux de gris. Ainsi, pour avoir une segmentation, il suffirait de trouver les valeurs des niveaux de gris délimitant chaque région (seuils). Toutes les méthodes de seuillage tentent de trouver les seuils optimaux, de sorte à séparer les différentes régions de l'image.

Les méthodes de seuillage utilisent principalement l'histogramme d'image, afin de déterminer les concentrations de pixels autour de certains niveaux de gris pour déterminer, par la suite, les différents seuils situés entre ces derniers. La figure I.4 représente un exemple illustratif de seuillage d'une image. On remarque qu'une fois les seuils déterminés, l'affectation des pixels s'effectue selon la formule suivante (Eq. I.4) :

$$I_{seg}(i, j) = k \quad \text{si} \quad S_k \leq I(i, j) < S_{k-1} \quad \text{pour} \quad k = 1, \dots, N \quad (\text{I.4})$$

où :

$I(i, j)$  : représente le niveau de gris du pixel ayant les coordonnées  $(i, j)$  dans l'image à segmenter,

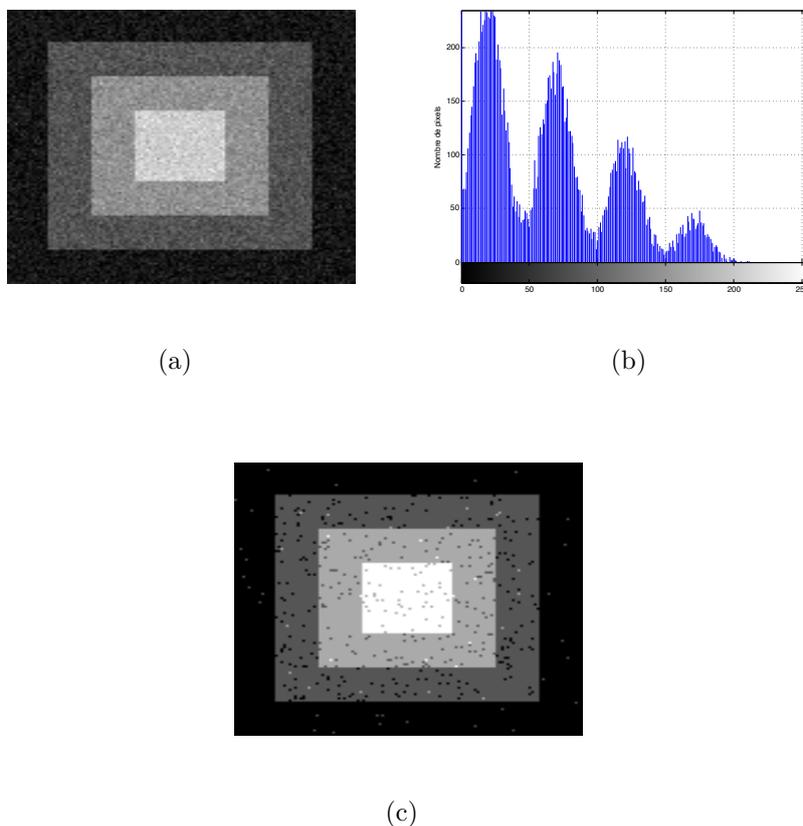
$I_{set}(i, j)$  : représente le label affecté au pixel ayant la position  $(i, j)$  dans l'image résultant de la segmentation,

$S_k$  : est le  $k^{\text{ème}}$  seuil,

$N$  : est le nombre de régions présentes dans l'image à segmenter.

L'exemple d'image montrée dans la figure I.4 est un exemple illustratif facile à segmenter. Or, en pratique, il est souvent plus compliqué de trouver les seuils à cause des recouvrements de classes et dans le cas où l'histogramme ne reflète pas le nombre de classes. Dans la littérature, plusieurs méthodes de seuillage ont été proposées [Sezgin & Sankur, 2004]. On peut les classer en deux catégories : les méthodes paramétriques et les méthodes non paramétriques.

Les méthodes paramétriques reposent sur l'hypothèse que les régions (classes) de l'image suivent des fonctions de densité de probabilité. Généralement, une combinaison de Gaussiennes est utilisée. La première étape consiste à estimer les paramètres des modèles, et la deuxième étape consiste à déterminer les seuils (intersection des modes).



**Figure I.4** – Exemple de seuillage d'une image synthétique faiblement bruitée par un bruit Gaussien. (a) Image à segmenter. (b) Histogramme de l'image. (c) résultat de la segmentation (seuils = [50, 100, 150]).

Les méthodes non paramétriques, comme leur nom l'indique, ne nécessitent pas d'estimation de paramètres pour trouver les seuils optimaux. Ces méthodes sont basées généralement sur l'optimisation d'un critère. Les méthodes non paramétriques de référence sont la méthode d'Otsu [Otsu, 1975], basée sur la minimisation du rapport entre la variance intraclasse et la variance interclasse, et la méthode Kapur [Kapur et al., 1985], basée sur la maximisation de l'entropie globale de Shannon.

### I.3.1.1 Améliorations par les métaheuristiques d'optimisation

Plusieurs auteurs ont exploité les métaheuristiques d'optimisation dans les méthodes de seuillage paramétriques, afin de trouver les paramètres optimaux des distributions approchant

l’histogramme, et dans les méthodes de seuillage non paramétriques, afin de trouver les seuils optimaux minimisant différents critères. Le recuit simulé a été utilisé dans les travaux de [Snyder et al., 1990; Nakib et al., 2007, 2008; Karasulu & Korukoglu, 2011]. La thèse de A. Nakib est dédiée à ce sujet [Nakib, 2008]. L’optimisation par colonies de fourmis a été utilisée par [Malisia & Tizhoosh, 2006; Zhao et al., 2008], et l’optimisation par essaim particulaire par [El Dor et al., 2014; Maitra & Chatterjee, 2008; Feng et al., 2005; Sathya & Kayalvizhi, 2010; Yin, 2007]. Les algorithmes génétiques ont été exploités par [Tao et al., 2003; Yin, 1999; Hongfu, 2002; Lai & Tseng, 2004; Bhanu et al., 1995]. De plus, d’autres métaheuristiques et leur hybridation ont été proposées par [Alihodzic & Tuba, 2014; Boussaïd et al., 2013; Lepagnot et al., 2011; Horng, 2011].

### I.3.2 Méthodes de segmentation par croissance de régions

Ces méthodes de segmentation consistent à choisir, en premier lieu, un ensemble de pixels représentatif de chaque région, appelés *germes*. Ensuite, les régions se construisent au fur et à mesure, en ajoutant les pixels connexes à leurs germes. Les pixels qui sont ajoutés à la région doivent vérifier un certain critère de similarité (ou de distance) entre les pixels à ajouter et la région construite. Plusieurs méthodes de croissance de régions ont été proposées dans la littérature, utilisant différents critères de similarité. Ces méthodes ne sont généralement pas utilisées seules, mais dans une chaîne de traitement [Pham et al., 2000]. Ces méthodes ont trois inconvénients majeurs : elles nécessitent l’intervention d’un opérateur pour initialiser les germes, utilisent des seuils pour décider d’ajouter un pixel ou pas et leur sensibilité aux bruits peut créer des trous à l’intérieur des régions. Plusieurs travaux ont tenté de pallier ces inconvénients, une majorité est regroupée dans l’article de Zhang [Zhang, 2006].

#### I.3.2.1 Améliorations par les métaheuristiques d’optimisation

Lors de notre étude bibliographique, nous n’avons trouvé qu’une seule méthode de croissance de régions améliorée par les métaheuristiques d’optimisation. I.N. Manousakas *et al.* [Manousakas et al., 1998] ont utilisé le recuit simulé dans la méthode *division fusion*, afin de réduire le nombre de régions pour la segmentation d’un volume IRM 3D.

### I.3.3 Méthodes de segmentation par chaînes de Markov

Les chaînes de Markov ne constituent pas, en elles-mêmes, des méthodes de segmentation, mais une modélisation statistique, qui peut être utilisée dans d’autres méthodes de segmentation

---

### I.3 La segmentation d'image et les métaheuristiques d'optimisations

---

[Pham et al., 2000]. Cette technique de modélisation, largement exploitée en traitement d'image, a été introduite par les travaux de S. Geman et D. Geman [Geman & Geman, 1984]. La segmentation d'image par les chaînes de Markov est basée sur le principe suivant : l'image est considérée comme une observation donnée, dont on cherche à connaître les objets (variables cachées) en disposant d'information à priori. Plus concrètement, on cherche la probabilité d'un objet en connaissant l'image et des informations à priori. En utilisant le théorème de Bayes, la probabilité à posteriori peut être calculée à partir de la probabilité à priori (Eq. I.5) :

$$P(x \in O_n | I(x)) = \frac{P(I(x) | x \in O_n) P(x \in O_n)}{\prod_{i=1}^n P(x \in O_i)} \quad (\text{I.5})$$

où :

$I(x)$  : représente l'intensité du pixel  $x$ ,

$O_i$  : représente le  $i^{\text{ème}}$  objet de l'image.

$P(x \in O_i)$  : est la probabilité que le pixel  $x$  appartienne à l'objet  $i$  avant d'avoir vu l'image, appelée la *probabilité à priori*,

$P(I(x) | x \in O_n)$  : est la probabilité de l'image sachant que le pixel  $x$  appartient à l'objet  $O_n$ , appelée la *vraisemblance*.

Afin de calculer la vraisemblance, l'histogramme normalisé de l'image peut être utilisé directement ou en l'estimant avec un estimateur de noyau (méthode de Parzen-Rozenblatt) pour réduire l'effet du bruit ou du sous-échantillonnage. Quant à la probabilité à priori, elle est introduite en utilisant les chaînes de Markov. Ces dernières permettent de modéliser la corrélation spatiale entre les pixels de l'image. En effet, la plupart des pixels de l'image appartiennent aux mêmes objets que leurs voisins. Le calcul de la probabilité à priori se fait à l'aide de la distribution de Gibbs (Eq. I.6) :

$$P(x \in O_n) = \frac{1}{Z} \cdot e^{-U(w)} \quad (\text{I.6})$$

où :

$Z$  : est une constante de normalisation, appelée aussi fonction de partition,

$U(w)$  : est une fonction de potentiel dans un système de voisinage local prédéfini. Elle augmente avec le nombre de pixels ayant la même appartenance dans le voisinage.

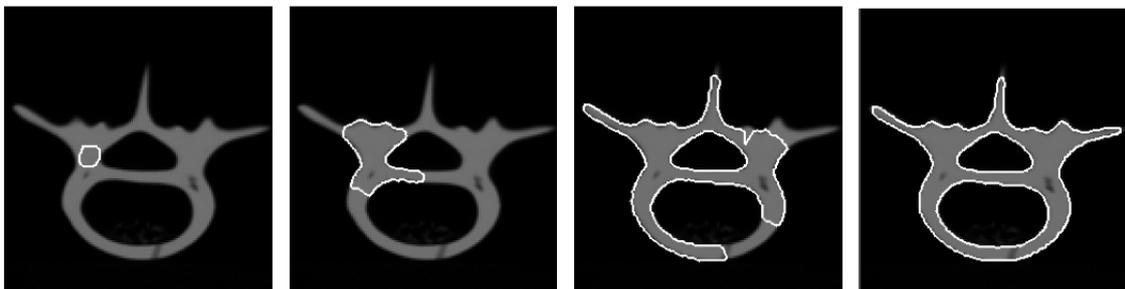
Finalement, l'objectif est de trouver l'estimation optimale de la segmentation connaissant l'image, qui est considérée comme un champ Markovien. Pour plus de détails sur ces méthodes, voir [Brunel, 2005].

### I.3.3.1 Améliorations par les métaheuristiques d'optimisation

La détermination des paramètres de la chaîne de Markov qui modélise l'image est un problème d'optimisation combinatoire. De ce fait, les métaheuristiques d'optimisation sont favorites pour résoudre ce problème. En ce sens, plusieurs travaux dans la littérature en ont été proposés en utilisant : le recuit simulé [Held et al., 1997; Lakshmanan & Derin, 1989], la recherche taboue [Zhang et al., 2008; Castellani et al., 2007], les algorithmes génétiques [Kim et al., 2000; Tseng & Lai, 1999; JoonKim et al., 1998] et les colonies de fourmis [Hegar-Masclé et al., 2007; Ouadfel & Batouche, 2003; Lu & Zhou, 2007]. Même si nous n'avons pas trouvé une exploration des autres métaheuristiques, plus récentes, dans les chaînes de Markov, nous pensons qu'elles ont un fort potentiel pour améliorer l'estimation des paramètres de ce type de modèles.

### I.3.4 Méthodes de segmentation par modèles déformables

Ces méthodes de segmentation ont pour but de déterminer les contours des objets présents dans l'image. Elles utilisent principalement des courbes paramétriques fermées ou des surfaces, qui partant d'une position initiale, se déforment jusqu'à épouser la forme de l'objet à détecter (cf. figure I.5). L'évolution de la courbe est régie par une force interne et une force externe jusqu'à atteindre un équilibre où l'énergie est minimale. La force interne est intrinsèquement liée à la courbe (ou la surface) elle-même, afin de la garder aussi lisse que possible. La force externe est définie par l'image, afin de guider la courbe ou la surface vers les contours désirés. Plusieurs méthodes, basées sur les méthodes déformables, ont été proposées dans la littérature. Parmi celles qui ont trouvé le plus de succès, on cite : les contours actifs (*snakes*) [Kass et al., 1988; Cohen & Cohen, 1993; McInerney & Terzopoulos, 2000], la méthode des surfaces de niveau (*level sets*) [Sethian, 1996; Cremers et al., 2007] et les contours actifs géodésiques [Caselles et al., 1997].



**Figure I.5** – Exemple d'évolution d'un modèle déformable (source : [McInerney & Terzopoulos, 1996]).

### I.3.4.1 Améliorations par les métaheuristiques d'optimisation

De même que les autres méthodes de segmentation, la segmentation par modèles déformables est basée sur l'optimisation d'un critère d'énergie. D'où l'utilisation des métaheuristiques d'optimisation. Le recuit simulé a été utilisé dans les travaux de [Storvik, 1994; Tang et al., 2007]. La recherche taboue a été utilisée par P. Yin [Yin, 2000]. Les algorithmes génétiques ont été exploités par [MacEachern & Manku, 1998]. L'optimisation par colonies de fourmis a été appliquée par [Wang et al., 2006]. L'optimisation par essaims particulaires a été proposée dans les travaux de [Tseng et al., 2009; Asl & Seyedin, 2006].

### I.3.5 Méthodes de segmentation par classification

La classification est une technique de reconnaissance de formes largement utilisée dans le domaine du traitement d'image [Chen et al., 2010]. Son objectif est de trouver une partition de l'espace des attributs des données [Friedman & Kandel, 1999]. L'espace des attributs (*feature space*) est l'ensemble des transformations des données brutes permettant de rehausser la séparabilité des différentes classes désirées. Plusieurs attributs ont été utilisés pour la segmentation d'image [Zhang, 1996]. Souvent, l'intensité des pixels est utilisée comme attribut des données, en introduisant les autres informations dans le processus de classification. Généralement, on distingue deux approches de classification : la classification supervisée et la classification non supervisée.

Les méthodes de classification supervisée nécessitent une base d'images déjà segmentées manuellement. Cette base permet de faire un apprentissage (*training*) du classifieur. Une fois les paramètres du classifieur déterminés, de nouvelles images peuvent être segmentées. L'exemple le plus simple de ces méthodes est le classifieur du plus proche voisin (*nearest-neighbor*) où chaque pixel est affecté à la même classe que celle du pixel présent dans la base d'apprentissage, ayant le plus proche niveau de gris.

À l'inverse des méthodes de classification supervisée, les méthodes de classification non supervisée, appelées aussi méthodes de regroupement (*clustering*), ne nécessitent pas une base d'images segmentées. Afin de compenser le manque d'une base d'apprentissage, ces méthodes alternent entre classification des pixels et caractérisation des classes ; en ce sens, les méthodes de regroupement apprennent d'elles-mêmes en utilisant les données disponibles [Pham et al., 2000]. Les méthodes de regroupement de données qui ont connu un large succès sont : l'algorithme des K-moyennes (*K-means* en anglais) ou ISODATA [Kanungo et al., 2002], l'algorithme espérance-

maximisation (*expectation-maximization* en anglais) [Makropoulos et al., 2012] et l'algorithme des C-moyennes floues (*fuzzy C-means* "FCM" en anglais) [Bezdek et al., 1992], sur lequel nous nous sommes basés dans cette thèse.

En se basant sur la théorie de la logique floue introduite par L. Zadeh [Zadeh, 1965], J. Bezdek a proposé pour la première fois l'algorithme des c-moyennes floues [Bezdek, 1981]. Très vite, cet algorithme a trouvé un large succès dans diverses applications [Gath et al., 1983; Granath, 1984; Huntsherger et al., 1985]. La méthode des C-moyennes floues a pour objectif de déterminer les centres des différentes classes, ainsi que les degrés d'appartenance de chaque pixel aux différentes classes. Les centres des différentes classes sont déterminés de telle sorte que la distance entre un centre de classe et les pixels appartenant à cette classe soit minimale (Eq. I.7) :

$$J = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(x_j, c_i) \quad (\text{I.7})$$

où :

$N$  : est le nombre de pixels de l'image,

$C$  : est le nombre de régions (classes) présentes dans l'image,

$u_{ij}$  : est le degré d'appartenance du pixel  $j$  à la classe  $i$ ,

$d(x_j, c_i)$  : est la mesure de distance entre le  $j^{\text{ème}}$  pixel de l'image et le centre de la classe  $i$ ,

$m$  : est le degré de flou.

Les degrés d'appartenance doivent vérifier les contraintes de logique floue. À savoir, un degré d'appartenance à une classe donnée est positif et ne peut dépasser 1, et la somme des degrés d'appartenance d'un pixel aux différentes classes vaut 1 (Eq. I.8).

$$\begin{cases} \forall j \in [1, N], \forall i \in [1, C] : u_{ij} \in [0, 1] \\ \forall j \in [1, N] : \sum_{i=1}^C u_{ij} = 1 \end{cases} \quad (\text{I.8})$$

Dans la version originale de la méthode des C-moyennes floues, le critère  $J$  est minimisé en utilisant les multiplicateurs de Lagrange. Cela garantit une convergence vers un minimum local.

À cause de son efficacité et de sa simplicité de mise en œuvre, la méthode FCM est largement utilisée dans les problèmes de segmentation d'image. Néanmoins, cette méthode présente les inconvénients suivants :

---

### I.3 La segmentation d'image et les métaheuristiques d'optimisations

---

- L'absence de l'information spatiale dans le critère de segmentation.
- L'utilisation de la distance Euclidienne, qui suppose que toutes les classes ont des formes hyper-sphériques à diamètre égal.
- La sensibilité aux solutions initiales. En effet, l'algorithme FCM ne fournit aucune garantie de convergence vers un optimum global.

Afin de pallier ces inconvénients, plusieurs travaux ont été proposés dans la littérature ; dans les paragraphes suivants, nous citons les principaux parmi eux.

Krishnapuram et Keller [Krishnapuram & Keller, 1993] ont proposé une approche possibiliste des C-moyennes floues (*possibilistic C-means* ou PCM), afin d'augmenter la robustesse de l'algorithme FCM vis-à-vis du bruit. Ils ont proposé un nouveau critère à minimiser permettant de remédier au caractère relatif des degrés d'appartenance, que génère l'algorithme FCM [Khodja, 1997].

Dans le but d'introduire la corrélation spatiale qui existe entre les pixels de l'image, plusieurs méthodes ont été proposées, en modifiant le critère de classification [Pham & Prince, 1999; Ahmed et al., 2002; Pham, 2002; Chen & Zhang, 2004; Szilagy et al., 2003; Tolia & Panas, 1998; Kang & Zhang, 2009; Cai et al., 2007; Krinidis & Chatzis, 2010; Fritz et al., 2013; Chuang et al., 2006; Ji et al., 2014; Liew & Yan, 2003]. Les modifications apportées consistent en général à contraindre un pixel à appartenir à la même classe que ses voisins.

Afin de pallier l'inconvénient de l'algorithme FCM dû à l'utilisation de la distance Euclidienne, D. Gustafson et W. Kessel [Gustafson & Kessel, 1978] ont proposé l'utilisation de la distance de Mahalanobis modifiée, en introduisant une matrice de covariance floue. I. Gath et A. Geva [Gath & Geva, 1989] ont proposé une distance proportionnelle à l'inverse d'une distribution Gaussienne. Plusieurs méthodes ont été inspirées de ces travaux et ont proposé des distances basées sur des noyaux [Huang et al., 2012; Wu et al., 2003; Zhang & Chen, 2002]. Une étude comparative de ces différentes méthodes est donnée par D. Graves et W. Pedrycz [Graves & Pedrycz, 2010].

#### I.3.5.1 Améliorations par les métaheuristiques d'optimisation

Les métaheuristiques d'optimisation ont été largement exploitées dans les problèmes de classification de manière générale, à cause de la nature combinatoire de ce problème. L'ouvrage de Das *et al.* [Das et al., 2009] est dédié spécialement à ce sujet. Plus récemment, Tsai *et al.*

[Tsai et al., 2014] ont proposé une étude des métaheuristiques appliquées aux problèmes de classification de données. Afin de remédier au problème de blocage dans des minima locaux de la méthode FCM, plusieurs travaux ont été proposés dans la littérature, qui exploitent les différentes métaheuristiques et leur hybridation [Maulik & Bandyopadhyay, 2000; Ng & Wong, 2002; Niknam & Amiri, 2010; Niknam et al., 2009; Izakian & Abraham, 2011; Liu et al., 2014].

### I.4 Conclusion

Dans ce chapitre, nous avons présenté les principales métaheuristiques d'optimisation et les méthodes de segmentation d'image. De plus, pour chaque famille de méthodes de segmentation d'image, nous avons cité les principaux travaux qui ont introduit les métaheuristiques d'optimisation.

Compte tenu du nombre croissant de ces travaux, l'utilisation des métaheuristiques pour résoudre les problèmes de segmentation d'image a fait ses preuves et s'est construite une solide base. Dans les chapitres suivants, nous présentons nos contributions dans ce domaine.

---

---

# Chapitre II

---

## Contribution à la segmentation d'image par classification floue optimisée par essaim particulaire mono-objectif

### II.1 Introduction

Dans ce chapitre, nous présentons une méthode de segmentation d'image basée sur la méthode de groupement des données c-moyennes floues (en anglais "*Fuzzy C-means*" (FCM)). La méthode FCM est largement utilisée en segmentation d'image, car, en plus de son efficacité, simplicité et rapidité, cette méthode est basée sur le principe de la logique floue, ce qui permet de définir des ensembles d'une manière plus souple, tolérant des informations imprécises, incomplètes et/ou incertaines. Ce qui est souvent le cas pour les images médicales. Cependant cette méthode présente quelques inconvénients liés au blocage dans des minima locaux et à la métrique de distance. La méthode que nous proposons dans ce chapitre permet de remédier à ces inconvénients.

Ce chapitre est organisé comme suit : dans la section [II.2](#) nous présentons la méthode FCM appliquée au problème de segmentation d'image, dans la section [II.3](#) on détaille la méthode proposée, dans la section [II.4](#) sont présentés les résultats expérimentaux et finalement nous terminons de chapitre par une conclusion présentée dans la section [II.5](#).

## II.2 Segmentation d'image via l'algorithme des c-moyennes floues

L'algorithme des c-moyennes floues (FCM) est une méthode de classification non-supervisée de données (regroupement de données) basée sur le principe de la logique floue où chaque exemple de données n'appartient pas uniquement à une seule classe, mais à toutes les classes avec certains degrés d'appartenance. Cette méthode fait partie des méthodes dites « méthodes des centres mobiles » où chaque classe est représentée par son centre (prototype), plus une donnée est proche d'un centre de classe, plus important est son degré d'appartenance à cette classe. L'objectif de cette méthode est donc de trouver à la fois les positions des centres des classes et les degrés d'appartenance des données aux différentes classes, en minimisant un critère quadratique de distance (Eq. (II.1)).

Soit un ensemble d'attributs de données  $X = \{x_1, x_2, \dots, x_N\}$ , l'algorithme des c-moyennes floues permet d'engendrer une partition de l'espace de ces données en calculant les centres des classes  $c_i$  et les degrés d'appartenance  $u_{ij}$  en minimisant la distance entre les données et les centres des classes :

$$J = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(x_j, c_i) \quad (\text{II.1})$$

sous les contraintes :

$$\begin{cases} \forall j \in [1, N], \forall i \in [1, C] : u_{ij} \in [0, 1] \\ \forall j \in [1, N] : \sum_{i=1}^C u_{ij} = 1 \end{cases} \quad (\text{II.2})$$

où :

$N$  : est le nombre d'exemples de la base de données,

$C$  : est le nombre de classes,

$u_{ij}$  : est le degré d'appartenance de l'exemple  $j$  à la classe  $i$ ,

$d(x_j, c_i)$  : est la mesure de distance entre le  $j^{\text{ème}}$  exemple et le centre de la classe  $i$ ,

$m$  : est le degré de flou.

Les contraintes du problème (Eqs. (II.2)) découlent directement du principe de la logique floue, à savoir le degré d'appartenance d'un exemple à une classe ne peut dépasser 1 et la somme des degrés d'appartenance d'un exemple à toutes les classes doit valoir 1.

---

## II.2 Segmentation d'image via l'algorithme des c-moyennes floues

---

Le problème d'optimisation de la fonction objectif (Eq. (II.1)) sous les contraintes (Eqs. (II.2)) est un problème mal-posé, car les degrés d'appartenance  $u_{ij}$  et les centres des classes  $c_i$  ne peuvent être déterminés simultanément. Afin de résoudre ce problème, une procédure d'alternance est utilisée. En effet, en première étape, les degrés d'appartenance sont déterminés en supposant les centres des classes fixes. En deuxième étape, les centres des classes sont déterminés en supposant les degrés d'appartenance fixes. Ces deux étapes sont répétées jusqu'à convergence de l'algorithme. Pour résoudre le problème d'optimisation avec contraintes, le problème est transformé en un problème sans contraintes à l'aide des multiplicateurs de Lagrange (Eq. (II.3)) :

$$L = J - \sum_{j=1}^N \lambda_j \left( \sum_{i=1}^C u_{ij} - 1 \right) \quad (\text{II.3})$$

*Etape 1* : Calcul des degrés d'appartenance en supposant les centres des classes fixes :

$$\frac{\partial L}{\partial u_{ij}} = 0 \implies u_{ij} = \frac{1}{\sum_{l=1}^C \left( \frac{d(x_j, c_i)}{d(x_j, c_l)} \right)^{\frac{2}{m-1}}} \quad (\text{II.4})$$

*Etape 2* : Calcul des centres des classes en supposant les degrés d'appartenance fixes :

$$\frac{\partial L}{\partial c_i} = 0 \implies c_i = \frac{\sum_{j=1}^N (u_{ij})^m x_j}{\sum_{j=1}^N (u_{ij})^m} \quad (\text{II.5})$$

La procédure générale de l'algorithme des c-moyennes floues est présentée dans l'Algorithme II.1.

Comme le problème de segmentation d'image est par définition un problème de partitionnement (groupement) des pixels (données), l'algorithme des c-moyennes floues permet de résoudre ce problème en considérant que :

$x_j$  : représente le niveau de gris du  $j^{\text{ème}}$  pixel de l'image,

$N$  : représente le nombre de pixels de l'image,

$C$  : représente le nombre de régions auxquelles on souhaite segmenter l'image,

$d()$  : représente la distance Euclidienne.

**Algorithme II.1** Méthode des c-moyennes floues

---

**Nécessite :**

Le nombre de classes  $C$

Le degré de flou  $m > 1$

L'erreur  $\epsilon$

1: Initialiser aléatoirement les centres des classes  $c_i^{(0)}$ .

2:  $k \leftarrow 1$

3: **Répéter**

4: Calculer les degrés d'appartenance  $u_{ij}^{(k)}$  en utilisant les centres  $c_i^{(k-1)}$  :

$$u_{ij}^{(k)} \leftarrow \frac{1}{\sum_{l=1}^c \left( \frac{d(x_j, c_l^{(k-1)})}{d(x_j, c_i^{(k-1)})} \right)^{\frac{2}{m-1}}}$$

5: Mettre à jour les centres  $c_i^{(k)}$  en utilisant  $u_{ij}^{(k)}$  :  $c_i^{(k)} \leftarrow \frac{\sum_{j=1}^N (u_{ij}^{(k)})^m x_j}{\sum_{j=1}^N (u_{ij}^{(k)})^m}$

6:  $k \leftarrow k + 1$

7: **Tant que**  $\|c_i^{(k)} - c_i^{(k-1)}\| < \epsilon$

8: **Retourner** : les centres des classes  $c_i$  et les degrés d'appartenance  $u_{ij}$ .

---

## II.3 Contribution à l'amélioration de l'algorithme des c-moyennes floues

La méthode de segmentation proposée se déroule en trois étapes :

1. Initialisation de l'algorithme des c-moyennes floues en utilisant l'optimisation par essaim particulière (OEP),
2. Segmentation de l'image (regroupement des pixels ) en utilisant le critère des c-moyennes floues amélioré,
3. Post-segmentation (détection et reclassification des pixels potentiellement mal classés).

Ces trois étapes sont détaillées dans les sections suivantes.

### II.3.1 Représentation et initialisation

Les performances de l'algorithme des c-moyennes floues dépendent énormément de la configuration initiale des centres des classes. Afin de pallier cet inconvénient, une des solutions, souvent utilisée, mais pas nécessairement la meilleure, consiste à exécuter l'algorithme en utilisant différentes configurations initiales dans l'espoir de trouver la meilleure solution globale. Toutefois, cette solution ne présente aucune garantie de convergence de l'algorithme vers l'optimum global, surtout dans le cas où le nombre de classes est important. Une autre solution,

---

### II.3 Contribution à l'amélioration de l'algorithme des c-moyennes floues

---

que nous jugeons plus efficace, consiste à exécuter l'algorithme une seule fois, en utilisant une métaheuristique d'optimisation, afin d'échapper aux optimums locaux et de se rapprocher au mieux de l'optimum global.

L'avantage des métaheurstiques réside dans leur robustesse pour résoudre les problèmes d'optimisation difficiles aux données incertaines, incomplètes ou trop bruitées. La solution fournie par les métaheurstiques d'optimisation est généralement une solution sous-optimale, souvent proche de l'optimale. Comme nous l'avons montré dans le chapitre 2, il existe plusieurs métaheurstiques d'optimisation dans la littérature. Pour notre problème nous avons choisi l'optimisation par essaim particulière (OEP). En plus des performances et de la reproductibilité de ses résultats dans plusieurs problèmes, cette métaheuristique est principalement conçue pour résoudre les problèmes d'optimisation à variables continues. Ce qui est le cas pour les méthodes de groupement ou de classification de données basées sur les centres mobiles où chaque centre représente une variable continue dans l'espace des attributs.

Dans le but d'optimiser l'étape d'initialisation de l'algorithme des c-moyennes floues, on utilise l'OEP afin de trouver la meilleure configuration initiale des positions des centres de classes. Ces centres de classes sont considérés comme une solution proche de la solution optimale et seront figolés par l'algorithme des c-moyennes floues afin d'approcher au mieux l'optimum global. La méthode OEP peut être considérée comme une procédure de recherche globale et la méthode FCM comme une procédure de recherche locale. Cette complémentarité est profitable et garantit des résultats plus performants.

Les équations de l'évolution de l'essaim de particules de l'algorithme OEP sont décrites ici (pour plus de détails se reporter au chapitre 2) :

$$V^{(k+1)} = w.V^{(k)} + c_1.rand_1.(Pbest^{(k)} - X^{(k)}) + c_2.rand_2.(Gbest^{(k)} - X^{(k)}) \quad (II.6)$$

$$X^{(k+1)} = X^{(k)} + V^{(k+1)} \quad (II.7)$$

où :

$X$  : est la position des particules,

$V$  : est la vitesse des particules,

$w$  : est le paramètre d'inertie,

$Pbest$  : est la meilleure position personnelle,  
 $Gbest$  : est la meilleure solution de l'essaim,  
 $rand_1, rand_2$  : sont des variables aléatoires entre 0 et 1,  
 $c_1, c_2$  : sont des constantes positives,  
 $k$  : est la variable d'itération.

Dans notre problème, les particules correspondent à des configurations constituées de nombres réels représentant les centres des différentes classes. Pour un nombre de classes  $C$ , chaque particule serait donc un vecteur de taille  $C$  où chaque composante représente un centre de classe. Une fois le nombre de classes fixé, les positions et les vitesses des particules sont initialisées aléatoirement. Les centres des classes sont initialisés dans la plage des niveaux de gris de l'image.

La condition d'arrêt de notre algorithme repose sur deux événements :

1. La fonction objectif n'évolue significativement plus après un certain nombre d'itérations

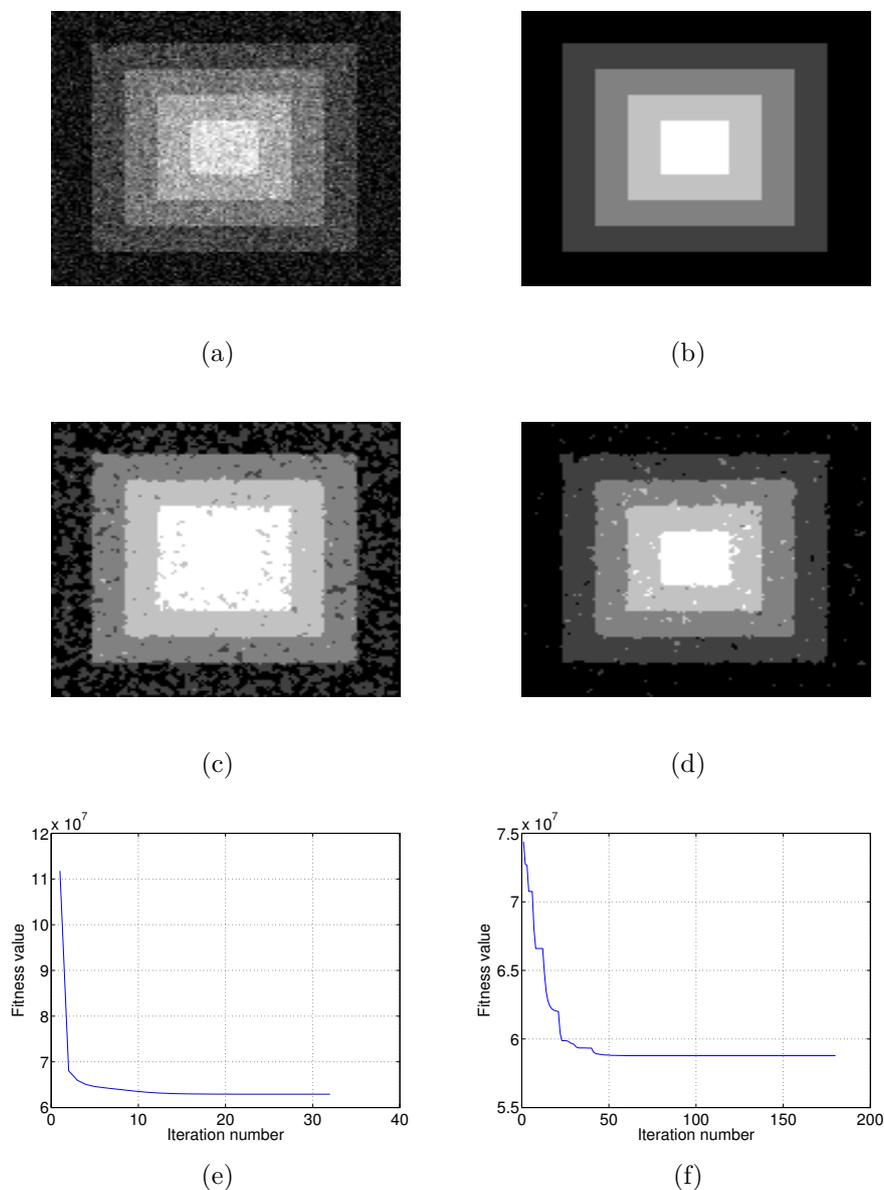
$nb_{stag}$  :

$$\forall k \in [0, nb_{MaxIter}], \forall n \in [k, k + nb_{stag}] : |J^{(n+1)} - J^{(n)}| < \epsilon \quad (\text{II.8})$$

2. Le nombre maximal d'itérations  $nb_{MaxIter}$  est consommé.

La figure II.1 montre un exemple de segmentation avec et sans utilisation de l'initialisation avec la méthode d'optimisation par essaim particulaire. L'image utilisée dans cet exemple est une image synthétique de taille  $128 \times 128$  contenant 5 classes de niveaux de gris 20, 70, 120, 170 et 220, affectée par un bruit Gaussien de variance 0,02 (après normalisation de l'image entre 0 et 1). La configuration initiale des centres des classes utilisée est définie aléatoirement (17, 45, 83, 135, 187). La configuration finale des centres des classes trouvée par la méthode FCM classique est (13,59, 31,99, 72,15, 122,83, 185,83). Ces valeurs correspondent à un optimum local du critère de segmentation. En revanche, la configuration finale des centres des classes dans le cas d'utilisation de l'OEP est (19,55, 68,19, 118,12, 166,59, 215,95), ce qui correspond à une solution beaucoup plus proche de l'optimum global. Cet exemple montre que l'utilisation de la métaheuristique d'optimisation OEP permet de remédier au problème de blocage de la méthode FCM dans les minima locaux et améliore de fait le résultat de segmentation.

Le pseudo-code de l'algorithme d'initialisation est donné dans l'Algorithme II.2.



**Figure II.1** – Exemple de résultat de segmentation avec et sans utilisation de l'initialisation avec la métaheuristique OEP (la post-segmentation n'est pas utilisée dans cet exemple). (a) Image à segmenter. (b) Image vérité terrain. (c) Résultat de segmentation sans utilisation de la métaheuristique OEP. (d) Résultat de segmentation avec utilisation de la métaheuristique OEP. (e) Evolution de la fonction objectif sans utilisation de la métaheuristique OEP. (f) Evolution de la fonction objectif avec utilisation de la métaheuristique OEP.

### II.3.2 Processus de segmentation

Dans les méthodes de segmentation basées sur l'optimisation d'un critère, le choix de ce dernier est une étape cruciale. Le critère classique de la méthode des c-moyennes floues présente

---

### Algorithme II.2 Initialisation de la méthode FCM avec l'OEP

---

#### Nécessite :

Le nombre de classes  $C$

Le degré de flou  $m > 1$

Le nombre de particules

Les paramètres  $c_1$  et  $c_2$

Le paramètre d'inertie initial  $w^{(0)}$

Le nombre maximal d'itérations  $nb_{MaxIter}$  et le nombre  $nb_{stag}$  d'itérations autorisé durant la phase de stagnation

L'erreur  $\epsilon$

- 1: Calculer l'image médiane  $\bar{X}$  de l'image  $X$
  - 2: Initialiser aléatoirement les centres des classes  $c_{ij}^{(0)}$  dans la plage des niveaux de gris de l'image  $X$ .
  - 3: Initialiser aléatoirement les vitesses des particules.
  - 4: Calculer les degrés d'appartenance ( $u_{ij}^{(0)}$ ) à l'aide de l'équation (II.12) en utilisant la distance Euclidienne.
  - 5: **Pour**  $k = 1$  **Jusqu'à**  $nb_{maxiter}$  **Faire**
  - 6:     **Pour** chaque particule **Faire**
  - 7:         Calculer la distance de Mahalanobis en utilisant  $u_{ij}^{(k-1)}$  dans les équations (III.6), (II.14) et (II.15)
  - 8:         Calculer  $u_{ij}^{(k)}$  avec l'équation (II.12)
  - 9:         Calculer la fonction *fitness* à l'aide de l'équation (II.10) en utilisant la distance de Mahalanobis
  - 10:        Mettre à jour la meilleure solution personnelle  $Pbest$
  - 11:     **Fin Pour**
  - 12:     Mettre à jour la meilleure solution globale  $Gbest$
  - 13:     **Pour** chaque particule **Faire**
  - 14:         Calculer la vitesse et la position de la particule à l'aide des équations (II.6) et (II.7)
  - 15:     **Fin Pour**
  - 16:     **Si** condition (II.8) **Alors**
  - 17:         Aller à 20
  - 18:     **Fin Si**
  - 19: **Fin Pour**
  - 20: **Retourner** :  $c_i$  les centres des classes (positions des particules) et les degrés d'appartenance  $u_{ij}$
- 

deux inconvénients majeurs :

1. *L'absence d'information spatiale* : seul le niveau de gris du pixel est pris en considération pour sa classification. Ce qui rend la méthode FCM très sensible vis-à-vis des bruits et artéfacts,
  2. *La distance Euclidienne* : l'utilisation de la distance Euclidienne est basée sur l'hypothèse
-

---

### II.3 Contribution à l'amélioration de l'algorithme des c-moyennes floues

---

que toutes les classes ont des formes super-sphériques à diamètres égaux. Or, cette hypothèse n'est pas vraie, surtout en problème de segmentation d'image, et induit en erreur la classification des pixels situés au niveau des frontières entre les régions.

Afin de remédier à ces inconvénients, nous proposons de modifier le critère de segmentation classique afin de prendre en compte l'information spatiale et en utilisant une métrique de distance qui s'affranchit de la variabilité de la géométrie des classes. Les modifications que nous apportons sont basées sur les travaux proposés par [Ahmed et al., 2002]. En effet, Ahmed et al [Ahmed et al., 2002] ont proposé un nouveau critère (Eq. (II.9)) en ajoutant un nouveau terme, qui contraint chaque pixel à appartenir à la même classe que ses voisins :

$$J = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(x_j, c_i) + \frac{\alpha}{N_R} \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m \sum_{r \in N_i} d^2(x_r, c_i) \quad (\text{II.9})$$

en désignant par :

$N_i$  : l'ensemble des pixels dans le voisinage du  $i^{\text{ème}}$  pixel,

$N_R$  : le nombre de pixels de l'ensemble  $N_i$ ,

$\alpha$  : le paramètre qui contrôle l'influence du terme du voisinage.

On remarque que le deuxième terme de ce critère est le même que le premier dans lequel la distance entre un pixel et un centre de classe est remplacée par la moyenne des distances des pixels voisins d'un pixel au centre d'une classe. Ce qui contraint le pixel à appartenir à la même classe que ses voisins. Ce critère nécessite de calculer pour chaque pixel l'ensemble des distances entre ses pixels voisins et les centres des classes, ce qui augmente le temps de calcul. Afin de le réduire, Chen and Zhang [Chen & Zhang, 2004] ont proposé de remplacer la moyenne des distances des pixels voisins au centre de classe par la distance entre la moyenne des pixels voisins au centre de classe. Cela revient à utiliser l'image moyenne déterminée au préalable. Le nouveau critère est le suivant (Eq. (II.10)) :

$$J = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(x_j, c_i) + \alpha \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(\bar{x}_j, c_i) \quad (\text{II.10})$$

où :

$\bar{x}_i$  : est le  $i^{\text{ème}}$  pixel de l'image moyenne.

La mise à jour des centres des classes et des degrés d'appartenance est obtenue de la même manière que dans le critère classique, en utilisant les multiplicateurs de Lagrange (Eqs. (II.11) et (II.12)) :

$$c_i = \frac{\sum_{j=1}^N u_{ij}^m (x_j + \alpha \bar{x}_j)}{(1 + \alpha) \sum_{j=1}^N (u_{ij})^m} \quad (\text{II.11})$$

$$u_{ij} = \frac{(d^2(x_j, c_i) + \alpha d^2(\bar{x}_j, c_i))^{\frac{1}{m-1}}}{\sum_{l=1}^C (d^2(x_j, c_l) + \alpha d^2(\bar{x}_j, c_l))^{\frac{1}{m-1}}} \quad (\text{II.12})$$

La deuxième amélioration apportée au critère de segmentation concerne les inconvénients dus à l'utilisation de la distance Euclidienne. Afin d'y remédier, nous proposons d'utiliser la distance de Mahalanobis qui prend mieux en considération la forme géométrique des classes. La distance de Mahalanobis est définie par l'équation suivante (Eq. (III.6)) :

$$d^2(x_j, c_i) = (x_j - c_i)^T V_i (x_j - c_i) \quad (\text{II.13})$$

où :

$V_i$  : est la matrice de normalisation induite par la classe  $i$ .

La matrice  $V_i$  est utilisée comme variable d'optimisation dans le critère de segmentation, afin que la normalisation de distance s'adapte à la structure topologique locale des données. La matrice  $V_i$  est calculée à l'aide des équations suivantes (Eqs. (II.14) et (II.15)) :

$$V_i = \left| \sum_i \right|^{\frac{1}{p}} \sum_i^{-1} \quad (\text{II.14})$$

$$\sum_i = \frac{\sum_{j=1}^N \sum_{i=1}^C u_{ij}^m (x_j - c_i) (x_j - c_i)^T}{\sum_{j=1}^N \sum_{i=1}^C u_{ij}^m} \quad (\text{II.15})$$

où  $\sum_i$  représente la matrice de covariance floue,  $|\cdot|$  est l'opérateur de calcul du déterminant des matrices et  $p$  représente la dimension du problème.

L'algorithme de segmentation est présenté dans l'Algorithme [II.3](#).

### II.3.3 Post-segmentation

En règle générale, les erreurs de classification augmentent avec l'importance du bruit présent dans les images. Ces erreurs se manifestent globalement sous deux formes :

1. La migration de pixels d'une région à l'autre dans le voisinage de leur frontière, ce qui conduit à une déformation des contours des objets détectés,
2. L'apparition d'artéfacts sous forme de grumeaux de pixels isolés dans des régions homo-

## II.3 Contribution à l'amélioration de l'algorithme des c-moyennes floues

---

---

### Algorithme II.3 Classification des pixels

---

#### Nécessite :

Les positions des centres des classes  $c_i^{(0)}$  et les degrés d'appartenance  $u_{ij}^{(0)}$  retournés par l'algorithme II.2

Le nombre maximal d'itérations  $nb_{MaxIter}$

L'erreur  $\epsilon$

1: **Pour**  $k = 1$  **Jusqu'à**  $nb_{MaxIter}$  **Faire**

2: Calculer la distance de Mahalanobis en utilisant  $u_{ij}^{(k-1)}$  et les équations (III.6), (II.14) et (II.15)

3: Calculer les degrés d'appartenance  $u_{ij}^{(k)}$  avec l'équation (II.12)

4: Mettre à jour les centres des classes  $c_i$  en utilisant  $u_{ij}^{(k)}$  et l'équation (II.11)

5: **Si**  $\|U^{k+1} - U^{(k)}\| \prec \epsilon$  **Alors**

6: Aller à 9

7: **Fin Si**

8: **Fin Pour**

9: **Retourner** :  $u_{ij}$  le degrés d'appartenance de chaque pixel de l'image aux différentes classes

---

gènes de même classe.

Ainsi, dans la méthode de post-segmentation que nous proposons, nous nous intéressons uniquement aux régions de l'image qui contiennent ces erreurs de classification potentielles. L'idée principale est de détecter et reclasser les pixels potentiellement mal classés en se basant sur les pixels de confiance (pixels supposés bien classés).

La méthode de correction des erreurs de classification que nous proposons se déroule en deux étapes. La première étape est la détection des pixels potentiellement mal classés. Nous définissons un pixel potentiellement mal classé comme un pixel n'appartenant pas à la même classe que de tous ses voisins immédiats ( $N_{3 \times 3}$ ). La deuxième étape est la reclassification de ces pixels. Pour ce faire, nous définissons un critère d'homogénéité locale à minimiser (Eq. (II.16)). Ce critère tient compte du niveau de gris du pixel à reclasser, des moyennes et des variances des différentes classes dans un voisinage plus élargi ( $N_{5 \times 5}$ ), ainsi que du nombre des pixels de chaque classe dans ce voisinage. A noter que chaque pixel potentiellement mal classé ne peut être réaffecté qu'aux classes de ses voisins immédiats ( $N_{3 \times 3}$ ).

Le critère de reclassification que nous définissons est le suivant (Eq. (II.16)) :

$$J_i^{(j)} = \frac{\alpha \sigma_j^{(i)} + \|x_i - \mu_j^{(i)}\|}{\beta N_j} + \sum_{k=1, k \neq j}^{N_{label}} \sigma_k^{(i)} \quad (\text{II.16})$$

où :

$x_i$  : est le  $i^{\text{ème}}$  pixel potentiellement mal classé à reclasser,

$N_{label}$  : est le nombre de classes présentes dans le voisinage  $N_{3 \times 3}$  du  $i^{\text{ème}}$  pixel détecté,

$N_j$  : est le nombre de pixels appartenant à la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\mu_j^{(i)}$  : est la moyenne locale de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\sigma_j^{(i)}$  : est la variance locale de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel après son affectation à cette classe,

$\sigma_k^{(i)}$  : est la variance locale de la classe  $k$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\alpha$  : est un paramètre qui détermine l'impact de la variance locale dans la réaffectation du pixel détecté,

$\beta$  : est un paramètre qui détermine l'impact du nombre de pixels appartenant à la classe  $j$ .

Le premier terme de ce critère représente l'homogénéité locale du niveau de gris de la nouvelle classe du pixel reclassé et le deuxième terme représente l'homogénéité locale des autres classes. Ainsi, ce critère atteint son minimum quand le pixel est affecté à la classe qui assure la meilleure compacité locale de toutes les classes. Dans ce critère nous, nous sommes basés uniquement sur l'information locale pour deux raisons. La première est qu'un pixel présente souvent des caractéristiques plus similaires à celles de ses voisins que celles des pixels éloignés (surtout en présence de bruits de faible fréquence). Il est donc naturel de tirer profit des informations de ses pixels voisins de confiance afin de guider sa réaffectation. La deuxième raison est que l'information globale est déjà prise en compte lors du processus de segmentation et qu'en raison de la variabilité des caractéristiques des pixels éloignés, leur utilisation risque de reproduire les mêmes erreurs de segmentation.

Les figures II.2 et II.3 montrent deux exemples de résultat de segmentation avec et sans application de la méthode de post-segmentation proposée. Nous remarquons dans la figure II.2 que les résultats de segmentation sont améliorés après application de la post-segmentation. En effet, pour ces exemples, tous les pixels et artéfacts isolés sont corrigés et la qualité des contours est améliorée. On remarque dans la figure II.3 que l'ensemble des pixels réellement mal classés (figure II.3(c)) appartiennent à l'ensemble des pixels détectés comme potentiellement

mal classés (figure II.3(b)). Ainsi, certains pixels détectés sont bien classés et réaffectés à leurs classes d'origine et la majorité des autres sont réaffectés à leurs classes appropriées. Le pseudo-code de la méthode de post-segmentation proposée est donné dans l'Algorithme II.4.

---

### Algorithme II.4 Post-segmentation

---

**Étape 1 :** Détection des pixels potentiellement mal classés ( $x_l$ )

```

1:  $l \leftarrow 1$ 
2: Pour tous les pixels  $x_i$  de l'image Faire
3:   Si ( $label(x_i) \neq label(x_j \in N_{vois})$ ) Alors
4:      $x_l \leftarrow x_i$ 
5:     Pour toute classe  $m$  dans dans le voisinage  $N_{Lv}$  du pixel  $x_l$  Faire
6:       Calculer la variance locale  $\sigma_m^{(l)}$ 
7:       Calculer la moyenne locale  $\mu_m^{(l)}$ 
8:     Fin Pour
9:      $l \leftarrow l + 1$ 
10:  Fin Si
11: Fin Pour
12: Retourner :  $x_l$ ,  $\sigma_m^{(l)}$  et  $\mu_m^{(l)}$ 

```

**Étape 2 :** Reclassification des pixels détectés

```

1: Pour tout pixel  $x_i$  détecté Faire
2:   Pour  $j = 1$  to  $N_{label}$  Faire
3:     Calculer la fonction objectif  $J_i^{(j)}$  en utilisant l'équation (II.16)
4:   Fin Pour
5:   Trouver  $j = argmin(J_i^{(j)})$ 
6:   Affecter le pixel  $x_i$  à la classe  $j$ 
7: Fin Pour

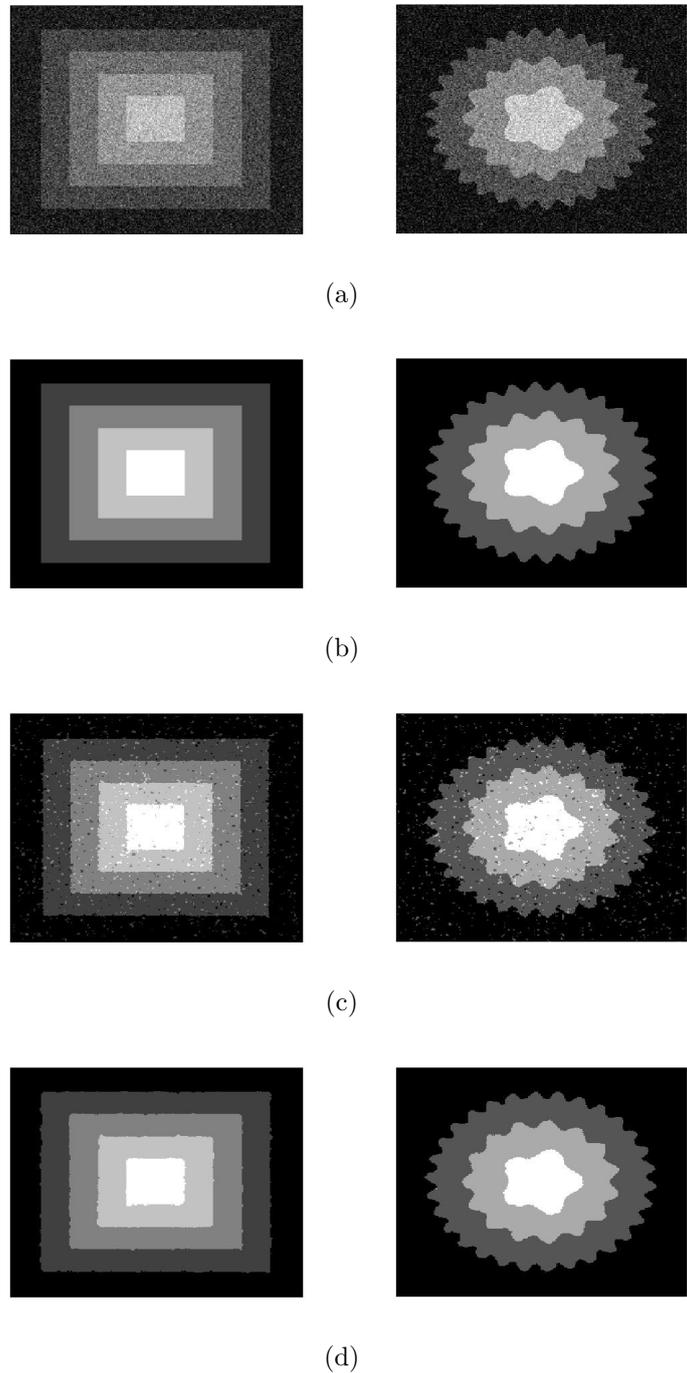
```

---

## II.4 Résultats expérimentaux

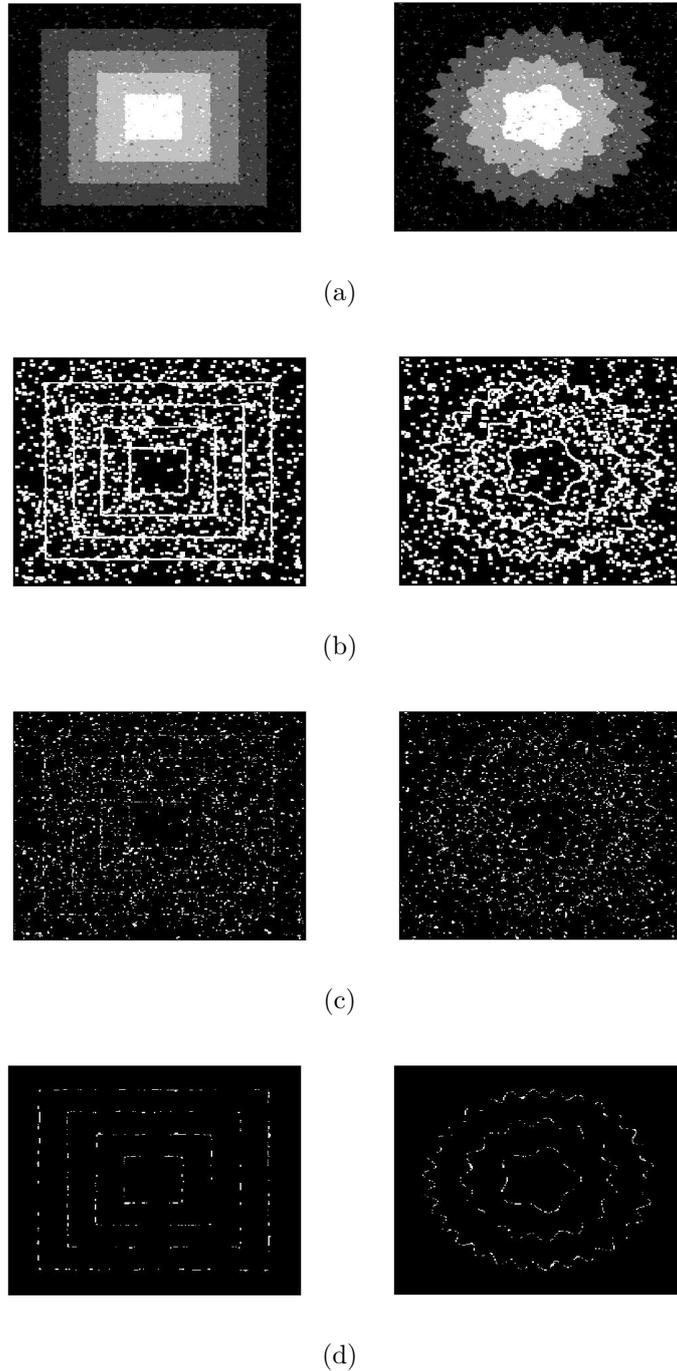
Dans cette section, nous présentons les résultats expérimentaux de la méthode proposée, ainsi qu'une comparaison avec des méthodes de segmentation basées sur la classification par logique floue : FCM, FCM-S1, FCM-S2, EnFCM, FGFCM et FLICM. Deux types d'images ont été utilisés :

1. Des images synthétiques qui contiennent différents nombres de classes auxquelles on a ajouté différents types et taux de bruits,
2. Des images IRM du cerveau simulées issues de la base BrainWeb [Bra, 2013].



**Figure II.2** – Exemple de segmentation avant et après post-segmentation. (a) Images à segmenter. (b) Images vérité terrain. (c) Résultats de segmentation avant post-segmentation. (d) Résultats de segmentation après post-segmentation.

Afin de mesurer et comparer objectivement les performances de la méthode proposée, la mesure de cohérence de segmentation est utilisée. Cette mesure est définie comme la somme



**Figure II.3** – Détection et réaffectation des pixels potentiellement mal classés. (a) Résultats de segmentation avant post-segmentation. (b) Pixels potentiellement mal classés détectés. (c) Pixels mal classés avant post-segmentation. (d) Pixels mal classés après post-segmentation.

des pixels bien classés divisée par le nombre total des pixels de l'image (Eq. (II.17)), ce qui

représente le taux de pixels bien classés :

$$SA = \sum_{i=1}^c \frac{\text{card}(A_i \cap C_i)}{\sum_{j=1}^c \text{card}(C_j)} \quad (\text{II.17})$$

où :

$c$  : est le nombre de classes (régions),

$A_i$  : est l'ensemble des pixels appartenant à la classe  $j$  dans l'image résultat de segmentation.

$C_i$  : est l'ensemble des pixels appartenant à la classe  $i$  dans l'image vérité terrain,

$\text{card}()$  : est l'opérateur cardinalité.

Comme les résultats des méthodes concurrentes dépendent des solutions initiales et comme la méthode proposée est basée sur une technique d'optimisation stochastique, une seule exécution ne permet pas d'analyser objectivement leurs performances. Ainsi, chaque algorithme est exécuté 10 fois en utilisant à chaque fois une configuration initiale aléatoire. La moyenne de ces 10 exécutions est ensuite retenue.

### II.4.1 Résultats sur des images synthétiques

Afin d'évaluer et de comparer les performances des différentes méthodes, elles sont testées en premier lieu sur une base d'images synthétiques de taille  $256 \times 256$ . Cette base est constituée d'images contenant 2, 3, 4 ou 5 classes affectées de bruits de type Gaussien, uniforme et sel & poivre. Les taux de ces bruits sont 10%, 15%, 20% et 25% pour les images contenant 2 ou 3 classes, et 3%, 5%, 8% et 10% pour celles contenant 4 ou 5 classes. Les figures II.4 et II.5 montrent des exemples de résultats de segmentation sur des images synthétiques contenant 5 classes affectées de 3% de bruit Gaussien, 3% de bruit uniforme et 10% de bruit sel & poivre (figure II.4(b)).

Visuellement, on remarque que les méthodes FCM-S1 (figure II.4(d)) et FLICM (figure II.5(c)) donnent de bons résultats pour les images affectées de bruits Gaussien et uniforme, mais les résultats sur l'image contenant un bruit sel & poivre sont mauvais. La méthode FCM-S2 (figure II.4(e)) arrive à traiter le bruit sel & poivre, mais le résultat n'est pas satisfaisant. D'autre part, la méthode proposée donne de bons résultats sur les trois types de bruits et ces

résultats sont assez satisfaisants. Cela est vérifié quantitativement par la cohérence de segmentation  $SA$  présentée dans les tableaux II.1 et II.2. Ces tableaux donnent le  $SA$  moyen des différentes méthodes en fonction des types de bruits et du nombre de classes, respectivement. On remarque que la méthode proposée est plus robuste que les méthodes de comparaison vis-à-vis des différents types et taux de bruits ainsi que le nombre de classes. Dans la figure II.6 sont présentés les  $SA$  moyens des différentes méthodes sur toute la base d'images synthétiques. De cette figure, on conclut que la méthode proposée (IFCMS) donne globalement de meilleurs résultats que les méthodes concurrentes.

|              | FCM   | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS        |
|--------------|-------|--------|--------|-------|-------|-------|--------------|
| Gaussien     | 47,48 | 85,16  | 78,68  | 78,07 | 88,92 | 91,99 | <b>93,11</b> |
| Uniforme     | 42,40 | 83,06  | 74,30  | 76,31 | 76,31 | 94,49 | <b>95,17</b> |
| Sel & poivre | 75,81 | 81,02  | 90,05  | 79,69 | 72,50 | 58,39 | <b>91,69</b> |

**Tableau II.1** – Cohérence de segmentation moyenne ( $SA\%$ ) des différentes méthodes en fonction des types de bruits pour la base d'images synthétiques.

|           | FCM   | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS        |
|-----------|-------|--------|--------|-------|-------|-------|--------------|
| 2 classes | 90,66 | 96,91  | 99,90  | 94,98 | 94,99 | 96,27 | <b>99,83</b> |
| 3 classes | 65,66 | 80,32  | 77,80  | 83,98 | 84,44 | 82,48 | <b>95,64</b> |
| 4 classes | 38,18 | 79,55  | 75,68  | 78,68 | 77,81 | 75,21 | <b>93,13</b> |
| 5 classes | 26,42 | 75,53  | 70,64  | 54,47 | 66,73 | 72,54 | <b>84,68</b> |

**Tableau II.2** – Cohérence de segmentation moyenne ( $SA\%$ ) des différentes méthodes en fonction des nombres de classes pour la base d'images synthétiques.

### II.4.2 Résultats sur des images IRM simulées

La segmentation vérité terrain des images IRM est souvent indisponible, ce qui rend l'évaluation quantitative des performances sur ce type d'image impossible. Une solution possible consiste à utiliser des images IRM simulées. Dans notre étude, nous utilisons la base de données BranWeb qui contient des images IRM du cerveau simulées. Les images de cette base sont produites par un simulateur IRM qui fournit des volumes réalistes. La disponibilité des images de segmentation vérité terrain nous permet d'évaluer quantitativement les performances des différentes méthodes de segmentation. La base que nous utilisons se compose de la 91<sup>ème</sup> coupe du plan axial générée avec la modalité T1 à laquelle on a ajouté un bruit de taux 3%,

7% et 9%. Le paramètre d'intensité d'inhomogénéité est fixé à 0%, 20% et 40% pour chaque taux de bruit. Ces images sont segmentées en 4 classes : matière grise (MG), matière blanche (MB), liquide céphalo-rachidien (LCR) et fond. Les figures II.7 et II.8 montrent un exemple de segmentation pour une image contenant un bruit de taux égal à 7% et un paramètre d'intensité d'inhomogénéité fixé à 20%. On remarque que sur ce type d'image il est très difficile de comparer visuellement les performances des différentes méthodes. Le tableau II.3 représente la mesure de cohérence de segmentation ( $SA$ ) des différentes méthodes pour chaque région de l'image ainsi que la moyenne générale. On remarque sur ce tableau que la méthode proposée donne de meilleurs résultats de segmentation sur cette base d'images.

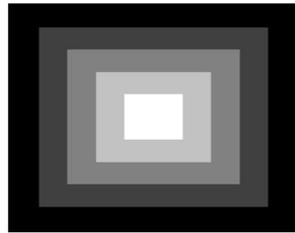
|         | FCM   | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS        |
|---------|-------|--------|--------|-------|-------|-------|--------------|
| LCR     | 93,93 | 93,17  | 93,25  | 92,44 | 91,64 | 90,52 | <b>94,50</b> |
| MG      | 89,66 | 90,61  | 92,71  | 85,14 | 85,27 | 92,55 | <b>93,81</b> |
| MB      | 88,81 | 92,46  | 92,75  | 90,92 | 90,97 | 91,77 | <b>93,10</b> |
| Moyenne | 90,80 | 92,08  | 92,90  | 89,50 | 89,29 | 91,61 | <b>93,80</b> |

**Tableau II.3** – Cohérence de segmentation moyenne ( $SA\%$ ) des différentes méthodes sur la base d'images IRM simulées pour chaque tissu ainsi que leur moyenne globale.

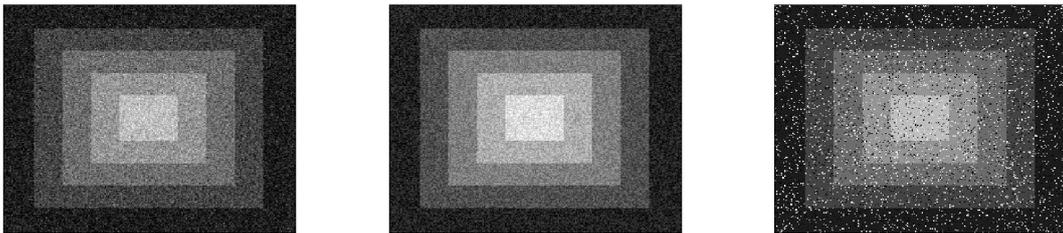
## II.5 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle méthode de segmentation d'image résultant de l'amélioration de la méthode de groupement de données c-moyennes floues. La première amélioration concerne l'étape d'initialisation, où la métaheuristique d'optimisation par essaim particulière est utilisée afin de pallier au problème de blocage dans les minima locaux. La deuxième amélioration concerne le critère de segmentation qui a été amélioré en introduisant l'information spatiale et la distance de Mahalanobis, afin que la segmentation soit plus robuste vis-à-vis du bruit et afin de prendre en considération la forme géométrique des différentes classes. Enfin, une étape de post-segmentation est utilisée afin de raffiner le résultat de segmentation, en détectant et reclassant les pixels potentiellement mal classés, grâce à un nouveau critère local d'affectation. La méthode proposée a été testée et comparée à 6 méthodes de la littérature basées sur la classification par logique floue. Deux types d'images ont été utilisées pour l'évaluation des performances : des images synthétiques contenant différents nombres de classes et taux de bruits, et des images IRM simulées contenant différents taux de bruit et

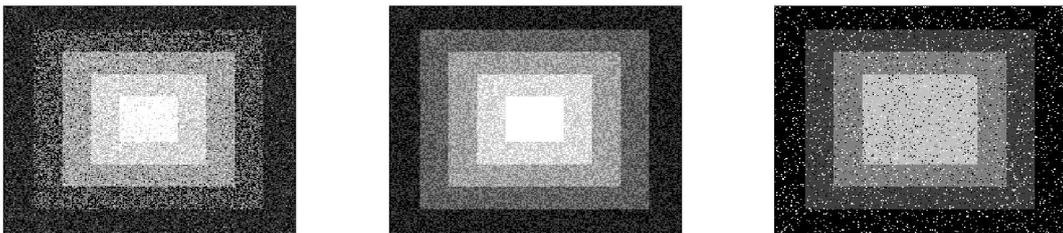
paramètres d'intensité d'inhomogénéité. Les résultats de segmentation ont été améliorés par la méthode proposée comparativement aux méthodes concurrentes. Cependant, en présence de bruits importants dans les images, on remarque qu'il subsiste encore des erreurs de segmentation au niveau des pixels situés au voisinage des frontières entre régions. Pour corriger ces erreurs, nous proposons d'ajouter un deuxième critère de segmentation basé contour. Les deux critères seront optimisés simultanément à l'aide d'une technique d'optimisation multiobjectif. Cette nouvelle méthode est présentée en détail dans le chapitre suivant.



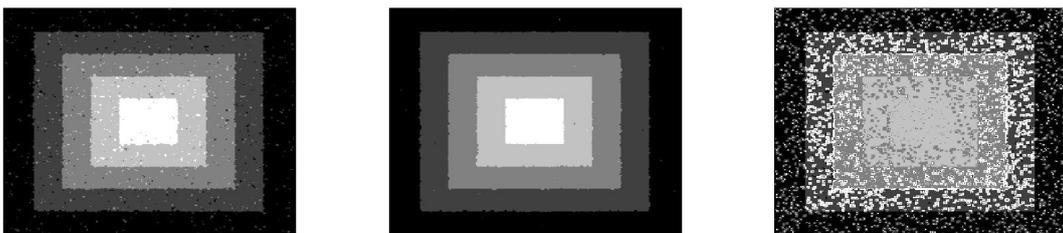
(a) Image originale



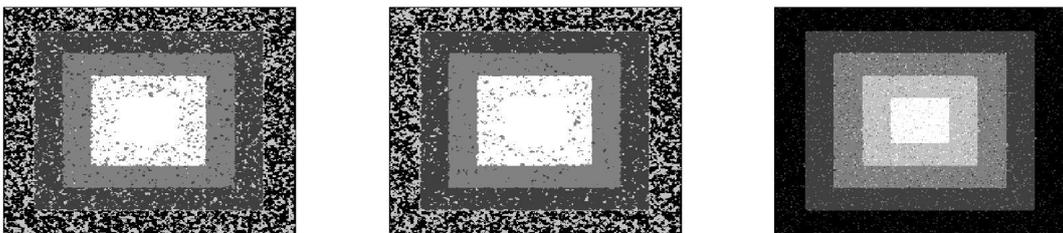
(b) Image bruitées (de gauche à droite : bruit Gaussien, uniforme et sel & poivre)



(c) Résultats de segmentation par la méthode FCM

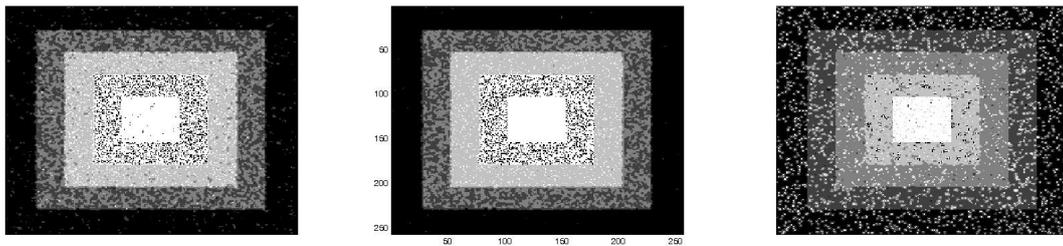


(d) Résultats de segmentation par la méthode FCM-S1

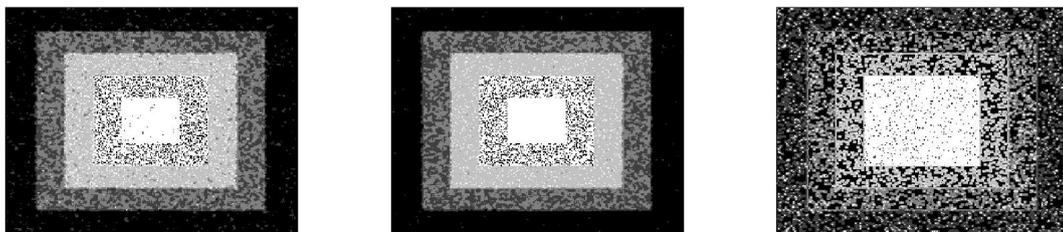


(e) Résultats de segmentation par la méthode FCM-S2

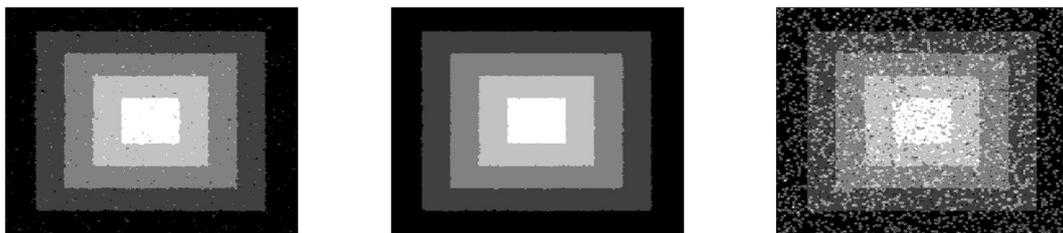
**Figure II.4** – Exemples de segmentations d'images synthétiques.



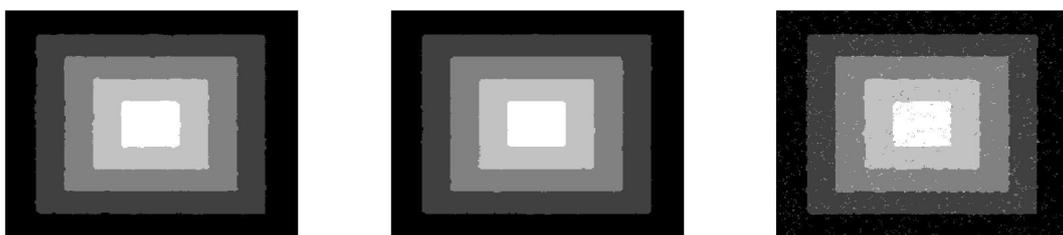
(a) Résultats de segmentation par la méthode EnFCM



(b) Résultats de segmentation par la méthode FGFCM

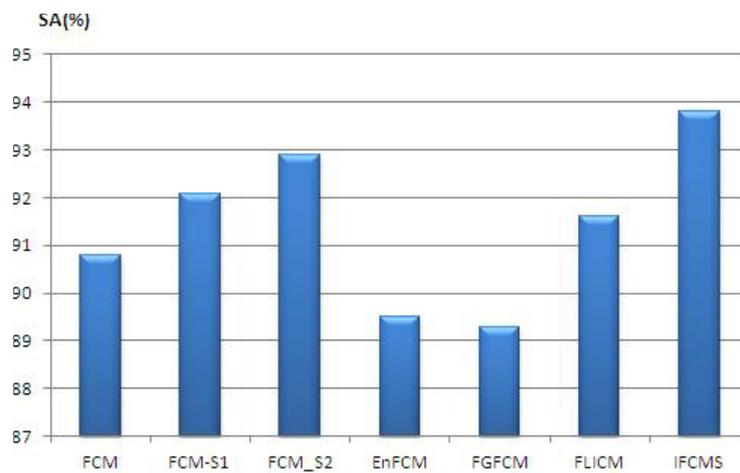


(c) Résultats de segmentation par la méthode FLICM

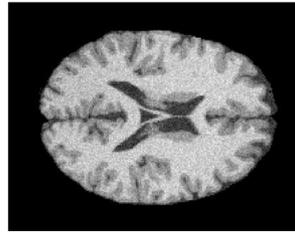


(d) Résultats de segmentation par la méthode IFCMS

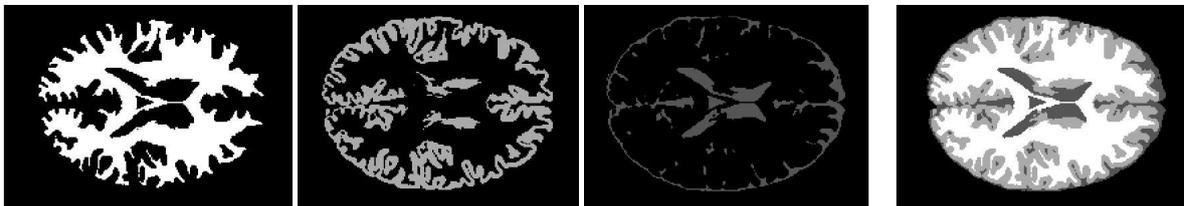
**Figure II.5** – Exemples de segmentations d'images synthétiques (suite des résultats).



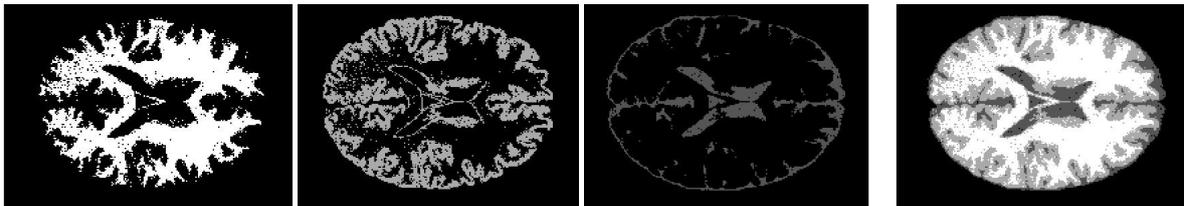
**Figure II.6** – Moyenne globale de la cohérence de segmentation ( $SA\%$ ) sur toute la base d'images synthétiques.



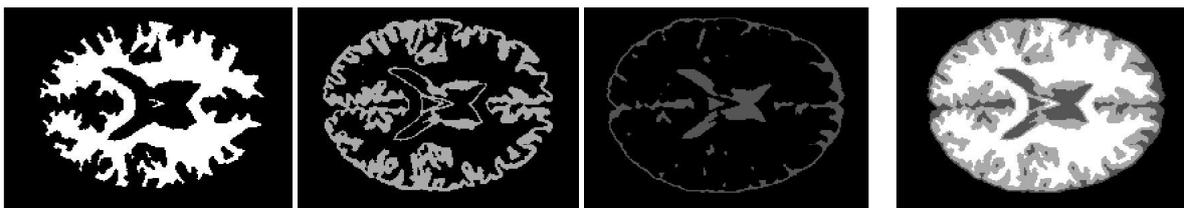
(a) Image à segmenter



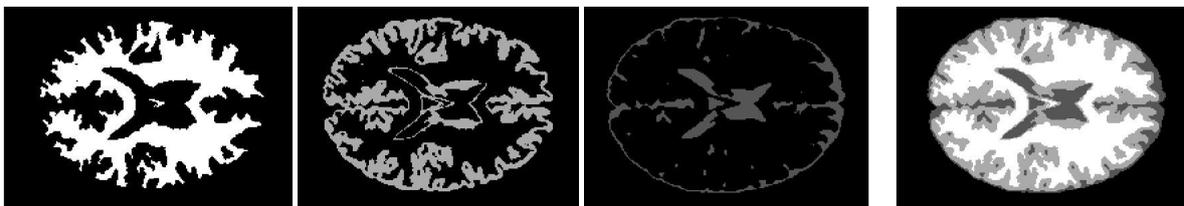
(b) Image vérité-terrain



(c) Résultats de segmentation par la méthode FCM

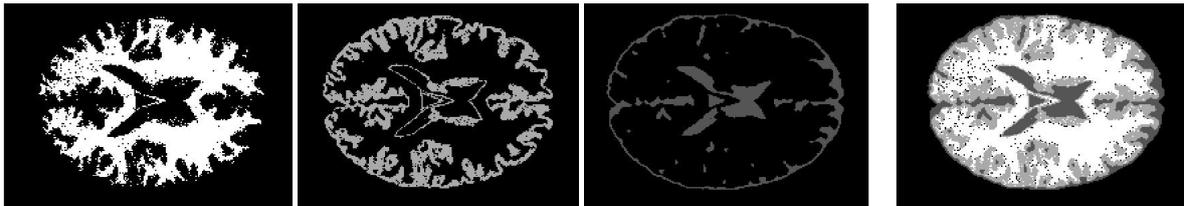


(d) Résultats de segmentation par la méthode FCM-S1

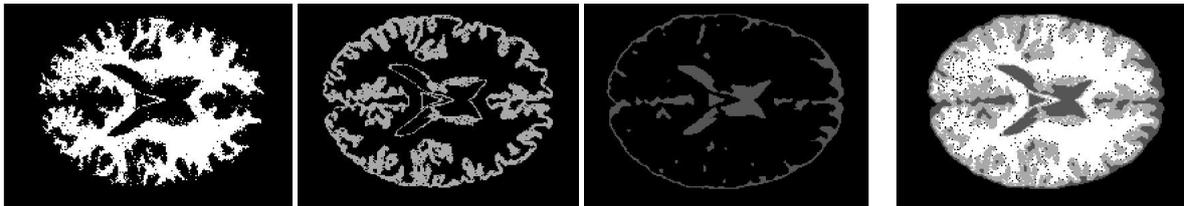


(e) Résultats de segmentation par la méthode FCM-S2

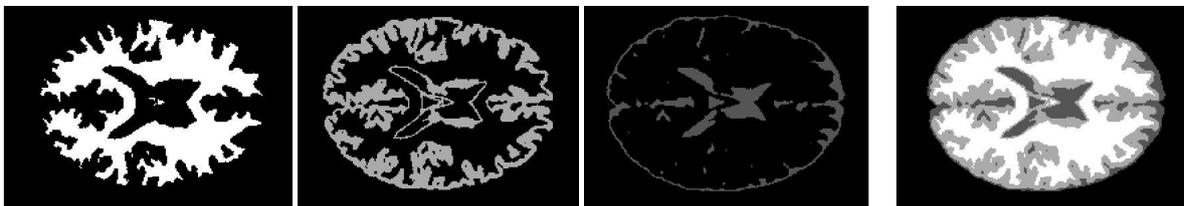
**Figure II.7** – Exemples de segmentations d'images IRM simulées de cerveau.



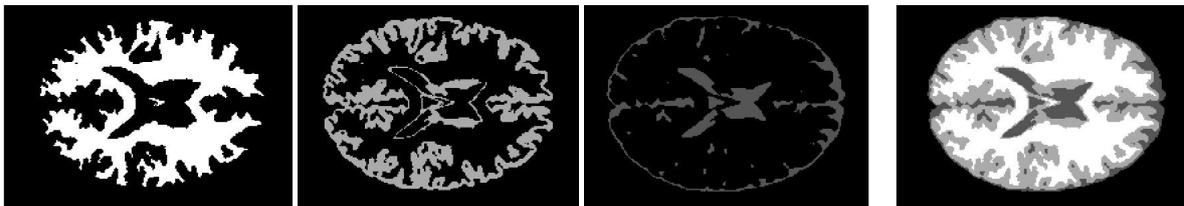
(a) Résultats de segmentation par la méthode EnFCM



(b) Résultats de segmentation par la méthode FGFCM



(c) Résultats de segmentation par la méthode FLICM



(d) Résultats de segmentation par la méthode IFCMS

**Figure II.8** – Exemples de segmentations d'images IRM simulées de cerveau (suite des résultats).

---

---

# Chapitre III

---

## Contribution à la segmentation d'image par classification floue optimisée par essaim particulaire multiobjectif

### III.1 Introduction

Dans le chapitre précédent, nous avons proposé une amélioration de la méthode de la segmentation d'image par logique floue (FCM). Néanmoins, malgré les bons résultats obtenus, nous avons remarqué que des erreurs de segmentation subsistent encore, notamment au niveau des contours. Dans le but d'améliorer encore ses performances, nous proposons dans ce chapitre d'ajouter un deuxième critère de segmentation basé contours. Les deux critères sont optimisés simultanément dans une approche d'optimisation multiobjectif via la métaheuristique OEP. De plus, nous assumons que la solution finale recherchée n'est pas forcément parmi les solutions du front de Pareto, mais répartie sur l'ensemble de ces solutions. Au lieu donc d'opter pour la solution la plus performante des solutions du front de Pareto, comme cela se fait couramment, nous proposons de combiner ces solutions, afin d'extraire la totalité des pixels bien classés répartis sur l'ensemble des ces solutions.

Ce chapitre est organisé comme suit : dans la section [III.2](#) nous présentons l'optimisation multiobjectif via la métaheuristique OEP, dans la section [III.3](#) on détaille la méthode de segmentation proposée, dans la section [III.4](#) sont présentés les résultats expérimentaux et finalement nous concluons ce chapitre dans la section [III.5](#).

## III.2 Optimisation multiobjectif par essaim particulaire (MOPSO)

Dans cette section, nous présentons les particularités de l'optimisation multiobjectif ainsi que la méthode d'optimisation par essaim particulaire (OEP) multiobjectif.

### III.2.1 L'optimisation multiobjectif

Un problème d'optimisation multiobjectif ou multicritère peut être défini comme un problème dont on recherche l'ensemble des variables de décision  $\bar{x}^* = [x_1^*, x_2^*, \dots, x_n^*]^T$  satisfaisant certaines contraintes et optimisant simultanément un ensemble de critères de dimension supérieure ou égale à deux  $\bar{f}(\bar{x}) = [f_1(\bar{x}), f_2(\bar{x}), \dots, f_r(\bar{x})]^T$ . Les contraintes définissent l'ensemble des solutions admissibles  $\mathbb{A}$ . Le vecteur  $\bar{x}^*$  représente une solution optimale dans  $\mathbb{A}$ . En optimisation multiobjectif simultanée de critères, il est possible qu'un conflit apparaisse. En effet, le plus souvent un critère ne peut être amélioré sans détériorer au moins un autre. De plus, il est rare de se trouver dans une situation où une seule solution  $\bar{x}^*$  représente la solution optimale pour toutes les fonctions objectifs.

Le concept d'optimalité de Pareto est fondamental dans le domaine de l'optimisation multiobjectif. La définition formelle de l'optimalité de Pareto, dans le cas d'un problème de minimisation, est énoncée comme suit : une solution  $\bar{x}^*$  est dite optimale au sens de Pareto si et seulement s'il n'existe aucune solution  $\bar{x}$  qui domine  $\bar{x}^*$ , i.e. il n'existe aucun  $\bar{x}$  tel que :

$$\forall i \in 1, 2, \dots, k, f_i(\bar{x}) \leq f_i(\bar{x}^*) \quad (\text{III.1})$$

et

$$\exists i \in 1, 2, \dots, k, f_i(\bar{x}) < f_i(\bar{x}^*) \quad (\text{III.2})$$

En d'autres termes, une solution est dite optimale au sens de Pareto s'il n'existe aucune autre solution qui améliore un critère sans en détériorer simultanément un autre. L'ensemble des solutions de Pareto (front de Pareto) est composé de solutions non dominées. Il existe dans la littérature plusieurs approches pour résoudre les problèmes d'optimisation multiobjectif [Deb, 2001; Coello, 1999]. Dans ce travail, nous utiliserons la métaheuristique d'optimisation par essaim particulaire (OEP) multiobjectif décrite dans la section suivante.

### III.2.2 Optimisation multiobjectif par essaim particulaire

Afin de résoudre les problèmes d'optimisation multiobjectif, la métaheuristique OEP originale a besoin d'être adaptée. Dans ce travail l'approche proposée par [Coello et al., 2004] est prise en considération. En termes d'optimisation multiobjectif, le but est de trouver l'ensemble des solutions du front de Pareto, ainsi la solution du problème ne se compose pas d'une seule mais de plusieurs solutions. Il est donc nécessaire de prendre en considération les points suivants :

1. Comment conserver l'ensemble des solutions trouvées au cours du processus de recherche en considérant toutes les solutions trouvées depuis le début ?
2. Comment choisir les particules leaders (meilleure solution personnelle et meilleure solution globale) ?
3. Comment maintenir la diversité de l'essaim ?

Habituellement, une archive externe est utilisée pour stocker l'ensemble des solutions non dominées trouvées au cours du processus de recherche (depuis le début de la recherche). Cette archive est mise à jour à chaque itération en comparant les solutions non dominées, trouvées dans l'itération en cours, à celles présentes dans l'archive, en utilisant les règles suivantes :

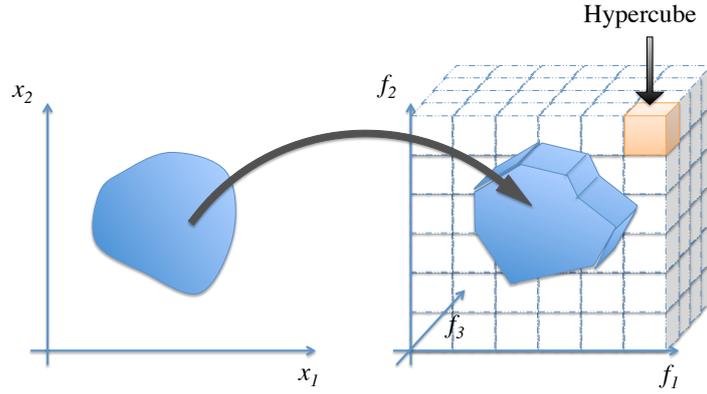
1. Si l'archive est vide (au début de la recherche), la nouvelle solution est stockée dans l'archive,
2. Si la nouvelle solution domine des solutions dans l'archive, ces dernières sont remplacées par la nouvelle,
3. Si la nouvelle solution est dominée par toutes les solutions présentes dans l'archive, l'archive reste telle quelle.

Vu que l'archive externe est mise à jour à chaque itération, cela risque d'augmenter considérablement le temps de calcul. Ainsi, la taille de cette archive est limitée. Une fois qu'elle est pleine, les particules situées dans les zones les moins denses sont prioritaires pour y rester afin de maintenir la diversité de l'essaim.

À noter que chaque particule contient une mémoire où est stockée la meilleure solution (au sens de Pareto) trouvée par cette particule (meilleure solution personnelle). Cette mémoire est aussi contenue dans l'archive externe.

La meilleure solution globale (leader) de l'essaim est choisie aléatoirement dans l'archive externe en favorisant les régions qui contiennent une forte densité de particules (régions pro-

metteuses). Les régions utilisées sont des hypercubes dans l'espace des critères (figure III.1). En premier lieu, une région (hypercube) est choisie en utilisant la roulette russe. Le critère utilisé pour la roulette russe est le nombre de particules (à maximiser). Puis une particule est choisie aléatoirement dans cette région si elle contient plus d'une particule.



**Figure III.1** – Espace de décision (gauche) et espace des critères (droite) divisé en régions (hypercubes).

Les équations de l'évolution de l'essaim sont les suivantes (Eqs. (III.3) et (III.4)) :

$$V^{(k+1)} = w.V^{(k)} + c_1.rand_1.(Pbest^{(k)} - X^{(k)}) + c_2.rand_2.(Rep(h)^{(k)} - X^{(k)}) \quad (III.3)$$

$$X^{(k+1)} = X^{(k)} + V^{(k+1)} \quad (III.4)$$

où :

$X$  : est la position des particules,

$V$  : est la vitesse des particules,

$w$  : est le paramètre d'inertie,

$rand_1, rand_2$  : sont des variables aléatoires entre 0 et 1,

$c_1, c_2$  : sont des constantes positives,

$k$  : est la variable d'itération,

$pbest(i)$  : est la meilleure solution personnelle obtenue en comparant la solution actuelle et la solution présente en mémoire. Si aucune ne domine l'autre, l'une des deux est choisie aléatoirement,

$Rep(h)$  : est la meilleure solution globale (leader) choisie dans l'archive,

$h$  : est l'indice de la particule choisie comme leader de l'archive.

## III.3 Contribution à la segmentation d'image par classification multiobjectif

Vu la complexité du problème de la segmentation d'image et la variabilité des données, un critère unique est en général insuffisant pour appréhender l'ensemble des caractéristiques et informations contenues dans l'image. Nous proposons dans, ce chapitre, une méthode de segmentation d'image basée sur une approche de classification multiobjectif (MO\_IFCMS). Dans cette approche, nous mettons en œuvre deux critères de segmentation complémentaires. Le premier critère est celui de la classification par c-moyennes floues amélioré, qui exploite l'information spatiale de l'image et la distance de Mahalanobis. Le deuxième critère prend en compte la position des contours des objets dans l'image segmentée et le nombre de composantes connexes. Les deux critères sont optimisés simultanément à l'aide de l'algorithme d'optimisation par essaim particulaire (OEP) multiobjectif. Le résultat de l'optimisation est un ensemble de solutions de segmentation réparties sur le front de Pareto, où aucune des solutions n'est meilleure qu'une autre pour les deux critères. Au lieu de choisir une solution dans le front de Pareto, nous proposons de construire une solution optimale en fusionnant toutes les solutions de ce front. Cette méthode est présentée en détail ci-dessous.

### III.3.1 Représentation et initialisation

Dans notre méthode, chaque particule est un vecteur, où chaque composante représente un centre de classe et la taille de la particule représente le nombre de classes. Par exemple, la particule  $X = (12, 4; 22, 5; 53, 0)$  représente trois centres de classes dont les valeurs sont 12,4; 22,5 et 53,0. Une fois le nombre de classes fixé, les positions et les centres de classes sont initialisés aléatoirement. Les positions des particules sont initialisées dans la plage des niveaux de gris de l'image traitée.

### III.3.2 Critères de segmentation

Dans cette méthode, deux critères de segmentation sont utilisés : le critère de la classification par c-moyennes floues amélioré  $J_{IFCMS}$ , qui mesure la compacité des classes et un critère basé contours, qui mesure la cohérence des contours dans le résultat de la segmentation. Le premier

critère est calculé en utilisant les équations suivantes (pour plus de détails, se reporter à la section II.3.2) :

$$J_{IFCMS} = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(x_j, c_i) + \alpha \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m d^2(\bar{x}_j, c_i) \quad (\text{III.5})$$

$$d^2(x_j, c_i) = (x_j - c_i)^T V_i (x_j - c_i) \quad (\text{III.6})$$

$$V_i = \left| \sum_i \right|^{\frac{1}{p}} \sum_i^{-1} \quad (\text{III.7})$$

$$\sum_i = \frac{\sum_{j=1}^N \sum_{i=1}^C u_{ij}^m (x_j - c_i) (x_j - c_i)^T}{\sum_{j=1}^N \sum_{i=1}^C u_{ij}^m} \quad (\text{III.8})$$

Pour le second critère, nous nous sommes basés sur l'idée partagée que la variance du niveau de gris au niveau des contours et régions doit être minimale pour une segmentation réussie. De plus, afin de réduire le nombre d'artéfacts et erreurs de classification, nous proposons de minimiser le nombre de composantes connexes. Ce deuxième critère est calculé comme suit : après calcul des centres des classes et des degrés d'appartenance, une segmentation d'image est déterminée en utilisant le principe du maximum de degrés d'appartenance. À partir de cette segmentation, on extrait les contours, les régions et les composantes connexes. L'équation du calcul du deuxième critère est la suivante :

$$J_{edge} = \sigma_\Gamma + \max_i(\sigma_{R_i}) + \text{card}(\Omega_c) \quad (\text{III.9})$$

où :

$\sigma_\Gamma$  : est la variance des niveaux de gris du contour  $\Gamma$  de l'image gradient de l'image originale,

$\sigma_{R_i}$  : est la variance des niveaux de gris des pixels de la  $i^{\text{ème}}$  région (classe)  $R$ ,

$\text{card}(\Omega_c)$  : représente le cardinal de l'ensemble des composantes connexes ( $\Omega_c$ ) formant l'image segmentée.

Les performances de la classification multiobjectif dépendent énormément des critères utilisés. Plus ils sont contradictoires, mieux c'est [Handl & Knowles, 2005]. Le premier critère utilisé ( $J_{IFCMS}$ ) mesure la variance globale des classes. La valeur minimale de ce critère donne une classification meilleure. Le deuxième critère utilisé ( $J_{edge}$ ) mesure la cohérence des contours du résultat de la segmentation. Il est composé de la variance des contours, du nombre de com-

---

### III.3 Contribution à la segmentation d'image par classification multiobjectif

---

posantes connexes et de la variance maximale des classes. Ces deux critères peuvent ne pas atteindre leurs minima pour une même solution (segmentation), surtout en présence de bruits et recouvrement de classes. Ainsi, la minimisation simultanée de ces deux critères dans une approche multiobjectif permet de trouver un ensemble de solutions, dont la variabilité peut être exploitée afin de trouver la solution la plus performante.

#### III.3.3 Processus d'optimisation

Afin de minimiser les deux critères et de trouver les solutions les plus proches de l'ensemble de Pareto optimal, nous utilisons dans ce travail la métaheuristique d'optimisation par essaim particulière multiobjectif présentée dans la section [III.2](#).

#### III.3.4 Combinaison des solutions du front de Pareto

Après convergence du processus d'optimisation, on obtient un ensemble de solutions non dominées (ensemble de Pareto). Ces solutions représentent différents résultats de segmentation. Afin d'avoir une solution unique, la méthode classique la plus répandue consiste à choisir une solution dans cet ensemble en définissant un nouveau critère de sélection. La qualité du résultat final dépend énormément du choix de ce critère, qui n'est pas trivial. En observant cet ensemble de segmentations, nous avons remarqué que certains pixels sont bien classés dans certaines solutions et mal classés dans d'autres et qu'aucune solution ne présente une classification parfaite pour l'ensemble des pixels. Ainsi, au lieu de choisir une solution parmi l'ensemble du front de Pareto, nous proposons de les combiner, afin de construire une solution globale plus performante. La méthode de combinaison des solutions du front de Pareto que nous proposons se déroule en deux étapes : détection des pixels mal classés puis affectation de ces pixels à leurs nouvelles classes. Ces deux étapes sont détaillées dans les sections suivantes.

##### III.3.4.1 Détection des pixels potentiellement mal classés

Dans cette étape, nous exploitons la variabilité de l'ensemble des solutions du front de Pareto et du degré d'appartenance pour extraire les pixels potentiellement mal classés, afin de les affecter en utilisant les autres pixels (pixels de confiance). L'ensemble des pixels potentiellement mal classés est composé de l'union de deux ensembles :

1. L'ensemble des pixels n'étant pas affectés à la même classe dans toutes les solutions du front de Pareto (segmentations) ;

2. L'ensemble des pixels ayant un degré d'appartenance inférieur à un seuil. Ce seuil est déterminé automatiquement pour chaque segmentation du front de Pareto, en calculant la moyenne et la variance des degrés d'appartenance pour chaque classe après affectation des pixels en appliquant le principe du maximum de degrés d'appartenance. L'équation du calcul du seuil est la suivante (Eq. (III.10)) :

$$T_j^{(i)} = \mu_j^{(i)} - \sigma_j^{(i)} \quad (\text{III.10})$$

où :

$T_j^{(i)}$  : est le seuil de la  $i^{\text{ème}}$  classe de la  $j^{\text{ème}}$  solution du front de Pareto,

$\mu_j^{(i)}$  : est la moyenne des degrés d'appartenance de la  $i^{\text{ème}}$  classe de la  $j^{\text{ème}}$  solution du front de Pareto,

$\sigma_j^{(i)}$  : est la variance des degrés d'appartenance de la  $i^{\text{ème}}$  classe de la  $j^{\text{ème}}$  solution du front de Pareto.

### III.3.4.2 Affectation des pixels potentiellement mal classés

La deuxième étape est la reclassification des pixels potentiellement mal classés extraits en utilisant les informations locales fournies par les autres pixels (pixels de confiance). La méthode utilisée est décrite en détail dans la section II.3.3. En effet, un critère d'homogénéité dans un voisinage local de  $5 \times 5$  pour chaque pixel est utilisé (Eq. (III.11)) afin de trouver sa nouvelle classe.

$$J_i^{(j)} = \frac{\alpha \sigma_j^{(i)} + \|x_i - \mu_j^{(i)}\|}{\beta N_j} + \sum_{k=1, k \neq j}^{N_{label}} \sigma_k^{(i)} \quad (\text{III.11})$$

où :

$x_i$  : est le  $i^{\text{ème}}$  pixel potentiellement mal classé à reclasser,

$N_{label}$  : est le nombre de classes présentes dans le voisinage  $N_{3 \times 3}$  du  $i^{\text{ème}}$  pixel détecté,

$N_j$  : est le nombre de pixels appartenant à la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\mu_j^{(i)}$  : est la moyenne locale de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\sigma_j^{(i)}$  : est la variance locale de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel après son affectation à cette classe,

$\sigma_k^{(i)}$  : est la variance locale de la classe  $k$  dans le voisinage  $N_{5 \times 5}$  du  $i^{\text{ème}}$  pixel détecté,

$\alpha$  : est un paramètre qui détermine l'impact de la variance locale dans la réaffectation du pixel détecté,

$\beta$  : est un paramètre qui détermine l'impact du nombre de pixels appartenant à la classe  $j$ .

Ainsi, le pixel extrait  $x_i$  est affecté à la classe  $j$  qui minimise le critère  $J_i^{(j)}$  ( $j = \text{argmin}(J_i^{(j)})$ ). La minimisation de ce critère assure une meilleure compacité locale des différentes classes (pour plus de détails, se reporter à la section [II.3.3](#)).

La démarche générale de la méthode proposée dans ce chapitre est donnée dans l'Algorithme [III.1](#).

## III.4 Résultats expérimentaux

Dans cette section, nous présentons l'évaluation des performances de la méthode proposée dans ce chapitre. Pour ce faire, la méthode proposée est comparée à sept méthodes de segmentation basées sur la classification floue : FCM, FCM-S1, FCM-S2, EnFCM, FGFCM, FLICM et IFCMS. Nous avons utilisé pour l'expérimentation une base de données composée de trois types d'images :

1. Une base d'images synthétiques contenant différents nombres de classes et affectées de différents types et taux de bruits,
2. Une base d'images IRM simulées du cerveau issues de la base BrainWeb [[Bra, 2013](#)],
3. Une base d'images IRM réelles du cerveau issues de la base IBSR (*Internet Brain Segmentation Repository*) du *Center for Morphometric Analysis (CMA)* du *Massachusetts General Hospital* [[IBS, 2013](#)].

Afin de comparer objectivement les différentes méthodes, les mesures suivantes sont utilisées :

–la sous-segmentation (*under segmentation*) ( $Uns$ ), représente le pourcentage de fausses segmentations négatives :

$$Uns = \frac{N_{fp}}{N_n} \times 100 \quad (\text{III.12})$$

–la sur-segmentation (*over segmentation*) ( $OvS$ ), représente le pourcentage de fausses segmentations positives :

$$OvS = \frac{N_{fn}}{N_p} \times 100 \quad (\text{III.13})$$

–l'erreur de segmentation ( $InC$ ), représente le pourcentage total de fausses segmentations :

$$InC = \frac{N_{fp} + N_{fn}}{N} \times 100 \quad (\text{III.14})$$

en désignant par  $N_{fp}$  le nombre de pixels n'appartenant pas à une classe mais affectés à cette classe (faux positifs).  $N_{fn}$  est le nombre de pixels appartenant à une classe mais non affectés à cette classe (faux négatifs).  $N_p$  est le nombre total de pixels appartenant à la classe (positifs).  $N_n$  est le nombre total de pixels n'appartenant pas à la classe (négatifs) et  $N$  est le nombre total de pixels de l'image.

Comme les résultats des méthodes de comparaison dépendent des solutions initiales et comme la méthode proposée est basée sur une technique d'optimisation stochastique, une seule exécution ne permet pas d'analyser objectivement les performances. Ainsi, chaque algorithme est exécuté 10 fois, en utilisant à chaque fois une configuration initiale aléatoire. La moyenne de ces 10 exécutions est retenue.

### III.4.1 Résultats sur des images synthétiques

Afin de comparer la sensibilité des différentes méthodes, elles sont appliquées à des images synthétiques de taille  $128 \times 128$  contenant 2, 3, 4 ou 5 régions (classes) et affectées de bruit Gaussien, uniforme ou sel et poivre. Pour chaque nombre de classes et pour chaque type de bruit, dix images sont créées avec différents taux de bruit. Pour les bruits Gaussien et uniforme, la variance des bruits varie de 0,002 jusqu'à 0,02 avec un pas de 0,02 (les images sont normalisées dans l'intervalle  $[0, 1]$ ). Pour le bruit sel et poivre, le taux varie de 1% jusqu'à 10% avec un pas de 1%.

Les figures III.2 et III.3 illustrent des exemples de segmentations pour des images synthétiques contenant 4 classes et affectées des bruit Gaussien de variance 0,01, uniforme de variance de 0,02 et sel & poivre d'un taux de 4%. Dans cet exemple, en remarque, visuellement, que les méthodes FLICM et FCMS (figures III.3(c) et III.3(d)) donnent des résultats de segmentation cohérents dans le cas d'images affectées de bruit Gaussien et uniforme, mais on remarque toujours la présence d'artefacts. D'autre part, la méthode proposée donne un meilleur résultat de segmentation sans présence d'artefacts (figure III.3(e)). Pour l'image contenant un bruit de type sel & poivre, la méthode proposée dans ce chapitre et la méthode IFCMS donnent de meilleurs résultats que les autres méthodes. On remarque aussi que la qualité des contours de la méthode proposée est supérieure à celle de la méthode IFCMS.

Le tableau III.1 donne les valeurs des paramètres d'évaluation (Eqs. (III.12), (III.13) et (III.14)) obtenus par les différentes méthodes de segmentation sur l'ensemble de la base des images synthétiques. Ce tableau montre que la méthode développée dans ce chapitre donne quantitativement le meilleur score pour les trois types de bruit. Dans la figure III.4 est présentée la mesure de cohérence de segmentation (taux de pixels bien classés,  $100 - InC$ ), pour les quatre meilleures méthodes, en fonction du taux de bruit Gaussien. Cette figure montre que la méthode proposée est plus robuste vis-à-vis du taux de bruit que les autres méthodes.

**Tableau III.1** – Mesures de performance de segmentation des différentes méthodes de segmentation sur la base d'images synthétiques.

| Bruit        | Métrique       | FCM   | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS | MO_IFCMS     |
|--------------|----------------|-------|--------|--------|-------|-------|-------|-------|--------------|
| Gaussien     | <i>UnS</i> (%) | 15,60 | 7,58   | 8,40   | 11,65 | 11,89 | 7,41  | 6,84  | <b>4,68</b>  |
|              | <i>OvS</i> (%) | 35,20 | 21,57  | 23,67  | 28,89 | 29,58 | 19,32 | 18,64 | <b>16,36</b> |
|              | <i>InC</i> (%) | 20,59 | 11,20  | 13,03  | 15,38 | 15,78 | 10,12 | 9,34  | <b>7,12</b>  |
| Uniforme     | <i>UnS</i> (%) | 15,28 | 4,84   | 6,42   | 9,10  | 10,32 | 4,60  | 3,76  | <b>2,04</b>  |
|              | <i>OvS</i> (%) | 31,61 | 16,57  | 17,93  | 22,43 | 26,03 | 16,30 | 14,22 | <b>12,82</b> |
|              | <i>InC</i> (%) | 20,99 | 7,57   | 9,74   | 11,30 | 13,30 | 7,12  | 6,26  | <b>4,90</b>  |
| Sel & poivre | <i>UnS</i> (%) | 12,84 | 11,92  | 7,11   | 13,94 | 18,03 | 16,05 | 6,91  | <b>6,24</b>  |
|              | <i>OvS</i> (%) | 27,43 | 27,74  | 14,35  | 32,11 | 39,66 | 35,29 | 12,84 | <b>12,00</b> |
|              | <i>InC</i> (%) | 13,01 | 16,16  | 7,73   | 18,13 | 24,03 | 18,46 | 6,54  | <b>5,93</b>  |
| Moyenne      | <i>UnS</i> (%) | 14,57 | 8,11   | 7,31   | 11,65 | 13,41 | 9,35  | 5,84  | <b>4,32</b>  |
|              | <i>OvS</i> (%) | 31,41 | 21,96  | 18,65  | 27,81 | 19,31 | 23,63 | 15,23 | <b>13,72</b> |
|              | <i>InC</i> (%) | 18,19 | 11,64  | 10,16  | 14,93 | 17,70 | 11,90 | 7,38  | <b>5,98</b>  |

#### III.4.2 Résultats sur des images IRM simulées

En l'absence de segmentations vérité-terrain de référence pour les images IRM, il est généralement impossible d'évaluer quantitativement les performances des méthodes de segmentation sur ce type d'image. Afin de pallier ce problème, trois solutions sont possibles :

1. Utiliser des images IRM simulées. Ces images sont produites par un simulateur IRM basé sur les équations des modèles de magnétisation nucléaire pour fournir le signal NMR et des modèles de bruits et de volume partiel.
2. Utiliser des images IRM réelles segmentées manuellement (ou semi-manuellement) par des experts.
3. Utiliser des mesures de validation non supervisées.

La première solution a l'avantage de fournir des images très proches des images réelles en ayant un contrôle sur les types et taux de bruits, ainsi que le volume partiel. La deuxième

méthode est sans doute la plus fiable, car elle utilise des images réelles similaires à celles pour lesquelles ces méthodes ont été conçues. Malheureusement, en plus de la variabilité des résultats de segmentation d'un expert à un autre, il est très difficile d'acquérir une base d'images réelles représentative. La troisième solution consistant de mesures de validation non supervisée est la solution la moins fiable, car ces mesures ne sont pas conçues pour un type d'image bien précis et ne prennent pas en considération les caractéristiques spécifiques aux images IRM. De plus, ce genre de mesure est très contesté. En effet, si ces mesures reflétaient parfaitement la qualité de la segmentation, il suffirait d'optimiser directement ces mesures. Ainsi, nous avons utilisé les deux premières solutions dans notre étude. Dans cette section la première méthode est utilisée. La première méthode est utilisée dans cette section, la deuxième dans la section suivante.

Les images que nous utilisons dans cette section proviennent de la base d'images Brain-Web, qui fournit des volumes IRM du cerveau simulés affectés de plusieurs taux de bruits et de volumes partiels ainsi que leurs segmentations vérité-terrain. La disponibilité de ces dernières nous permet d'évaluer et de comparer quantitativement les performances des différentes méthodes de segmentation. Dans notre expérimentation, nous avons utilisé les images axiales de modalité T1 du volume IRM cérébral. Ce volume est affecté de bruits de taux 3%, 7% et 9%, et une intensité de non uniformité de paramètres 0%, 20% et 40% pour chaque taux de bruit. Ces images sont segmentées en quatre classes : fond, liquide céphalorachidien (LCR), matière grise (MG) et matière blanche (MB).

Les figures III.5 et III.6 montrent un exemple de segmentation de la 85<sup>ème</sup> image du volume IRM affecté de bruit de taux 9%, et de 40% d'intensité de non uniformité. Ces figures montrent qu'il n'est pas évident de tirer des conclusions visuellement sur ce type d'image. Néanmoins, on remarque que les méthodes FCM, EnFCM et FGFCM (figures III.5(c), III.6(a) et III.6(b)) donnent de moins bons résultats que les autres méthodes. Les mesures de performances des différentes méthodes sont données dans le tableau III.2. On remarque dans ce tableau que les méthodes FCMS2, FLICM et IFCMS donnent de bons résultats, mais la méthode que nous avons proposée améliore les performances de segmentation et donne de meilleurs scores.

### III.4.3 Résultats sur des images IRM réelles

Dans cette section, nous utilisons des images IRM du cerveau réelles qui proviennent de la base IBSR (*Internet Brain Segmentation Repository*) du *Center for Morphometric Analysis* (CMA) du *Massachusetts General Hospital* [IBS, 2013]. Cette base fournit en plus des volumes

**Tableau III.2** – Mesures de performance de segmentation des différentes méthodes de segmentation sur la base d’images IRM simulées.

| Région           | Métrique       | FCM  | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS | MO_IFCMS    |
|------------------|----------------|------|--------|--------|-------|-------|-------|-------|-------------|
| LCR <sup>a</sup> | <i>UnS</i> (%) | 0,82 | 1,00   | 0,54   | 1,20  | 1,05  | 0,57  | 0,54  | <b>0,42</b> |
|                  | <i>OvS</i> (%) | 5,34 | 8,77   | 8,40   | 10,22 | 11,02 | 8,43  | 8,22  | <b>5,22</b> |
|                  | <i>InC</i> (%) | 1,15 | 1,57   | 1,12   | 1,86  | 1,77  | 1,13  | 1,10  | <b>0,93</b> |
| MG <sup>b</sup>  | <i>UnS</i> (%) | 2,33 | 1,97   | 2,09   | 1,38  | 1,44  | 2,23  | 2,07  | <b>1,27</b> |
|                  | <i>OvS</i> (%) | 8,74 | 8,98   | 6,48   | 13,42 | 13,03 | 6,45  | 6,43  | <b>6,16</b> |
|                  | <i>InC</i> (%) | 3,78 | 3,55   | 3,08   | 4,10  | 4,06  | 3,12  | 3,05  | <b>2,74</b> |
| MB <sup>c</sup>  | <i>UnS</i> (%) | 1,53 | 1,42   | 1,22   | 1,97  | 2,01  | 1,25  | 1,21  | <b>1,02</b> |
|                  | <i>OvS</i> (%) | 7,38 | 5,33   | 5,14   | 5,83  | 5,78  | 5,16  | 5,12  | <b>4,76</b> |
|                  | <i>InC</i> (%) | 2,77 | 2,25   | 2,05   | 2,79  | 2,81  | 2,06  | 2,03  | <b>1,72</b> |
| Moyenne          | <i>UnS</i> (%) | 1,56 | 1,46   | 1,28   | 1,51  | 1,50  | 1,35  | 1,27  | <b>0,90</b> |
|                  | <i>OvS</i> (%) | 7,15 | 7,69   | 6,67   | 9,82  | 9,94  | 6,68  | 6,59  | <b>5,38</b> |
|                  | <i>InC</i> (%) | 2,56 | 2,45   | 2,08   | 2,91  | 2,88  | 2,10  | 2,06  | <b>1,79</b> |

*a.* Liquide céphalorachidien

*b.* Matière grise

*c.* Matière blanche

IRM réels, des segmentations vérité-terrain réalisées d’une manière semi-automatique par des experts. Ce qui nous permet d’évaluer quantitativement les performances des différentes méthodes de segmentation. Dans notre expérimentation, nous avons utilisé la 115<sup>ème</sup> image de 18 volumes de modalité T1. Ces images sont aussi segmentées en quatre classes (régions) : fond, liquide céphalorachidien (LCR), matière grise (MG) et matière blanche (MB).

Les figures III.7 et III.8 montrent un exemple de segmentation d’image IRM réelle. On remarque aussi qu’il n’est pas évident de comparer visuellement les différentes méthodes. Les valeurs des mesures de performance des différentes méthodes sont présentées dans le tableau III.3. On remarque que, pour ce type d’image, la méthode proposée donne de meilleurs scores que les autres méthodes.

## III.5 Conclusion

Ces dernières années, la tendance en matière de segmentation d’image est de combiner deux ou plusieurs méthodes afin de cumuler leurs avantages pour traiter au mieux la variabilité des images et la richesse d’information qu’elles contiennent. Dans cet esprit, nous avons proposé dans ce chapitre une méthode de segmentation d’image basée sur une approche de classifi-

## Chapitre III. Segmentation d'image par optimisation multiobjectif

**Tableau III.3** – Mesures de performance de segmentation des différentes méthodes de segmentation sur la base d'images IRM réelles.

| Région           | Métrique       | FCM   | FCM_S1 | FCM_S2 | EnFCM | FGFCM | FLICM | IFCMS | MO_IFCMS     |
|------------------|----------------|-------|--------|--------|-------|-------|-------|-------|--------------|
| LCR <sup>a</sup> | <i>UnS</i> (%) | 9,06  | 7,82   | 9,19   | 7,41  | 7,87  | 7,91  | 9,17  | <b>5,86</b>  |
|                  | <i>OvS</i> (%) | 30,50 | 26,62  | 28,92  | 28,40 | 27,67 | 27,31 | 29,08 | <b>22,52</b> |
|                  | <i>InC</i> (%) | 9,51  | 8,23   | 9,62   | 7,86  | 8,30  | 8,34  | 9,60  | <b>5,34</b>  |
| MG <sup>b</sup>  | <i>UnS</i> (%) | 3,17  | 3,39   | 3,40   | 2,54  | 2,58  | 3,24  | 3,39  | <b>2,12</b>  |
|                  | <i>OvS</i> (%) | 46,91 | 43,15  | 47,43  | 46,65 | 48,02 | 47,40 | 47,37 | <b>44,74</b> |
|                  | <i>InC</i> (%) | 12,49 | 11,82  | 12,77  | 11,77 | 12,14 | 12,23 | 12,75 | <b>10,12</b> |
| MB <sup>c</sup>  | <i>UnS</i> (%) | 1,38  | 1,71   | 1,42   | 2,85  | 2,74  | 1,52  | 1,42  | <b>1,16</b>  |
|                  | <i>OvS</i> (%) | 17,35 | 17,91  | 18,50  | 13,58 | 14,09 | 16,02 | 18,42 | <b>13,42</b> |
|                  | <i>InC</i> (%) | 3,73  | 4,09   | 3,93   | 4,41  | 4,39  | 4,06  | 3,92  | <b>3,51</b>  |
| Moyenne          | <i>UnS</i> (%) | 4,53  | 4,30   | 4,67   | 4,26  | 4,39  | 4,22  | 4,66  | <b>3,04</b>  |
|                  | <i>OvS</i> (%) | 31,58 | 29,22  | 31,61  | 29,54 | 29,62 | 30,24 | 31,62 | <b>26,89</b> |
|                  | <i>InC</i> (%) | 8,57  | 8,04   | 8,77   | 8,03  | 8,27  | 8,21  | 8,75  | <b>6,32</b>  |

a. Liquide céphalorachidien

b. Matière grise

c. Matière blanche

cation multiobjectif optimisée par la métaheuristique d'optimisation par essaim particulaire (OEP). L'OEP est utilisée pour optimiser simultanément deux critères de segmentation complémentaires (l'un basé contours et l'autre basé régions). Le critère basé région est la mesure de classification par logique floue améliorée. Le deuxième critère utilisé est basé sur des mesures de statistique au niveau des contours et régions détectés ainsi que sur le nombre de composantes connexes afin de réduire l'apparition d'artefacts dus à la présence de bruit dans les images. Le résultat du processus d'optimisation (OEP multiobjectif) est un ensemble de solutions (segmentations) non dominées (front de Pareto) où aucune n'est meilleure que l'autre pour les deux critères. Au lieu de choisir une solution dans cet ensemble en utilisant un nouveau critère de sélection, nous avons proposé une méthode de combinaison de toutes les solutions afin de donner naissance à une meilleure solution globale. La méthode proposée dans ce chapitre a été testée et comparée à d'autres méthodes de la littérature sur une base comportant trois types d'images : des images synthétiques, des images IRM du cerveau simulées et des images IRM du cerveau réelles. Les résultats expérimentaux ont montré que la méthode proposée donne de meilleurs résultats de segmentation, et qu'elle est plus robuste vis-à-vis des types et des de bruit que les méthodes auxquelles nous l'avons comparée.

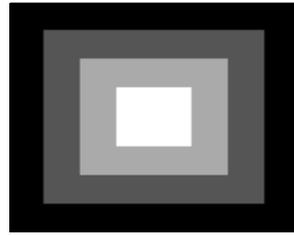
---

**Algorithme III.1** MO\_IFCMS

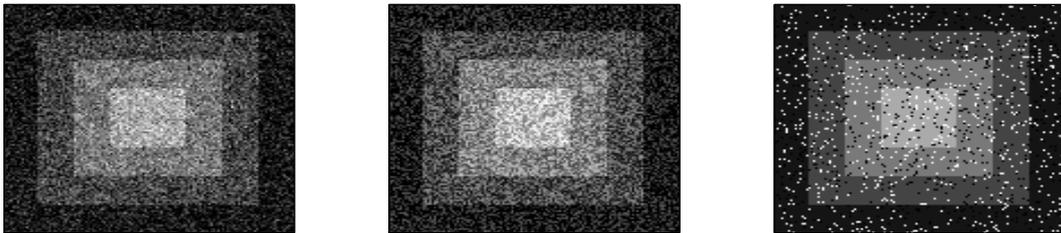
---

**Nécessite :** Le nombre de classes  $C$  et les paramètres de l'OEP multiobjectif.

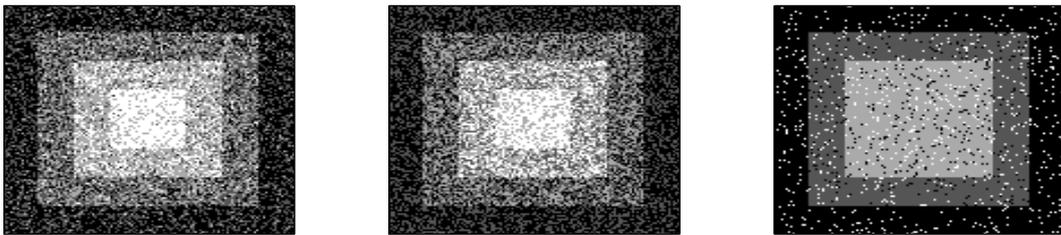
- 1: Initialiser aléatoirement les positions des particules (centres des classes)  $c_i^{(0)}$  dans la plage des niveaux de gris présents dans l'image et initialiser les vitesses des particules à 0.
  - 2: **Pour** chaque particule **Faire**
  - 3:     Calculer les fonctions objectifs en utilisant les équations (Eqs. (III.5)) and ((III.9))
  - 4: **Fin Pour**
  - 5: Sauvegarder les solutions non dominées dans l'archive externe et générer les hypercubes comme système de coordonnées (voir section III.2)
  - 6: Initialiser les meilleures solutions personnelles pour chaque particule avec sa position initiale
  - 7: **Pour**  $k = 1$  Jusqu'au nombre maximal d'itérations **Faire**
  - 8:     **Pour** chaque particule **Faire**
  - 9:         Choisir un leader ( $Rep(h)$ ) de l'archive externe en utilisant la roulette russe (voir section III.2)
  - 10:         Mettre à jour les vitesses des particules en utilisant l'équation (Eq. (III.3))
  - 11:         Mettre à jour les positions des particules (centres des classes) en utilisant l'équation (Eq. (III.4))
  - 12:         Maintenir les positions des particules dans l'espace de recherche. Si la position d'une particule sort du domaine de recherche, elle prend la valeur limite et sa vitesse est multipliée par  $(-1)$
  - 13:         Calculer les fonctions objectifs en utilisant les équations (Eqs. (III.5)) et ((III.9))
  - 14:         Mettre à jour la mémoire des particules : la meilleure solution (au sens de Pareto) entre la nouvelle et l'ancienne est retenue dans la mémoire. Si aucune ne domine l'autre, l'une des deux est choisie aléatoirement.
  - 15:     **Fin Pour**
  - 16:     Mettre à jour le contenu de l'archive externe
  - 17: **Fin Pour**
  - 18: **Pour** chaque solution (particule) dans l'archive externe **Faire**
  - 19:     Calculer les degrés d'appartenance en utilisant l'équation (Eq. (II.12))
  - 20:     Trier les centres de classes pour que chaque classe ait le même label dans l'ensemble des solutions
  - 21:     Déterminer la segmentation d'image en utilisant le principe du maximum du degré d'appartenance
  - 22:     Calculer pour chaque classe le seuil  $T$  en utilisant l'équation (Eq. (III.10))
  - 23: **Fin Pour**
  - 24: Déterminer l'ensemble des pixels n'ayant pas le même label dans l'ensemble des solutions et les pixels ayant un degré d'appartenance inférieur au seuil  $T$ . Ces pixels représentent l'ensemble des pixels potentiellement mal classés.
  - 25: **Pour** chaque pixel extrait ( $x_i$ ) **Faire**
  - 26:     **Pour**  $j = 1$  to  $N_3$  **Faire**
  - 27:         Calculer la fonction objectif  $J_i^{(j)}$  à l'aide de l'équation (Eq. (III.11))
  - 28:     **Fin Pour**
  - 29:     Trouver  $j = argmin(J_i^{(j)})$
  - 30:     Affecter le pixel  $x_i$  à la classe  $j$
  - 31: **Fin Pour**
-



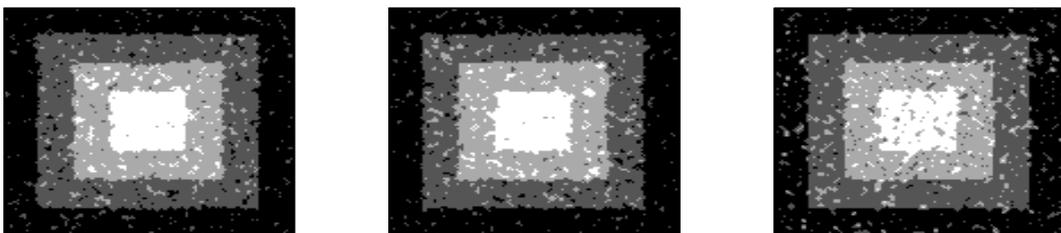
(a) Image originale



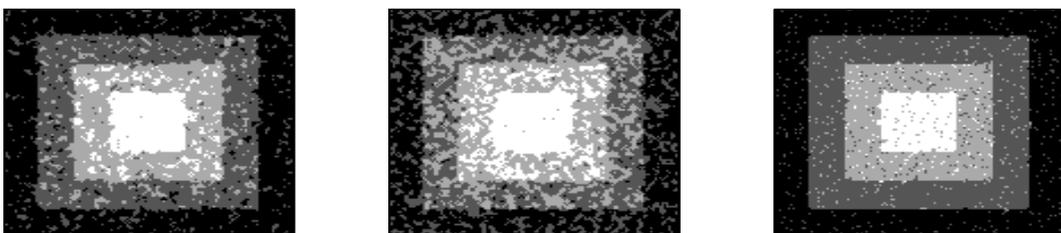
(b) Images bruitées (de gauche à droite : bruit Gaussien, uniforme et sel & poivre)



(c) Résultats de segmentation de la méthode FCM

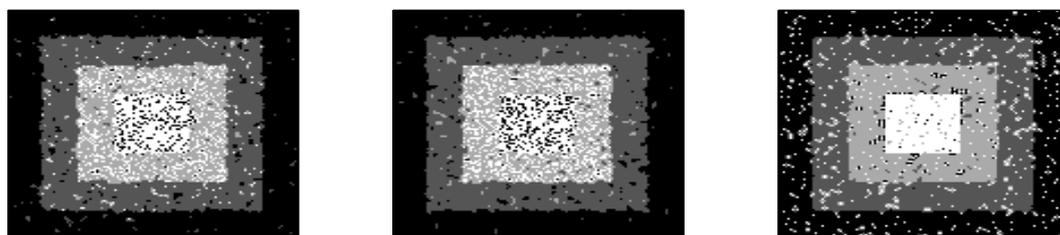


(d) Résultats de segmentation de la méthode FCM-S1



(e) Résultats de segmentation de la méthode FCM-S2

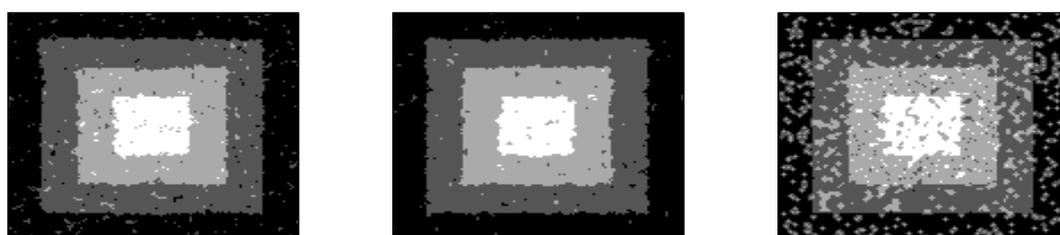
**Figure III.2** – Exemples de segmentations d'images synthétiques.



(a) Résultats de segmentation de la méthode EnFCM



(b) Résultats de segmentation de la méthode FGFCM



(c) Résultats de segmentation de la méthode FLICM

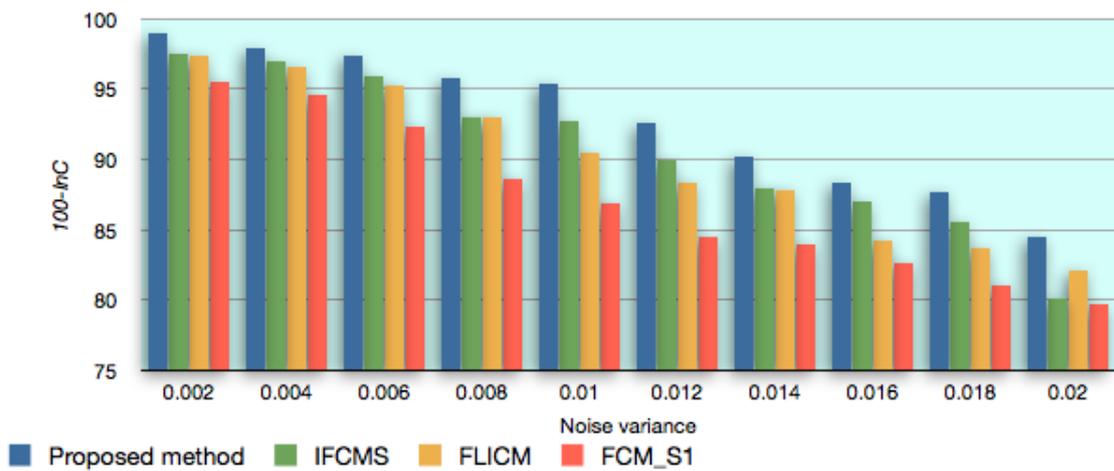


(d) Résultats de segmentation de la méthode IFCMS

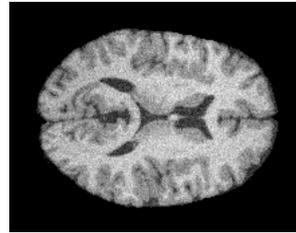


(e) Résultats de segmentation de la méthode proposée

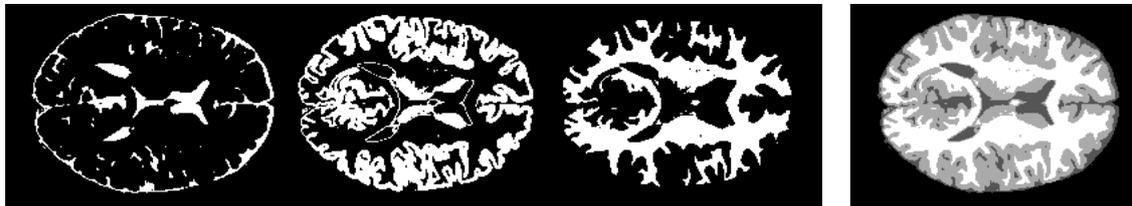
**Figure III.3** – Exemples de segmentations d’images synthétiques (suite des résultats).



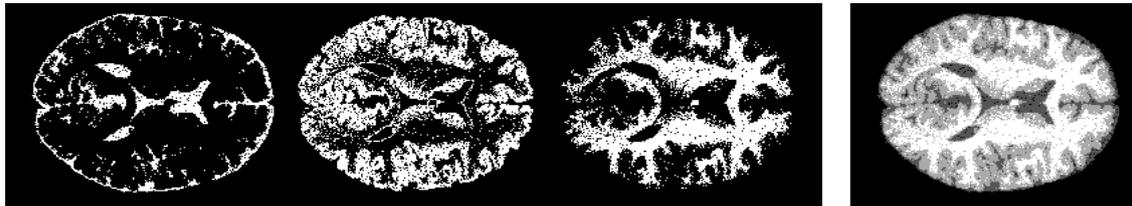
**Figure III.4** – Taux de bonne segmentation des quatre meilleures méthodes en fonction du taux de bruit Gaussien.



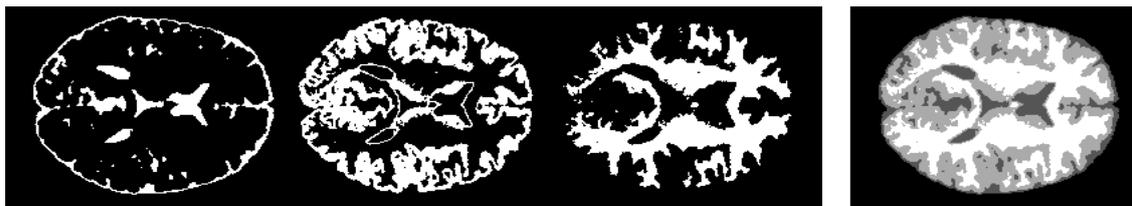
(a) Image à segmenter



(b) Segmentation vérité-terrain (gauche : images régions, droite : segmentation globale)



(c) Résultats de segmentation de la méthode FCM

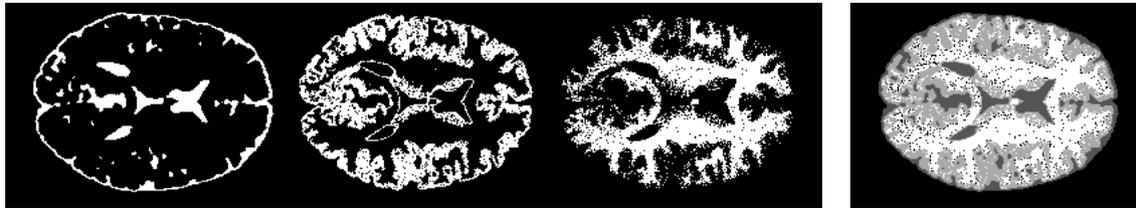


(d) Résultats de segmentation de la méthode FCM-S1

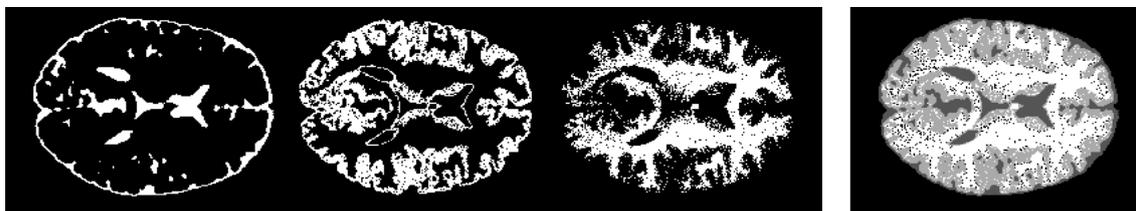


(e) Résultats de segmentation de la méthode FCM-S2

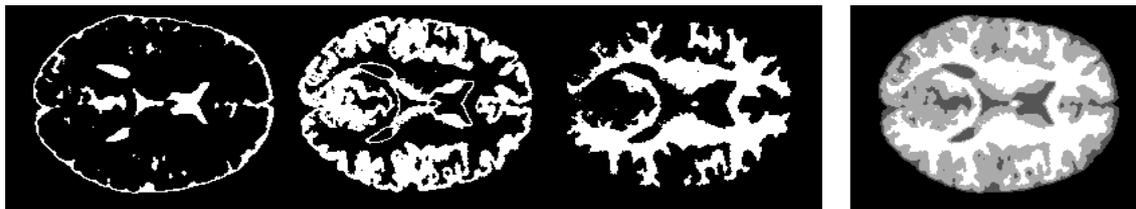
**Figure III.5** – Exemples de segmentations d'images IRM simulées.



(a) Résultats de segmentation de la méthode EnFCM



(b) Résultats de segmentation de la méthode FGFCM



(c) Résultats de segmentation de la méthode FLICM

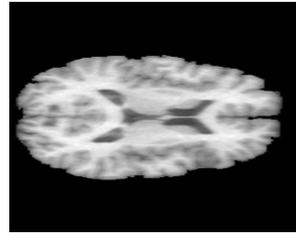


(d) Résultats de segmentation de la méthode IFCMS



(e) Résultats de segmentation de la méthode MO\_IFCMS

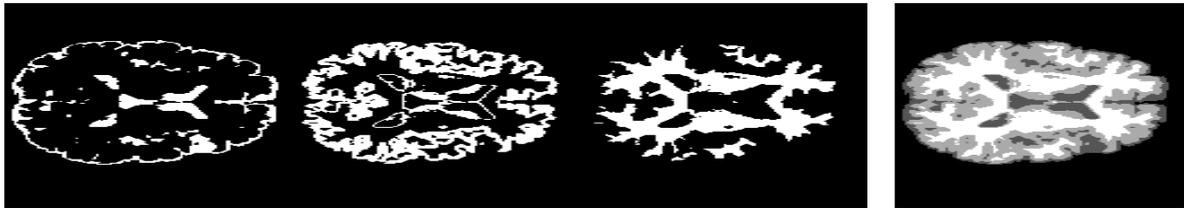
**Figure III.6** – Exemples de segmentations d'images IRM simulées (suite des résultats).



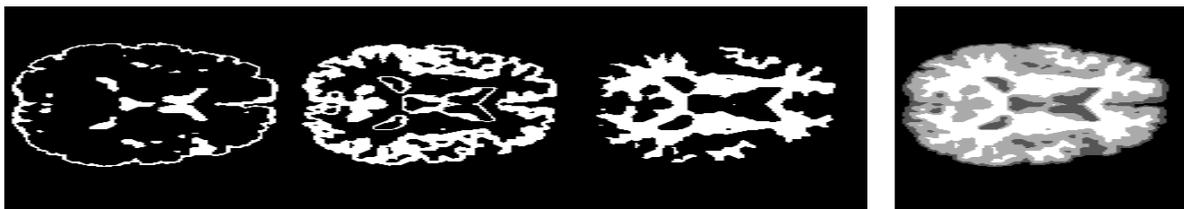
(a) Image à segmenter



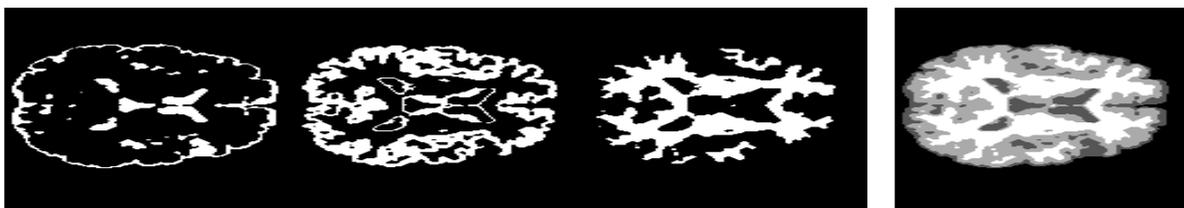
(b) Segmentation vérité-terrain (gauche : images régions, droite : segmentation globale)



(c) Résultats de segmentation de la méthode FCM

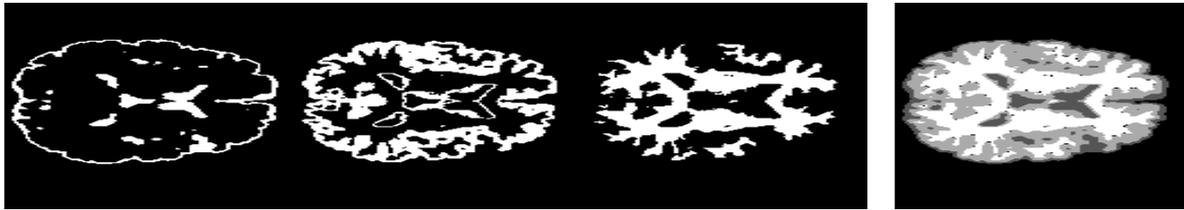


(d) Résultats de segmentation de la méthode FCM-S1



(e) Résultats de segmentation de la méthode FCM-S2

**Figure III.7** – Exemples de segmentations d'images IRM réelles.



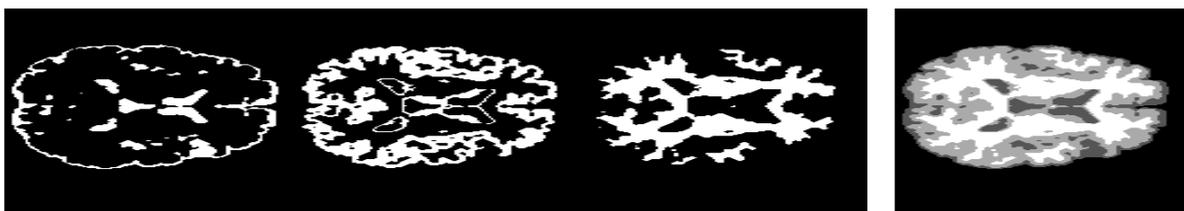
(a) Résultats de segmentation de la méthode ENFCM



(b) Résultats de segmentation de la méthode FGFCM



(c) Résultats de segmentation de la méthode FLICM



(d) Résultats de segmentation de la méthode IFCMS



(e) Résultats de segmentation de la méthode MO\_IFCMS

**Figure III.8** – Exemples de segmentations d'images IRM réelles (suite des résultats).

---

---

# Chapitre IV

---

## Post-segmentation d'image via la métaheuristique des colonies de fourmis

### IV.1 Introduction

Toutes les méthodes de segmentation d'image sont sujettes à des erreurs. L'expérience montre qu'une majeure partie de ces erreurs se manifeste au niveau des frontières entre régions. Dans ces zones, la probabilité d'erreur est en effet relativement élevée. Pour réduire ces erreurs, nous proposons dans ce chapitre une méthode de post-traitement qui nécessite essentiellement l'image originale à segmenter et le résultat de la segmentation de cette image que l'on peut produire par n'importe quelle méthode de segmentation. Cette méthode d'amélioration des résultats de la segmentation est entièrement générique, car elle ne nécessite aucune connaissance a priori sur l'image à segmenter et peut être appliquée en aval de toute méthode de segmentation basée-régions sans aucune modification de l'algorithme. La procédure repose sur une première étape de détection des pixels mal classés suivie d'une étape de reclassification de ces derniers. Une fois les pixels mal classés localisés, leur réaffectation à leurs régions naturelles respectives pose un problème d'optimisation qui est par nature combinatoire. Pour le résoudre, on a opté pour la métaheuristique des colonies de fourmis.

Ce chapitre est organisé comme suit : dans la section [IV.2](#) nous présentons l'optimisation par colonies de fourmis, dans la section [IV.3](#) est détaillée la méthode proposée, suivie dans la section [IV.4](#) par les résultats expérimentaux, enfin nous concluons ce chapitre dans la section [IV.5](#).

## IV.2 Optimisation par colonies de fourmis

Les algorithmes de colonies de fourmis sont une famille de métaheuristiques d’optimisation inspirées du comportement collectif de fourmis réelles dans leur procédé de dépôt et de suivi de pistes. Ces algorithmes ont été initiés par Dorigo *et al.* [Dorigo, 1992; Dorigo *et al.*, 1996] en s’inspirant des travaux de Deneubourg *et al.* [Deneubourg *et al.*, 1983] qui ont modélisé le comportement *aléatoire* des fourmis.

Le premier algorithme d’optimisation, basé sur les colonies de fourmis, intitulé “Ant System” (AS), a été proposé pour résoudre le problème du voyageur de commerce [Dorigo *et al.*, 1996]. Depuis, plusieurs améliorations et variantes ont vu le jour et ont été appliquées dans plusieurs domaines, avec plus ou moins de succès.

Il était remarqué [Goss *et al.*, 1989; Beckers *et al.*, 1992] que les fourmis réelles sont capables d’emprunter le chemin le plus court entre leur nid et une source de nourriture grâce à un comportement collaboratif collectif, sachant qu’aucun individu n’a une vision globale du parcours. Le comportement des fourmis est basé sur les principes suivants :

- *L’auto-organisation* : les fourmis sont capables, en effectuant individuellement des tâches simples, de résoudre des problèmes complexes. Le comportement simple et local de chaque individu fait émerger un modèle (“pattern”, en anglais) de niveau global grâce aux interactions.
- *La stigmergie* : qui est la technique de communication entre les individus (fourmis) en modifiant, d’une manière dynamique, l’environnement où elles évoluent. En effet, en se déplaçant, les fourmis déposent de la phéromone<sup>1</sup> pour marquer le chemin parcouru. En absence de cette substance, les fourmis se déplacent aléatoirement dans l’environnement. Par contre, en sa présence, une fourmi la détecte et peut suivre sa trace avec une probabilité proportionnelle à son intensité. Ainsi, plus une piste est parcourue, plus elle est attirante.
- *Un contrôle décentralisé* : cela signifie qu’il n’y a pas de décision prise à un niveau donné ou par un seul individu. En effet, chaque individu effectue des actions relativement simples en se basant uniquement sur des informations locales de l’environnement, sans

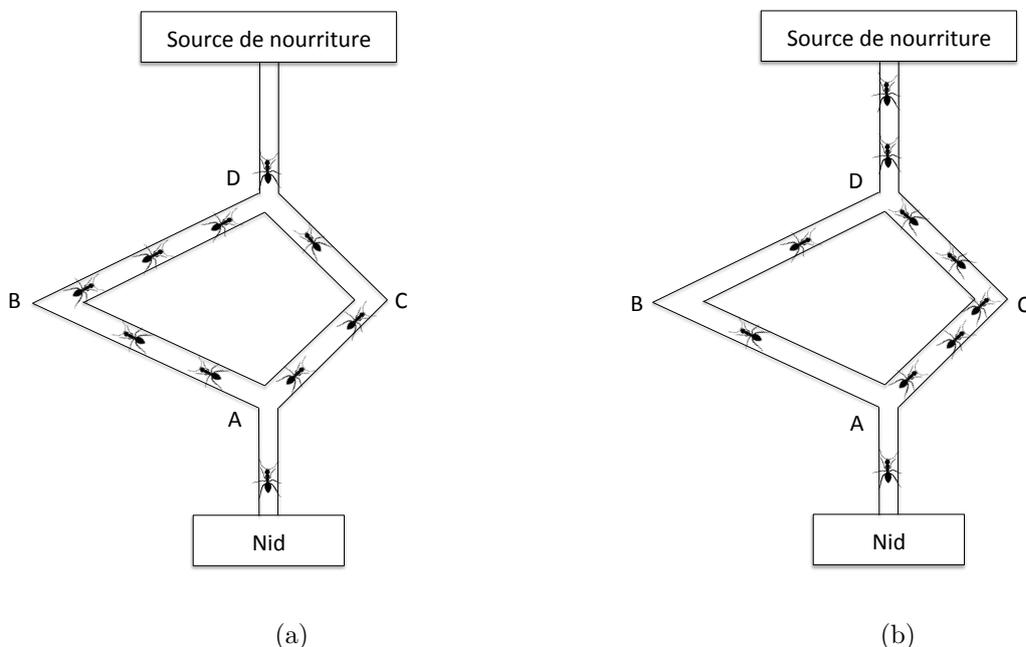
---

1. Substance chimique, qui, émise à dose infime par un animal dans le milieu extérieur, provoque chez un congénère des réactions comportementales spécifiques. (dictionnaire Larousse).

vision du problème dans sa globalité.

- Une *hétérarchie dense* : par opposition à une structure *hiérarchique*, où la population est dirigée par un individu, l'hétérarchie dense est une structure *horizontale*, où les individus sont fortement connectés, agissant ainsi sur les propriétés globales du système.

La figure IV.1 illustre un exemple du processus d'optimisation de chemin entre un nid de fourmis et une source de nourriture. En effet, au début de l'expérience (figure IV.1(a)), les fourmis arrivent au point (A), et comme il n'y a pas de traces de phéromone, elles choisiront l'une des deux directions aléatoirement avec la même probabilité. Les fourmis qui ont emprunté le chemin ACD arriveront, en toute logique, plus rapidement à la source de nourriture que les autres et feront le chemin inverse plus tôt. Ainsi la quantité de phéromone déposée sur le parcours ACD sera plus importante que celle déposée dans ABD. Comme un parcours contenant plus de phéromone a plus de chance d'être emprunté, plus de fourmis vont choisir le parcours ACD (figure IV.1(b)). Ainsi, à terme, le chemin le plus court est renforcé et sera emprunté par la majorité des individus ; de plus, si on considère le phénomène d'évaporation du phéromone, après un certain temps, tous les individus parcourront le chemin le plus court.



**Figure IV.1** – Optimisation du chemin parcouru par une colonie de fourmis. (a) au début de la recherche. (b) à la fin de la recherche.

Afin d'exploiter ce comportement dans un algorithme d'optimisation, une analogie est faite

entre l'environnement où évoluent les fourmis et l'espace de recherche du problème, la source de nourriture (qualité/quantité) et la fonction objectif du problème, et enfin les traces de phéromone et une mémoire. De plus, quelques modifications sont nécessaires pour avoir un algorithme efficace. En effet, une fourmi artificielle (agent) possède une mémoire, n'est pas complètement aveugle (possède des informations sur l'environnement) et le temps est discret [Monmarché, 2000].

### IV.2.1 Système de fourmis "Ant System" (AS)

Dans cette section, nous décrirons le principe général de l'algorithme "Ant System" proposé dans [Dorigo et al., 1996], appliqué au problème du voyageur de commerce.

Initialement, chaque fourmi est placée aléatoirement dans une ville (sommet de graphe), puis se déplace au fil des itérations d'une ville à une autre construisant ainsi un trajet. A chaque itération  $t$ , une fourmi  $k$  choisie de se déplacer de la ville  $i$  où elle se trouve vers une ville  $j$  pas encore visitée avec une probabilité  $p_{ij}^k(t)$  (Eq.(IV.1)) :

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}]^\beta}{\sum_{y \in N_i^k} ([\tau_{iy}(t)]^\alpha \times [\eta_{iy}]^\beta)} & \text{si } j \in N_i^k \\ 0 & \text{sinon} \end{cases} \quad (\text{IV.1})$$

où :

$\tau_{ij}(t)$  : est la densité de trace de phéromone présente dans le chemin  $(i, j)$  à l'itération  $t$ ,

$\eta_{ij}$  : est une mesure de visibilité (heuristique à priori) qui vaut l'inverse de la distance entre la ville  $i$  et la ville  $j$ ,

$\alpha$  et  $\beta$  : sont des paramètres qui contrôlent l'influence respective de la trace de phéromone et de la mesure de visibilité,

$N_i^k$  : est l'ensemble des villes, non visitées encore par la fourmi  $k$ , voisines de la ville  $i$ .

La mise à jour des traces de phéromone est effectuée une fois que toutes les fourmis ont visité l'ensemble des villes à l'aide de l'équation (Eq. (IV.2)) :

$$\tau_{ij}(t+1) = (1 - \rho) \times \tau_{ij}(t) + \sum_{k=1}^{Nb\_ants} \Delta\tau_{ij}^k \quad (\text{IV.2})$$

où :

$\rho$  : est un paramètre d'évaporation de phéromone ( $0 \leq \rho \leq 1$ ),

$Nn\_ants$  : est le nombre de fourmis,

$\Delta\tau_{ij}^k$  : est la quantité de phéromone déposée par la fourmi  $k$  sur le chemin  $(i, j)$ . Cette quantité est déterminée par l'équation (Eq. (IV.3)) :

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L^k} & \text{si la fourmi } k \text{ a parcouru le chemin } (i, j) \\ 0 & \text{sinon} \end{cases} \quad (\text{IV.3})$$

avec :

$L^k$  : la longueur du trajet parcouru par la fourmi  $k$ ,  
 $Q$  : une constante.

### IV.2.2 Système de colonie de fourmis “Ant Colony System” (ACS)

L'apport majeur dans l'algorithme “Ant Colony System” est l'introduction de la notion de mise à jour locale [Dorigo & Gambardella, 1997; Gambardella & Dorigo, 1996] en plus de la mise à jour qui se produit à la fin de la construction. En effet, pour chaque fourmi (individu) à chaque itération, une mise à jour est effectuée (Eq. (IV.4)) au niveau du dernier chemin (arc) visité :

$$\tau_{ij}(t+1) = (1 - \varphi) \times \tau_{ij}(t) + \varphi \times \tau_0 \quad (\text{IV.4})$$

où :

$\varphi$  : représente le paramètre d'évaporation de phéromone ( $0 \leq \varphi \leq 1$ ),  
 $\tau_0$  : représente la valeur initiale des traces de phéromone.

Cette évaporation du phéromone dans le chemin parcouru favorise la diversification, en encourageant les fourmis à parcourir de nouveaux chemins.

La mise à jours globale des traces de phéromone est effectuée à la fin de chaque itération en utilisant l'équation suivante (Eq. (IV.5)) :

$$\tau_{ij}(t+1) = \begin{cases} (1 - \rho) \times \tau_{ij}(t) + \rho \times \Delta\tau_{ij} \\ \tau_{ij}(t) \end{cases} \quad (\text{IV.5})$$

De plus, durant le processus de construction, la règle de proportionnalité pseudo-aléatoire est utilisée. En effet, la probabilité pour une fourmi de se déplacer d'une ville  $i$  vers une ville  $j$  dépend d'une variable aléatoire  $q$  distribuée uniformément sur l'intervalle  $[0, 1]$ , et d'un paramètre  $q_0$ . Ainsi,  $j = \arg \max_{c_{il} \in N(s^p)} \{\tau_{il} \eta_{il}^\beta\}$  si  $q \leq q_0$ , équation (IV.1) sinon.

La méthode de post-segmentation que nous proposons est inspirée de cette version de l’algorithme d’optimisation par colonie de fourmis, i.e. “Ant Colony System”.

## **IV.3 Contribution à la segmentation d’image par colonie de fourmis**

La méthode que nous proposons est un processus de post-traitement visant à corriger les erreurs de la segmentation produite par un algorithme de segmentation basée-régions. La première étape de notre méthode est la détection des erreurs potentielles. La deuxième étape consiste à corriger ces erreurs. L’étape de détection est décrite dans la section [II.3.3](#). Une fois les pixels potentiellement mal classés détectés, le problème consiste à réaffecter chacun d’eux à sa nouvelle classe en minimisant un certain critère. Ce problème est un problème d’optimisation combinatoire. Pour le résoudre, nous avons opté pour la métaheuristique des colonies de fourmis. Cette méthode est détaillée dans les sections suivantes.

### **IV.3.1 Représentation et initialisation**

Dans notre méthode, chaque pixel détecté comme potentiellement mal classé peut être considéré comme une fourmi. Le déplacement des fourmis d’une ville à une autre représente par analogie le changement d’appartenance d’un pixel aux différentes classes. Cependant, quelques différences subsistent. Dans notre modèle, une solution est construite après un seul changement de classe pour tous les pixels, alors que, dans le problème du voyageur de commerce, une solution est construite après un tour complet des fourmis. De plus, dans notre modèle, un pixel peut ne pas changer de classe, mais il est considéré comme un déplacement (dépôt de phéromone).

Initialement, les traces de phéromone sont déposées dans toutes les directions possibles, i.e. toutes les réaffectations possibles des pixels. La valeur de densité de phéromone initiale  $\tau^{(0)}$  est fixée constante à  $\tau_{init}$ .

### **IV.3.2 Processus de construction**

A chaque itération  $t$ , chaque pixel  $k$  change son label actuel  $i$  et prend le label de l’un des pixels ( $j$ ) présents dans son voisinage immédiat  $N_{3 \times 3}$ , en fonction de la fonction de transition de probabilité (Eq. [\(IV.1\)](#)).

---

### IV.3 Contribution à la segmentation d'image par colonie de fourmis

Le point le plus important à considérer dans le processus de construction est la mesure de visibilité  $\eta_{ij}$ . Dans notre travail, nous définissons une mesure de distance entre le pixel en question et les différentes classes dans son voisinage immédiat (Eq. (IV.6)) :

$$\eta_{ij} = \frac{1}{d_{ij}} = \frac{e^{\left(\frac{-|x_i - \mu_j|}{\sigma_j^2 \times Nb_j}\right)}}{\sum_{l \in \Omega_i} e^{\left(\frac{-|x_i - \mu_l|}{\sigma_l^2 \times Nb_l}\right)}} \quad (\text{IV.6})$$

où :

$x_i$  : est la valeur du niveau de gris du pixel  $i$ ,

$\mu_j$  : est la moyenne des niveaux de gris de la classe  $j$  dans le voisinage immédiat  $N_{5 \times 5}$  du pixel  $i$ ,

$\sigma_j$  : est la variance des niveaux de gris de la classe  $j$  dans le voisinage immédiat  $N_{5 \times 5}$  du pixel  $i$ ,

$Nb_j$  : est le nombre de pixels de la classe  $j$  dans le voisinage immédiat  $N_{5 \times 5}$  du pixel  $i$ ,

$\Omega_i$  : est l'ensemble des classes présentes dans un voisinage immédiat  $N_{3 \times 3}$  du pixel  $i$ .

Cette mesure permet d'évaluer la distance entre une classe et un pixel, en tenant compte des caractéristiques statistiques, ainsi que du nombre de pixels de confiance présents dans cette classe. L'avantage de cette mesure est qu'elle est basée uniquement sur l'information locale dans l'image. En effet, un pixel possède des caractéristiques plus proches de celles de ses voisins, que celles des pixels éloignés. Ainsi, cette mesure complète un critère global utilisé dans la méthode de segmentation initiale.

#### IV.3.3 Mises à jour

Dans la méthode proposée, deux opérations de mise à jour de phéromone sont effectuées :

- *Mise à jour locale* : après chaque itération et pour chaque individu (pixel) une mise à jour s'effectue en utilisant l'équation (Eq. (IV.7)) :

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \rho \times \Delta_{ij} \quad (\text{IV.7})$$

où :

$\rho$  : est le paramètre d'évaporation locale.

$\Delta_{ij} = \eta_{ij}$ .

- *Mise à jour globale* : après déplacement de tous les pixels, la mise à jour des traces de phéromone s'effectue en utilisant l'équation (Eq. (IV.8)) :

$$\tau(t+1) = (1 - \psi)\tau(t) + \psi \times \Theta_t \quad (\text{IV.8})$$

où :

$\psi$  : représente le paramètre d'évaporation globale.

$\Theta(t)$  : représente le rapport entre la meilleure valeur actuelle de la fonction objectif et la meilleure valeur trouvée jusqu'à présent (Eq. (IV.9)) :

$$\Theta(t) = \frac{\min_{k \in [0,t]} \{J(k)\}}{J(t)} \quad (\text{IV.9})$$

où :

$J(k)$  : est la valeur de la fonction objectif globale à l'itération  $t$ .

La valeur de  $\Theta(t)$  représente le taux d'amélioration du critère. Ainsi, une amélioration de ce dernier induit un dépôt de phéromone proportionnel à cette amélioration, incitant ainsi les individus à se diriger vers cette zone (prometteuse), et à l'inverse une détérioration du critère provoque une évaporation des traces de phéromone.

Le critère  $J$  que nous utilisons est celui de la segmentation par contour actif défini par les équations suivantes (Eqs. (IV.10)) :

$$J(\nu) = E_{interne}(\nu) + E_{externe}(\nu) \quad (\text{IV.10})$$

où  $\nu$  représente un contour (une solution). Ce critère est composé de deux énergies : interne et externe.

L'énergie interne  $E_{interne}$  reflète la cohérence de la forme géométrique des contours ; sa valeur est déterminée par l'équation (Eq. (IV.11)) :

$$E_{interne}(\nu) = \frac{1}{2} \int_{\nu} \left( \alpha(s) \left\| \frac{d\nu}{ds}(s) \right\| + \beta(s) \left\| \frac{d^2\nu}{ds^2}(s) \right\| \right) \quad (\text{IV.11})$$

où :

$s$  : est la coordonnée curviligne de la courbe du contour  $\nu$ ,

$\alpha$  et  $\beta$  : sont des paramètres de poids.

---

### IV.3 Contribution à la segmentation d'image par colonie de fourmis

---

Le premier terme de cette fonction est la première dérivée de la courbe, qui représente la tension de la courbe. Plus la courbe se distend, plus sa valeur augmente. Le deuxième terme représente la courbure des contours. Plus la courbe est incurvée, plus sa valeur augmente.

L'énergie externe  $E_{externe}$  reflète l'adéquation des contours aux valeurs des niveaux de gris de l'image ; sa valeur est déterminée par l'équation (Eq. (IV.12)) :

$$E_{externe}(\nu) = - \int_{\nu} \|\nabla I(\nu(s))\|^2 ds \quad (\text{IV.12})$$

où :

$\nabla I(\nu(s))$  : représente les valeurs du gradient de l'image  $I$  en  $\nu(s)$ . Plus sa valeur est importante, meilleure est l'adéquation, i.e. le contour  $\nu(s)$  déterminé se trouve dans des pixels de fort gradient.

La méthode proposée dans ce chapitre est présentée dans l'algorithme IV.1.

---

#### Algorithme IV.1 Post-segmentation par colonie de fourmis

---

**Nécessite :** Une image  $I$ , sa segmentation  $I_{set}$  par une méthode basée-régions et les différents paramètres de la méthode.

- 1: Détecter l'ensemble des pixels potentiellement mal classés par la méthode décrite dans la section II.3.3 .
  - 2: Initialiser les traces de phéromone  $\tau^{(0)}$  à la valeur  $\tau_{init}$ .
  - 3: **Pour**  $t = 1$  Jusqu'au nombre maximal d'itérations **Faire**
  - 4:     **Pour** chaque pixel  $i$  potentiellement mal classé (fourmi) **Faire**
  - 5:         **Pour** chaque classe  $j$  présente dans le voisinage  $N_{3 \times 3}$  du pixel  $i$  **Faire**
  - 6:             Calculer la moyenne locale  $\mu_j$  de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  de  $i$ .
  - 7:             Calculer la variance locale  $\sigma_j$  de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  de  $i$ .
  - 8:             Déterminer le nombre de pixels  $Nb_j$  de la classe  $j$  dans le voisinage  $N_{5 \times 5}$  de  $i$
  - 9:         **Fin Pour**
  - 10:         Calculer la mesure de visibilité à l'aide de l'équation (IV.6).
  - 11:         Changer la classe du pixel  $i$  suivant la probabilité déterminée par (IV.1).
  - 12:         Mettre à jour la trace de phéromone dans le chemin emprunté à l'aide de l'équation (IV.7) (mise à jour locale).
  - 13:     **Fin Pour**
  - 14:     Calculer la valeur de la fonction objectif avec l'équation (IV.10).
  - 15:     Mettre à jour les traces de phéromone à l'aide de l'équation (IV.8) (mise à jour globale).
  - 16:     Mettre à jour la meilleure solution et sa *fitness* trouvée jusqu'à présent.
  - 17: **Fin Pour**
-

## IV.4 Résultats expérimentaux

Dans cette section, nous présentons des résultats expérimentaux de la méthode proposée sur la base d'images synthétiques. Cette méthode est appliquée comme un processus de post-traitement de segmentation.

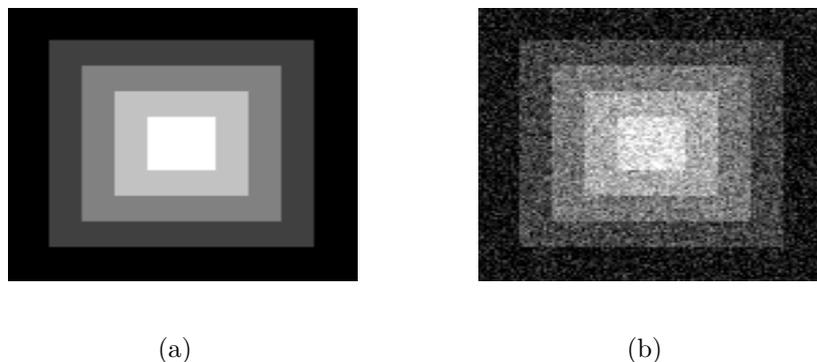
La base d'images utilisée contient des images synthétiques de taille  $128 \times 128$ , contenant 2, 3, 4 ou 5 classes et affectées de bruits Gaussien, Uniforme ou sel & poivre. Les bruits Gaussien et uniforme sont d'une variance de 0,002 jusqu'à 0,01 avec un pas de 0,02 (les images sont normalisées dans l'intervalle  $[0, 1]$ ). Le bruit sel & poivre est d'un taux variant de 1% jusqu'à 5% avec un pas de 1%.

La mesure de performance utilisée est la mesure de cohérence de la segmentation  $SA$  définie dans l'équation (II.17). Cette mesure représente le taux de pixels bien classés. La valeur retenue est la moyenne de 10 exécutions.

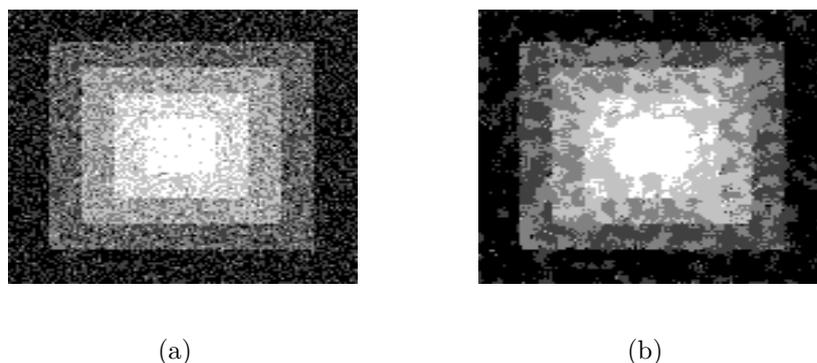
Nous avons appliqué notre méthode après utilisation des méthodes de segmentation basées sur la logique floue (car réputées pour leurs performances) : FCM, FCM-S1, FCM-S2 et FLICM.

La figure IV.2 montre un exemple d'image, de la base utilisée, contenant 5 classes et affectée d'un bruit Gaussien de variance 0,008. Les résultats de segmentation de cette image avec les méthodes FCM, FCM-S1, FCM-S2 et FLICM, avec et sans utilisation de post-segmentation, sont présentés dans les figures IV.3, IV.4, IV.5 et IV.6, respectivement. Dans ces figures, on remarque que les résultats de segmentation sont améliorés dans les cas d'utilisation de la post-segmentation. Néanmoins, le résultat de la méthode FCM avec post-segmentation n'est pas acceptable. Cela est dû à la qualité de la segmentation sans post-segmentation. En effet, la méthode que nous proposons est basée sur les pixels de confiance. Un nombre très faible de ces pixels ne peut pas conduire à une correction acceptable.

Le tableau IV.1 donne les résultats quantitatifs des résultats de segmentation des différentes méthodes, avec et sans utilisation de la post-segmentation. On remarque que la méthode proposée, pour les quatre méthodes, arrive à améliorer les résultats de la segmentation.



**Figure IV.2** – Exemple d’image synthétique. (a) image originale. (b) image affectée de bruit Gaussien.



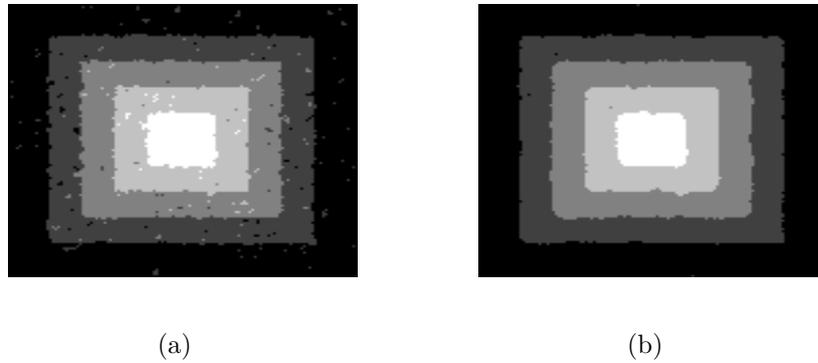
**Figure IV.3** – Exemple de résultat de segmentation par la méthode FCM. (a) sans post-segmentation. (b) avec post-segmentation

|              | FCM   | FCM+PS       | FCM-S1 | FCM-S1+PS    | FCM-S2 | FCMS2+PS     | FLICM | FLICM+PS     |
|--------------|-------|--------------|--------|--------------|--------|--------------|-------|--------------|
| Gaussien     | 60,17 | <b>68,44</b> | 87,12  | <b>94,04</b> | 85,42  | <b>93,18</b> | 93,99 | <b>95,43</b> |
| Uniforme     | 54,25 | <b>67,03</b> | 86,36  | <b>93,34</b> | 72,71  | <b>90,59</b> | 96,14 | <b>98,12</b> |
| Sel & poivre | 71,61 | <b>77,22</b> | 83,02  | <b>86,69</b> | 94,50  | <b>95,39</b> | 65,02 | <b>71,87</b> |

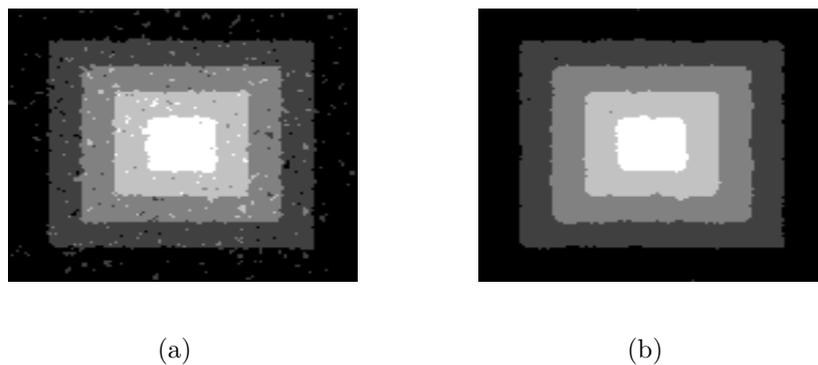
**Tableau IV.1** – Cohérence de segmentation moyenne ( $SA\%$ ) des différentes méthodes, avec et sans utilisation de la post-segmentation (PS).

## IV.5 Conclusion

Dans ce chapitre, nous avons présenté une méthode de post-segmentation qui permet de corriger les erreurs de segmentation produites par les méthodes de segmentation basées-régions. La méthode développée permet de détecter et de reclasser les pixels potentiellement mal classés. Ces pixels sont ceux situés au niveau des frontières entre régions, là où se produisent généralement les erreurs. Une fois ces pixels détectés, on est confronté à un problème d’optimisation

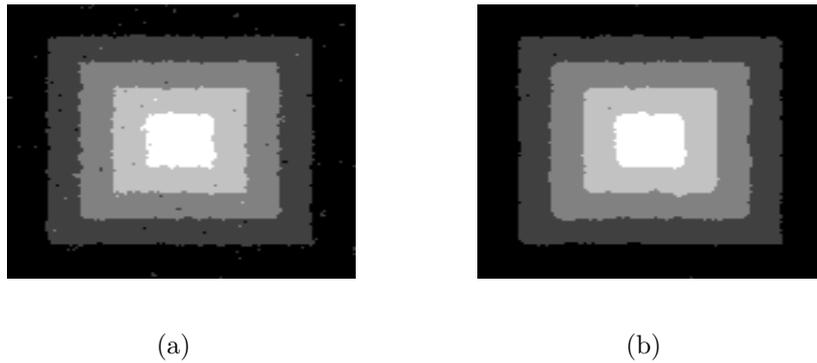


**Figure IV.4** – Exemple de résultat de segmentation par la méthode FCM-S1. (a) sans post-segmentation. (b) avec post-segmentation.



**Figure IV.5** – Exemple de résultat de segmentation par la méthode FCM-S2. (a) sans post-segmentation. (b) avec post-segmentation.

combinatoire. Pour résoudre ce problème, nous avons utilisé la métaheuristique d'optimisation par colonie de fourmis dans sa version "Ant Colony System". La méthode proposée a été testée sur une base d'images synthétiques contenant plusieurs types et taux de bruit, pour corriger les résultats de segmentation de plusieurs algorithmes de segmentation. Les résultats montrent que notre méthode permet de corriger la majorité des erreurs.



**Figure IV.6** – Exemple de résultat de segmentation par la méthode FLICM. (a) sans post-segmentation. (b) avec post-segmentation.

---

---

# Chapitre V

---

## Application aux images de type "Tomographie par Émission de Positons" (TEP)

### V.1 Introduction

Dans ce chapitre, nous exploitons les différentes méthodes développées dans les chapitres précédents pour traiter le problème de segmentation des images de type tomographie par émission de positons (TEP).

Ce chapitre est organisé comme suit : dans la section [V.2](#) nous présentons l'imagerie TEP, dans la section [V.3](#) nous présentons les principales caractéristiques et difficultés de la segmentation des images TEP, dans la section [V.4](#), nous présentons les résultats de l'application des différentes contributions. Ce chapitre est finalement conclu dans la section [V.5](#).

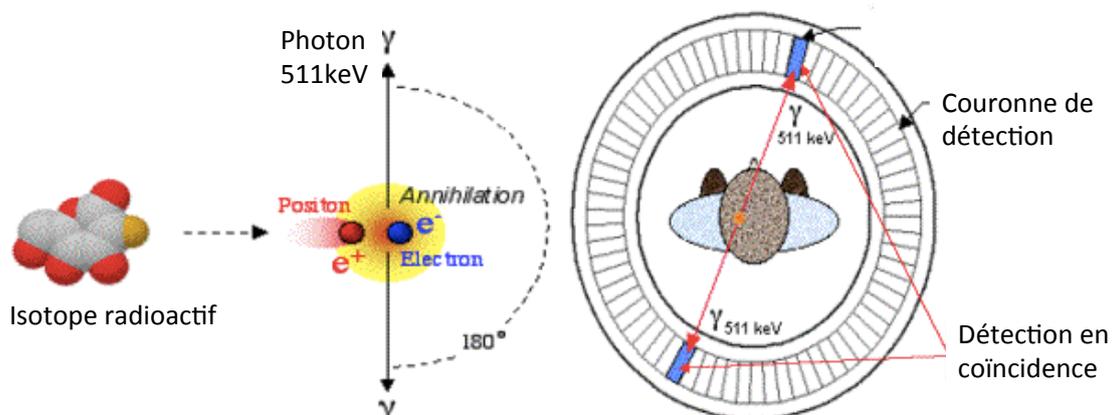
### V.2 La tomographie par émission de positons

L'imagerie nucléaire consiste à mesurer la distribution d'un traceur radioactif injecté par voie orale ou intraveineuse, générant ainsi des images fonctionnelles du corps humain. Il existe principalement trois types d'imagerie en médecine nucléaire [[Toennies, 2012](#)] : la scintigraphie, la tomographie d'émission monophotonique (TEMP) et la tomographie par émission de positons (TEP). Dans cette section, nous nous intéressons uniquement aux images TEP, du fait de la coopération existante entre le laboratoire LISSI et le CHU Henri-Mondor (Créteil) pour traiter ce type d'images. Toutes les images considérées dans ce chapitre nous sont fournies par le CHU Henri-Mondor (Créteil). Ce travail a été effectué en collaboration avec Mme. Olena Tankyevych, qui est maître de conférences au LiSSi.

La TEP permet de générer des volumes 3D de l'activité métabolique (du glucose généralement) dans un corps vivant en se basant sur l'émission de positons. Ces derniers sont émis par la désintégration d'isotopes radioactifs.

### V.2.1 Principe de l'imagerie TEP

Certains isotopes radioactifs (oxygène, fluor, etc.) instables, i.e. en excès de protons, à la rencontre d'électrons (annihilation), émettent deux photons gamma ( $\gamma$ ) dans des directions diamétralement opposées. Ces deux photons, très énergétiques (511 keV), sont interceptés, quasi-simultanément, par un couple de détecteurs situés de part et d'autre d'une couronne de détection (cf. figure V.1). La ligne reliant ces deux détecteurs est appelée *ligne de réponse*. C'est ainsi que la détection en coïncidence des photons permet de situer leur position d'émission. À partir de ces signaux, un volume 3D de la distribution d'activité de l'organisme est construit en faisant apparaître, en fonction de la concentration du traceur, une brillance plus importante dans les zones de forte émission (activité métabolique des cellules). On parle dans ce cas d'imagerie fonctionnelle, qui, à l'inverse de l'imagerie structurale, ne fournit pas des images anatomiques.



**Figure V.1** – Schéma de principe général d'un scanner TEP. (adapté de [www.cermep.fr](http://www.cermep.fr))

## V.2.2 Isotopes radioactifs

Comme mentionné ci-dessus, la TEP est basée sur l'émission de positons générés à partir de certains atomes instables. Ces atomes n'existent pas naturellement. Ils sont créés artificiellement dans des cyclotrons<sup>1</sup>. Cependant, les isotopes émetteurs de positons ont une durée de *demi-vie*<sup>2</sup> relativement courte. Ce qui nécessite la présence d'un cyclotron près du scanner TEP. De plus, le nombre d'isotopes exploitables en TEP est limité, car ces derniers doivent avoir une demi-vie, à la fois, assez longue pour qu'ils soient utilisables, et en même temps assez courte pour éviter une irradiation attardée. De même, l'énergie de l'isotope émis ne doit pas être très grande pour éviter que le parcours dans la matière ne soit important. Les isotopes radioactifs les plus utilisés en TEP sont présentés dans le tableau V.1. Le Fluor 18 (<sup>18</sup>F) est le plus utilisé en oncologie, car il présente un bon compromis entre la demi-vie, l'énergie du positon émis et le parcours dans la matière. Le schéma de désintégration du <sup>18</sup>F est le suivant :



Afin de retourner à l'état stable le <sup>18</sup>F subit une désintégration de type  $\beta^+$  générant un isotope stable de l'oxygène <sup>18</sup>O non désintégré et émet un neutrino ( $\nu$ ) et un positon ( $e^+$ ). L'imagerie TEP s'intéresse à l'émission de ce dernier.

| Isotope          | Demi-vie<br>(minutes) | Énergie maximale<br>du positon (MeV) | Parcours moyen de l'isotope<br>dans l'eau (mm) |
|------------------|-----------------------|--------------------------------------|--|
| <sup>11</sup> C  | 20                    | 0,98                                 | 1,1  |
| <sup>18</sup> F  | 110                   | 0,64                                 | 0,6  |
| <sup>15</sup> O  | 2,0                   | 1,72                                 | 2,7  |
| <sup>13</sup> N  | 10                    | 1,19                                 | 1,5  |
| <sup>68</sup> Ga | 68                    | 1,90                                 | 3,1  |
| <sup>82</sup> Rb | 1,2                   | 3,35                                 | 5,9  |

**Tableau V.1** – Caractéristiques des isotopes radioactifs les plus utilisés en tomographie par émission de positons (TEP).

---

1. Accélérateur circulaire de particules atomiques. (Larousse.)  
 2. Le temps nécessaire pour que la moitié des isotopes reviennent à l'état stable.

### V.2.3 Radiotraceurs

Afin d'ingérer les isotopes dans un organisme vivant, il est nécessaire de les incorporer dans des molécules organiques. Cette opération donne naissance à un *radiotraceur* (ou traceur radioactif). Ce dernier, en fonction de ses caractéristiques, permet de mettre en évidence des activités biochimiques ou physiologiques (métabolisme du glucose, synthèse de protéine, débit sanguin, etc.). Ainsi, selon la fonction ciblée dans l'organisme, il faut déterminer la molécule organique qui s'implique dans cette fonction (si possible, uniquement dans cette fonction, sans la dénaturer), puis enclaver l'isotope radioactif dans cette molécule avant de l'inclure dans l'organisme.

En oncologie, vu que les tumeurs ont un développement rapide, et consomment beaucoup d'énergie pour assurer leurs fonctions de base, la fonction ciblée est le métabolisme du glucose. La molécule porteuse la plus utilisée pour mettre en évidence cette fonction est la *fluoro-déoxy-glucose* (FDG). Estampillée par le  $^{18}\text{F}$  cette molécule produit le radiotraceur  $^{18}\text{F}$ -FDG. Ce radiotraceur se comporte comme le glucose. De plus, il s'agglomère dans les régions qui le métabolisent permettant ainsi de mesurer l'activité métabolique. Cependant, ce radiotraceur n'est pas discriminatif des tumeurs. En effet, d'autres organes, tels que le cœur et le cerveau, consomment beaucoup d'énergie et présentent une forte activité métabolique. De plus, les reins et la vessie se chargent d'éliminer le glucose non consommé par les cellules, et présentent, donc, une activité radioactive. La figure V.2 montre un exemple d'image TEP où l'on remarque la présence d'activité métabolique dans ces différents organes, en plus des cellules cancéreuses.

### V.2.4 Système de détection

En imagerie TEP, le taux et le lieu d'activité sont mesurés à partir de la détection en coïncidence des deux photons  $\gamma$  d'annihilation. Le système de cette détection est composé d'un cristal inorganique scintillant<sup>3</sup> relié à un tube photomultiplicateur (PMT). Le photon  $\gamma$  interagit avec le cristal qui génère plusieurs photons. Ces photons sont conduits vers le PMT qui les transforme en impulsions électriques. Ces dernières sont ensuite traitées par un système d'acquisition en vue de la reconstruction des images. Dans la figure V.3 est présenté un schéma global du système de détection.

En résumé, les détecteurs placés dans la couronne restent à l'écoute et dès que deux photons sont détectés dans une courte fenêtre de temps (nanoseconde), la ligne de réponse peut

---

3. Un scintillateur est un matériau qui émet de la lumière suite à l'absorption d'un rayonnement ionisant (photon ou particule chargée). (Wikipédia)

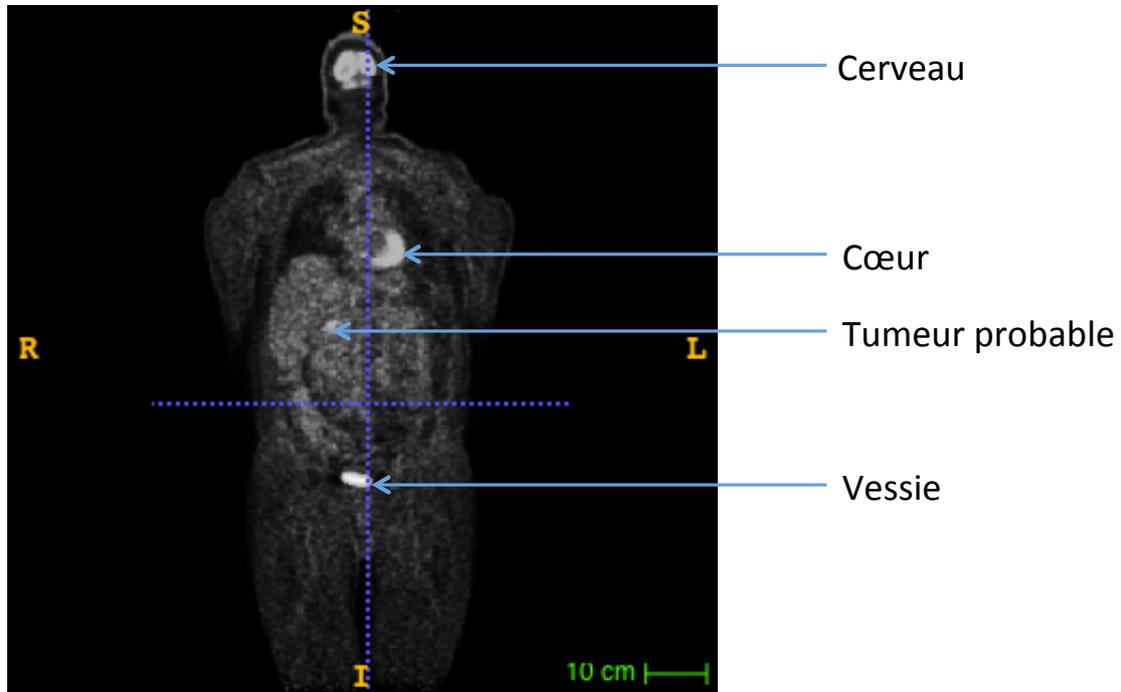


Figure V.2 – Exemple d'image TEP.

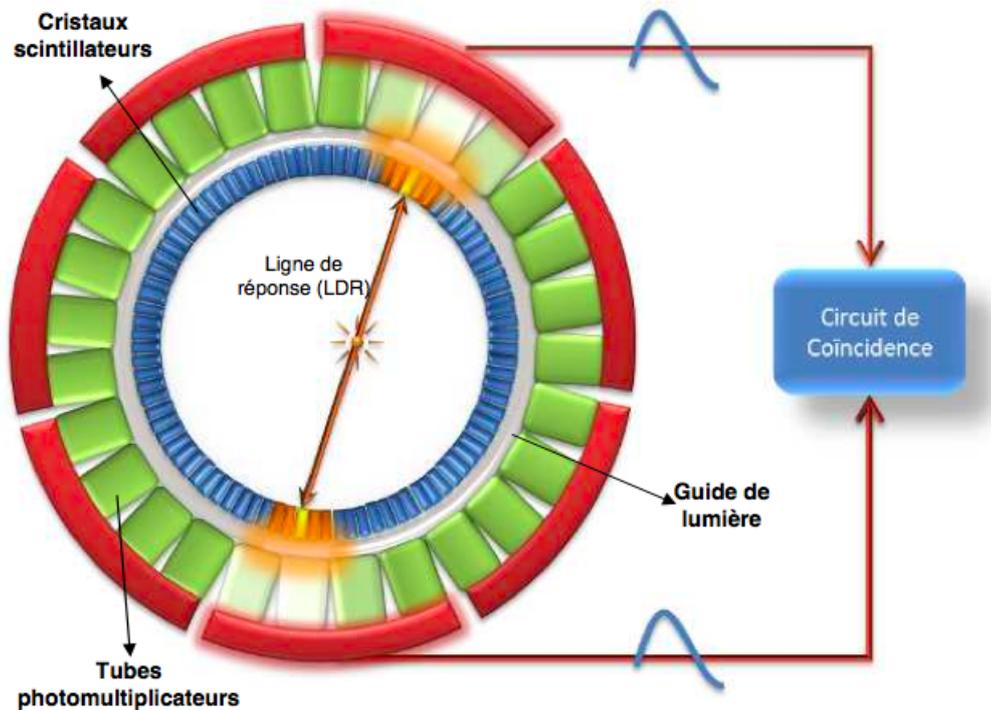
être déterminée en reliant ces deux détecteurs. Les deux photons sont ensuite transformés en signaux électriques analogiques, puis numériques. Ces signaux sont traités par des algorithmes de traitement et de reconstruction pour produire un volume 3D.

## V.3 Caractéristiques des images TEP

Dans cette section, nous présentons les différentes caractéristiques des images TEP afin d'évaluer les difficultés liées à la segmentation de ce type d'images.

### V.3.1 Le bruit

La reconstruction des images TEP est basée sur la détection quasi simultanée et le comptage des photons  $\gamma$  colinéaires résultant de l'émission de positons. Cependant, à cause du caractère aléatoire de l'émission, des erreurs de détection, dues à la colinéarité non parfaite et à l'atténuation des photons, et des bruits relativement importants apparaissent immédiatement dans



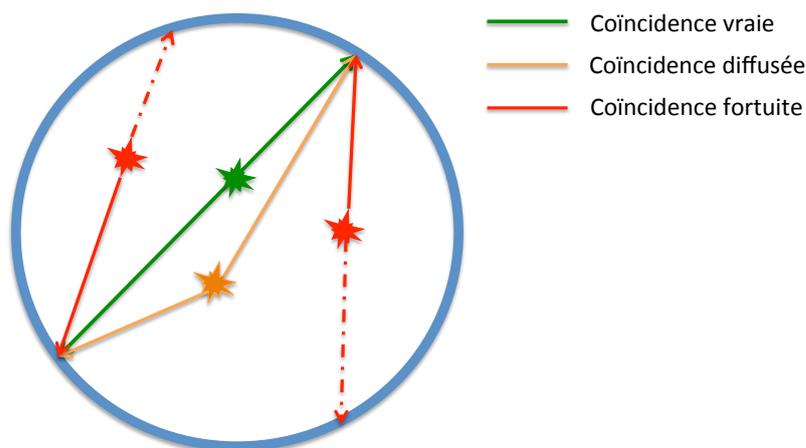
**Figure V.3** – Détection en coïncidence des photons en TEP. (source : [Merheb, 2007])

les images. Ainsi on distingue principalement trois sources de bruit dans les images TEP, en plus du mouvement du patient et des facteurs technologiques (stabilité mécanique, sensibilité des capteurs, etc.) :

1. La désintégration intrinsèquement aléatoire des isotopes,
2. Les fausses coïncidences,
3. L'atténuation des photons.

À cause du caractère aléatoire de l'émission de positons, le nombre de photons émis dans une période de temps subit des variations. Souvent, cette émission aléatoire est modélisée par une distribution de Poisson et traitée en fin de chaîne. Néanmoins, vu que l'acquisition se fait en un temps limité, il est difficile de quantifier ce bruit. En pratique, on réalise généralement plusieurs acquisitions de voxels connus homogènes, ensuite les paramètres de la loi de Poisson sont déterminés à partir de cet ensemble de réalisations, i.e. la différence entre les réalisations doit suivre une loi de Poisson.

En plus de la détection des photons d'annihilation correspondant à des "coïncidences vraies", d'autres photons peuvent être interceptés par les capteurs, créant ainsi une autre source de bruit dans les images TEP. Ces fausses coïncidences sont les coïncidences diffusées et les coïncidences fortuites. La figure V.4 illustre ces différentes coïncidences. La coïncidence vraie résulte de la détection des deux photons de la même désintégration sans subir d'interactions. La coïncidence diffusée est liée à la détection de deux photons d'une même désintégration, qui ont subi des intégrations dans les tissus humains, modifiant ainsi leurs trajectoires. La dernière coïncidence représente les photons captés fortuitement dans la même fenêtre de temps. Ces photons ne proviennent pas de la même désintégration. Ce type d'erreurs est corrigé en pratique en éliminant certains photons détectés en fonction de leurs énergies.



**Figure V.4** – Les différentes coïncidences susceptibles d'être interceptées par la même paire de capteurs en imagerie TEP.

La dernière source de bruit est due à l'atténuation des photons, car ces derniers, par effet photoélectrique, sont absorbés par les électrons de la matière, et perdent leur énergie. Ainsi l'activité mesurée subit une décroissance, qui suit une loi exponentielle en fonction du milieu traversé et de la distance parcourue. Les erreurs causées par l'atténuation sont en pratique corrigées par l'estimation de la cartographie des coefficients d'atténuation du patient en utilisant des sources externes ( $\gamma$ -caméra) ou à partir d'images CT (*computed tomography*).

### V.3.2 La résolution spatiale

La résolution spatiale peut être définie comme la plus petite distance distinguable séparant deux objets dans l'image. La résolution des images TEP est faible. Cela est dû aux causes suivantes :

1. Le parcours du positon avant l'annihilation,
2. La dispersion des photons colinéaires émis,
3. La géométrie cylindrique du détecteur,
4. La taille et le nombre de détecteurs,
5. Le nombre de cristaux.

En plus de ces causes interviennent la taille de la source d'activité, l'échantillonnage et le filtrage appliqués durant la reconstruction.

### V.3.3 L'effet de volume partiel

L'effet de volume partiel est défini comme la perte d'activité apparente dans les petites régions du fait de l'échantillonnage et de la résolution spatiale limitée. En imagerie TEP, cet effet est important. À cause de la faible résolution des images TEP, le signal provenant d'un voxel en activité se répand sur les pixels présents dans un voisinage plus ou moins grand. Ainsi, certains voxels contiennent une portion du signal résultant d'une activité réelle à leur endroit, alors que d'autres voxels, qui sont théoriquement non actifs, contiennent une portion du signal. Cela se traduit par un certain nombre de voxels, présentant une activité plus grande que celle de la structure réelle en activité. De plus, en raison de l'échantillonnage spatial, un voxel peut contenir plusieurs tissus, qui n'ont pas le même taux d'activité. En conséquence, ces voxels mesurent l'apport d'activité de l'ensemble des tissus sous-jacents.

Les images PET ont une résolution spatiale d'environ 6 à 8 mm. Cela veut dire que les objets plus petits que 15 mm sont sous-évalués. Cela veut dire aussi que, en clinique, les petites activités à hauteur de petites lésions doivent être interprétées de manière plus sensible que des grosses lésions (plus grandes que 2 cm). C'est particulièrement important dans le cas des études de suivi thérapeutique. Une réduction de signal après un traitement peut être due en partie à une diminution de taille de la lésion et en partie à une réponse métabolique effective. Il est envisageable alors, si on connaît le volume des lésions ciblées, de compenser les effets de volume partiel et de s'approcher de l'activité réelle des objets étudiés via les coefficients de correction.

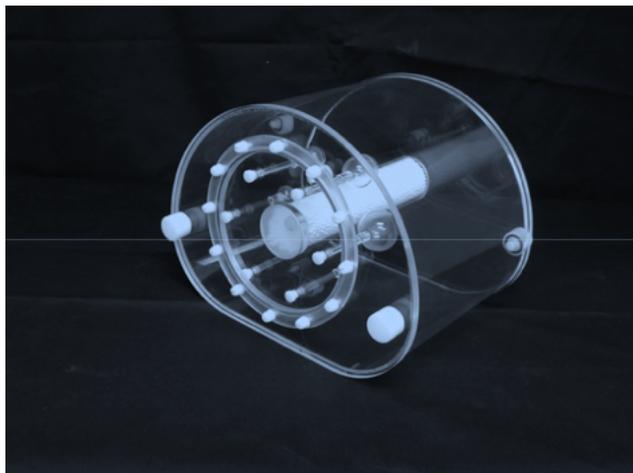
### V.4 Résultats expérimentaux

Dans cette section, nous présentons les résultats expérimentaux des différentes méthodes développées dans les chapitres précédents sur deux bases d'images TEP de test. La première base contient des images TEP obtenues à partir d'un fantôme NEMA IEC composé de 6 sphères. La deuxième base contient des images issues d'examens cliniques de patients réels.

À noter qu'il a fallu étendre les algorithmes proposés, conçus initialement pour segmenter des images 2D, à la segmentation des images 3D. Cette extension ne nécessite aucune modification des algorithmes et se fait naturellement, en ajoutant la 3<sup>ème</sup> dimension.

#### V.4.1 Résultats sur des images de fantômes

Pour ces tests, nous disposons d'une base d'images acquises à partir d'un fantôme NEMA IEC, qui est un fantôme cylindrique dont le diamètre et la hauteur interne sont d'environ 20 cm. Il contient 6 sphères de volumes différents (cf. figure V.5). La base utilisée contient 5 images de fantôme acquises avec différents paramètres de TEP présentant des contrastes différents entre les sphères et le fond. La figure V.6 montre un exemple de deux volumes différents. Ces volumes sont segmentés en deux classes.



**Figure V.5** – Fantôme NEMA IEC (source : *Data spectrum corporation*).

Trois exemples de résultats de segmentation sont montrés dans les figures V.7, V.8 et V.9. Pour des raisons de clarté, nous considérons la plus grande sphère des volumes comme la première sphère, la deuxième plus grande sphère comme la deuxième sphère, etc. On remarque



**Figure V.6** – Exemple d’images TEP acquises à partir du fantôme NEMA IEC à 6 sphères.

pour les trois exemples que les deux plus petites sphères sont difficilement détectables, car ces sphères, en plus de leurs petites tailles, sont les plus affectées par l’effet du volume partiel. De ce fait, elles sont considérées comme artéfacts par l’ensemble des méthodes exploitant le voisinage.

Sur le premier fantôme, on remarque que les deux méthodes IFCMS et MO\_IFCMS, en segmentant le volume en 2 classes, arrivent à extraire uniquement les sphères et considère le reste du volume comme du fond. Pour les autres méthodes, les deux régions segmentées sont le cylindre du fantôme, pour la première région, et le fond plus les sphères, pour la seconde. Les sphères peuvent être isolées facilement par un post-traitement. Cette différence de résultats est due au blocage des autres méthodes dans des minima locaux. De plus, on remarque que les résultats des méthodes IFCMS et MO\_IFCMS sont supérieurs à ceux des autres méthodes avec un léger avantage pour l’algorithme MO\_IFCMS.

Sur le deuxième et le troisième fantôme, en segmentant uniquement en 2 classes, les méthodes FCM, FCMS1 et FCMS2 n’arrivent pas à détecter les sphères et sont assimilées aux cylindres (seul le résultat de la FCM est présenté). Pour les autres méthodes, on remarque qu’elles arrivent à extraire les plus grands volumes. Cependant, la qualité des contours est meilleure pour la méthode MO\_IFCMS où ces contours sont plus réguliers.

### V.4.2 Résultats sur les examens cliniques

#### V.4.2.1 Données

Nous disposons pour cette base d'images de trois volumes d'examens cliniques de patients réels. Ces volumes nous ont été fournis par le CHU Henri-Mondor (Créteil). Sur ce type d'images, nous exploitons les algorithmes IFCMS, MO\_IFCMS, FCM et un seuillage à 85% utilisé habituellement pour ces images. Des segmentations en deux régions sont effectuées sur des régions d'intérêt délimitées manuellement en considérant les régions qui présentent des cellules cancéreuses. La figure V.10 montre, pour chacun des trois volumes, la région d'intérêt considérée (les régions montrées dans cette figure sont approximatives) pour la segmentation.

#### V.4.2.2 Problématique générale

Les images dont nous disposons concernent des patients atteints de lymphome (cancer du système lymphatique). Ce cancer se caractérise par la multiplication anarchique des lymphocytes (cellules du système immunitaire) qui s'accumulent pour former des tumeurs dans divers éléments du système lymphatique (les ganglions). Comme les cellules immunitaires circulent dans tout l'organisme, des organes peuvent être également touchés. Des tumeurs peuvent donc se développer dans des zones très variées de l'organisme.

Pour évaluer le pronostic lymphome chez un patient, il existe un grand nombre de critères. Parmi ces critères on peut citer : le IPI (*International Pronostic Index*), qui prend en compte cinq variables (l'âge du patient, des paramètres physiologiques, le stade de la maladie, etc.), et la valeur du SUV (*standardized uptake value*), qui est une mesure quantitative. Après un premier et un deuxième traitement, la différence de SUV entre les images TEP du premier traitement et les images TEP du deuxième traitement sont comparées pour donner un pronostic. Le critère général pour le SUV est que si sa valeur est supérieure à 2,5, la zone concernée est considérée comme étant une lésion. La diminution ou l'augmentation de cette valeur de SUV sera donc un indicateur pour le succès ou l'échec d'un traitement. L'inconvénient de cette méthode est qu'elle ne prend en compte qu'un voxel de la lésion. D'autres critères pertinents pourraient être pris en compte comme le nombre de lésions dans l'organisme, la masse de la lésion et sa texture.

Dans notre étude, les diagnostics des patients sont connus. L'objectif est d'évaluer, après traitement, le taux de rémission. Étant donné que de nombreux paramètres ne sont aujourd'hui

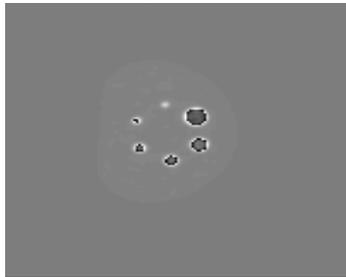
par pris en compte pour le pronostic, les médecins cherchent, donc, à évaluer si une autre méthode de pronostic prenant plus de paramètres en compte serait plus efficace que celle utilisée actuellement.

### V.4.2.3 Résultats

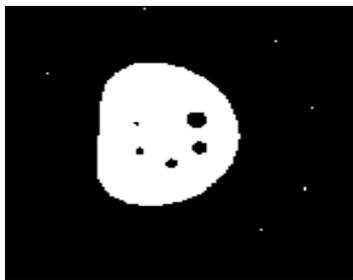
Les résultats de segmentation des volumes des patients 1, 2 et 3 sont présentés dans les figures [V.11](#), [V.12](#) et [V.13](#), respectivement. On remarque sur ces figures que la méthode de seuillage est la moins performante et présente beaucoup d'erreurs de classification. Les méthodes FCM, IFCMS et MO-IFCMS donnent de bons résultats de segmentation. La méthode MO-IFCMS a l'avantage de lisser les contours et présente des régions plus régulières reflétant ainsi la forme naturelle des tumeurs. Cela est dû principalement au critère contour intégré dans cette méthode.

## V.5 Conclusion

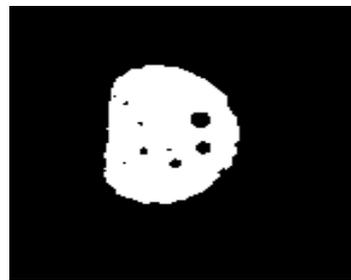
Dans ce chapitre, nous avons présenté une application des méthodes développées sur des images de tomographie par émission de positons. Nous avons d'abord présenté le principe général de cette technique d'imagerie ainsi que les principales caractéristiques et difficultés liées à la segmentation d'images acquises avec cette technique. Ensuite, nous avons appliqué les méthodes IFCMS et MO-IFCMS ainsi que d'autres méthodes de comparaison à deux types d'images : des images de fantômes et des images d'examen cliniques réels provenant du CHU Henri-Mondor (Créteil). À noter que les algorithmes développés ont été appliqués comme des procédures génériques, sans aucune adaptation spécifique aux images TEP. Les résultats obtenus montrent l'efficacité des méthodes proposées, avec un avantage de la méthode MO-IFCMS. Néanmoins, les résultats de segmentation peuvent davantage être améliorés, en exploitant les autres modes d'acquisition (CT, IRM, etc.). Ces développements sont en cours.



(a) Coupe du volume à segmenter



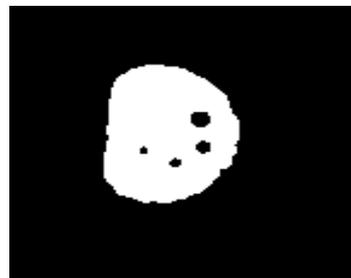
(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM



(d) Résultat de la méthode FMS-S1



(e) Résultat de la méthode FMS-S2

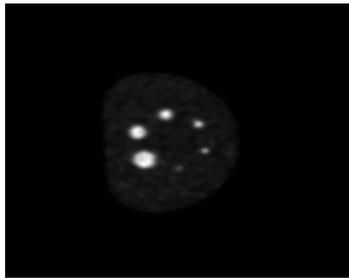


(f) Résultat de la méthode IFCMS

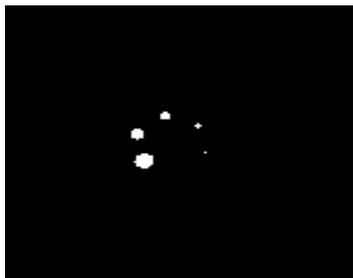


(g) Résultat de la méthode MO-IFCMS

**Figure V.7** – Résultats de segmentation du volume du fantôme 1.



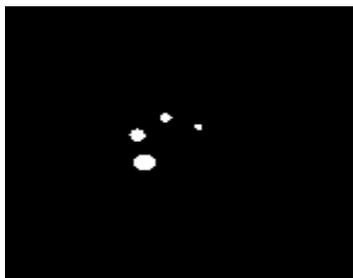
(a) Coupe du volume à segmenter



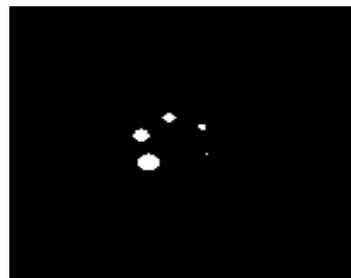
(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM

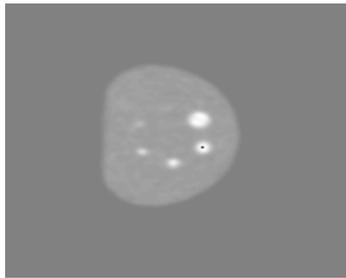


(d) Résultat de la méthode IFCMS

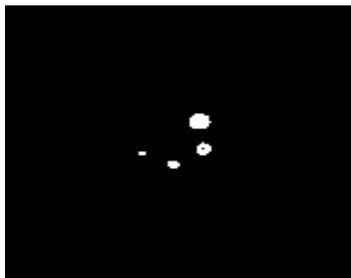


(e) Résultat de la méthode MO-IFCMS

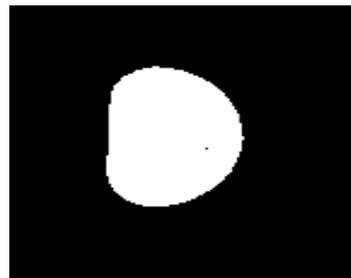
**Figure V.8** – Résultats de segmentation du volume du fantôme 2.



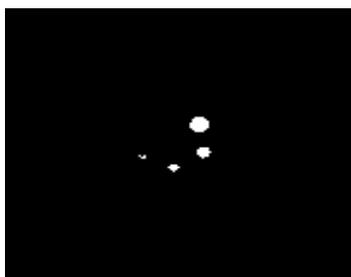
(a) Coupe du volume à segmenter



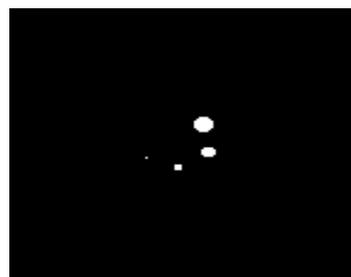
(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM

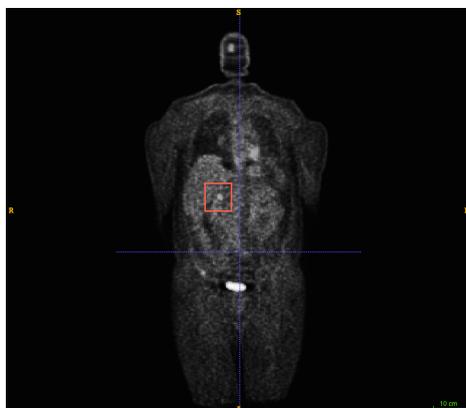


(d) Résultat de la méthode IFCMS

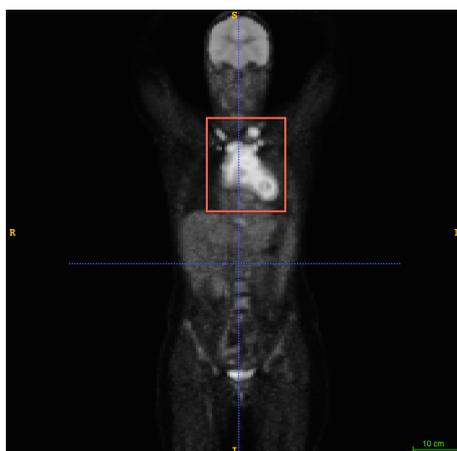


(e) Résultat de la méthode MO-IFCMS

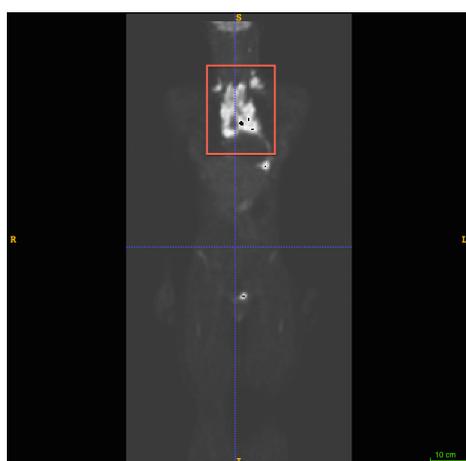
**Figure V.9** – Résultats de segmentation du volume du fantôme 3.



(a) Patient 1

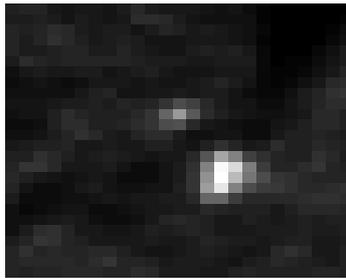


(b) Patient 2



(c) Patient 3

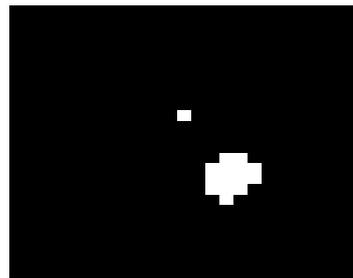
**Figure V.10** – Régions d'intérêt considérées pour la segmentation des trois volumes d'examens cliniques.



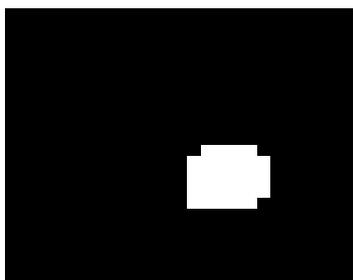
(a) Coupe du volume à segmenter



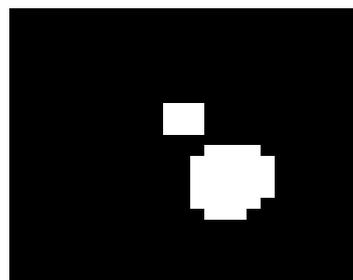
(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM

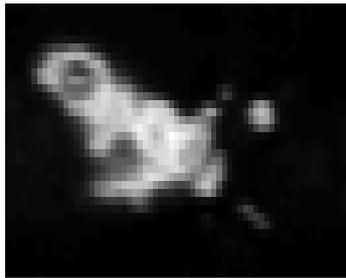


(d) Résultat de la méthode IFCMS



(e) Résultat de la méthode MO-IFCMS

**Figure V.11** – Résultats de segmentation des images du Patient 1.



(a) Coupe du volume à segmenter



(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM

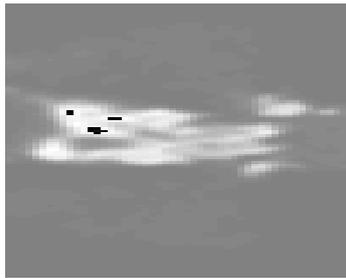


(d) Résultat de la méthode IFCMS

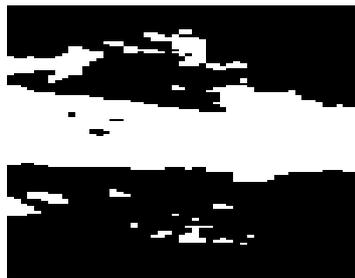


(e) Résultat de la méthode MO-IFCMS

**Figure V.12** – Résultats de segmentation des images du Patient 2.



(a) Coupe du volume à segmenter



(b) Résultat de la méthode de seuillage



(c) Résultat de la méthode FCM



(d) Résultat de la méthode IFCMS



(e) Résultat de la méthode MO-IFCMS

**Figure V.13** – Résultats de segmentation des images du Patient 3.

---

## Conclusion générale et perspectives

Cette thèse a eu comme objectif l'exploitation des métaheuristiques d'optimisation dans le problème de segmentation automatique d'images. Ce travail est stimulé par les constats suivants : d'une part, les méthodes de segmentation classiques sont basées principalement sur l'estimation de paramètres de modèles ou l'optimisation de critères. Confrontées à ces problèmes, elles ont souvent recours à des hypothèses simplificatrices pour les surmonter. D'autre part, les métaheuristiques d'optimisation sont un excellent outil permettant de résoudre n'importe quel problème d'optimisation, sans connaissance approfondie de ce dernier. De ce fait, nous nous sommes intéressés dans notre travail à l'amélioration de la méthode de segmentation par logique floue. De notre étude trois méthodes ont vu le jour : une approche mono objectif, une approche multiobjectif et finalement une méthode de post-segmentation.

La première approche proposée porte sur l'amélioration de la méthode de segmentation d'images par l'algorithme des C-moyennes floues (FCM). Ce dernier est largement utilisé à cause de son efficacité qui est due principalement à son caractère flou. En effet, à l'inverse des méthodes de classification dites dures (*hard clustering*), l'algorithme FCM à l'avantage d'attribuer des degrés d'appartenance, ce qui permet de réduire la propagation d'erreurs, de mieux traiter les bruits et de tenir compte intrinsèquement des caractéristiques physiques des images (ex. : effet de volumes partiels). Néanmoins, cette méthode présente les inconvénients suivants : forte dépendance aux solutions initiales conduisant au blocage dans des minima locaux, utilisation de la distance Euclidienne supposant que les classes ont des formes géométriques hyper-sphériques à diamètres égaux, et finalement, absence de prise en compte de l'information spatiale, ce qui revient à négliger la corrélation spatiale entre les pixels. Afin de pallier ces défauts, nous avons développé une méthode utilisant la métaheuristique d'optimisation par essaim particulière, pour s'affranchir du problème d'initialisation, et nous avons mis au point un nouveau critère qui exploite la distance de Mahalanobis et qui tient compte de l'information

spatiale pour remédier aux deux autres inconvénients. De plus, nous avons ajouté un étage de post-segmentation afin de raffiner le résultat de segmentation.

La deuxième méthode développée est une extension de la première méthode. Ayant constaté que la majorité des erreurs de segmentation générées par la première méthode sont localisées au niveau des frontières entre les régions, nous avons introduit, en plus du premier critère, qui est basé région, un critère complémentaire basé contour. Ces deux critères sont optimisés simultanément dans une approche multiobjectif, à l'aide de la métaheuristique d'optimisation par essaim particulière, dans sa version multiobjectif. Le résultat de cette opération est un ensemble de solutions dont aucune n'est meilleure qu'une autre pour les deux critères, nous avons conçu une nouvelle méthode de combinaison, qui permet de tirer avantage de chacune de ces solutions, au lieu d'en choisir une seule.

La dernière méthode proposée est un algorithme de post-segmentation utilisable avec n'importe quelle méthode de segmentation. Cet algorithme vise à corriger les erreurs potentielles de segmentation. À partir d'un résultat de segmentation, en premier lieu, les contours sont extraits. Chaque contour est considéré comme un chemin qu'il faut modifier pour trouver le parcours optimal. Ce problème pouvant être formulé comme un problème de voyageur de commerce, nous avons opté pour la méthode de colonie de fourmis, réputée pour résoudre efficacement ce type de problème. Les fourmis se déplacent le long des contours en exploitant des informations locales et la cohérence globale du contour, afin de trouver le chemin optimal.

Les méthodes proposées ont été testées et comparées à d'autres méthodes de la littérature, en premier lieu sur des bases de données comportant : des images synthétiques contenant plusieurs classes et affectées de plusieurs types et taux de bruit, des images IRM simulées contenant plusieurs taux de bruit et paramètres d'inhomogénéité et des images IRM réelles. Puis nos méthodes ont été exploitées sur des images de tomographie par émission de positons. Les résultats obtenus montrent le potentiel des méthodes développées.

A partir de ce travail, de nombreuses perspectives se peuvent être formulées. En effet, les algorithmes développés peuvent être considérés comme des procédures générales de segmentation qu'on peut utiliser en modifiant les critères, les attributs ou les méthodes d'optimisation, selon l'application visée. Ainsi, en exploitant diverses combinaisons, plusieurs algorithmes peuvent être mis au point.

Il est aussi intéressant d'exploiter les idées suggérées dans les problèmes de segmentation multimodale.

L'exploration de la post-segmentation nous paraît une piste de développement prometteuse, en considérant le point de vue suivant : la plupart des méthodes de segmentation sont à même de segmenter correctement la majorité des pixels qui se situent dans des zones homogènes. De ce fait, au lieu de concevoir des méthodes complexes dès le départ, on peut envisager d'utiliser n'importe quelle méthode de segmentation relativement simple et efficace et se focaliser sur la post-segmentation, qui permet de détecter et reclasser les pixels potentiellement mal classés. En effet, la première segmentation nous fournit de l'information supplémentaire, qui peut être exploitée dans le deuxième étage. Un exemple simple serait de faire l'apprentissage d'un classifieur sur les pixels de confiance et de reclasser par la suite les autres pixels.

Enfin, il serait aussi intéressant d'étendre les méthodes développées à la segmentation de séquences d'images, qui se ramène à un problème d'optimisation dynamique.

Au-delà du problème de segmentation d'image, nous pensons que les méthodes développées peuvent être adaptées aux problèmes de classification de manière générale.

---

## Références bibliographiques

- [Bra, 2013] Brainweb : Simulated brain database, 2013. URL <http://brainweb.bic.mni.mcgill.ca/brainweb/>.
- [IBS, 2013] IBSR : Internet brain segmentation repository, 2013. URL <http://www.cma.mgh.harvard.edu/ibsr/>.
- [Ahmed et al., 2002] Mohamed N Ahmed, Sameh M Yamany, Nevin Mohamed, Aly A Farag, & Thomas Moriarty. A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data. *Medical Imaging, IEEE Transactions on*, 21(3): 193–199, 2002.
- [Alihodzic & Tuba, 2014] Adis Alihodzic & Milan Tuba. Improved Bat algorithm applied to multilevel image thresholding. *The Scientific World Journal*, 2014, 2014.
- [Asl & Seyedin, 2006] Mahdi Ahmadi Asl & Seyed Alireza Seyedin. Active contour optimization using particle swarm optimizer. In *Information and Communication Technologies, 2006. ICTTA '06. 2nd*, vol. 1, pp. 1522–1523. IEEE, 2006.
- [Bandyopadhyay et al., 2008] Sanghamitra Bandyopadhyay, Sriparna Saha, Ujjwal Maulik, & Kalyanmoy Deb. A simulated annealing-based multiobjective optimization algorithm : AMOSA. *Evolutionary Computation, IEEE Transactions on*, 12(3): 269–283, 2008.
- [Beckers et al., 1992] Ralph Beckers, Jean-Louis Deneubourg, & Simon Goss. Trails and U-turns in the selection of a path by the ant *Lasius Niger*. *Journal of theoretical biology*, 159(4): 397–415, 1992.
- [Bezdek, 1981] James C Bezdek. *Pattern recognition with fuzzy objective function algorithms*.

- Kluwer Academic Publishers, 1981.
- [Bezdek et al., 1992] James C Bezdek, LO Hall, & LP Clarke. Review of MR image segmentation techniques using pattern recognition. *Medical physics*, 20(4): 1033–1048, 1992.
- [Bhanu et al., 1995] Bir Bhanu, Sungkee Lee, & John Ming. Adaptive image segmentation using a genetic algorithm. *Systems, Man and Cybernetics, IEEE Transactions on*, 25(12): 1543–1567, 1995.
- [Bonabeau et al., 1999] Eric Bonabeau, Marco Dorigo, & Guy Theraulaz. *Swarm intelligence : from natural to artificial systems*. Number 1. Oxford university press, 1999.
- [Boussaïd et al., 2013] Ilhem Boussaïd, Amitava Chatterjee, Patrick Siarry, & Mohamed Ahmed-Nacer. Hybrid BBO-DE algorithms for fuzzy entropy-based thresholding. In *Computational Intelligence in Image Processing*, pp. 37–69. Springer, 2013.
- [Brunel, 2005] Nicolas Brunel. *Sur quelques extensions des chaînes de Markov cachées et couples. Applications à la segmentation non-supervisée de signaux radar*. PhD thesis, Université Pierre et Marie Curie (Paris VI), 2005.
- [Cai et al., 2007] Welling Cai, Songcan Chen, & Daoqiang Zhang. Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. *Pattern recognition*, 40(3): 825–838, 2007.
- [Caselles et al., 1997] Vicent Caselles, Ron Kimmel, & Guillermo Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1): 61–79, 1997.
- [Castellani et al., 2007] Umberto Castellani, Andrea Fusiello, Riccardo Gherardi, & Vittorio Murino. Automatic selection of MRF control parameters by reactive tabu search. *Image and Vision Computing*, 25(11): 1824–1832, 2007.
- [Černý, 1985] Vladimír Černý. Thermodynamical approach to the traveling salesman problem : An efficient simulation algorithm. *Journal of optimization theory and applications*, 45(1): 41–51, 1985.
- [Charles, 1859] Darwin Charles. *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life*. Murray, London, 1859.
- [Charnes & Cooper, 1977] Abraham Charnes & William Wager Cooper. Goal programming and multiple objective optimizations : Part 1. *European Journal of Operational Research*, 1(1): 39–54, 1977.
- [Chen et al., 2010] Chi-hau Chen, Louis-François Pau, & Patrick Shen-pei Wang. *Handbook of pattern recognition and computer vision*. World Scientific, 2010.

- [Chen & Zhang, 2004] Songcan Chen & Daoqiang Zhang. Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure. *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on*, 34(4): 1907–1916, 2004.
- [Chuang et al., 2006] Keh-Shih Chuang, Hong-Long Tzeng, Sharon Chen, Jay Wu, & Tzong-Jer Chen. Fuzzy c-means clustering with spatial information for image segmentation. *computerized medical imaging and graphics*, 30(1): 9–15, 2006.
- [Clerc, 2010] Maurice Clerc. *Particle swarm optimization*, vol. 93. John Wiley & Sons, 2010.
- [Coello, 1999] Carlos A Coello Coello. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information systems*, 1(3): 269–308, 1999.
- [Coello et al., 2004] Carlos A Coello Coello, Gregorio Toscano Pulido, & M Salazar Lechuga. Handling multiple objectives with particle swarm optimization. *Evolutionary Computation, IEEE Transactions on*, 8(3): 256–279, 2004.
- [Coello Coello et al., 1995] Carlos A Coello Coello, Alan D Christiansen, & Arturo Hernández Aguirre. Multiobjective design optimization of counterweight balancing of a robot arm using genetic algorithms. In *Tools with Artificial Intelligence, 1995. Proceedings., Seventh International Conference on*, pp. 20–23. IEEE, 1995.
- [Cohen & Cohen, 1993] Laurent D Cohen & Isaac Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(11): 1131–1147, 1993.
- [Collette & Siarry, 2003] Yann Collette & Patrick Siarry. *Multiobjective Optimization : Principles and Case Studies*. Springer, 2003.
- [Cremers et al., 2007] Daniel Cremers, Mikael Rousson, & Rachid Deriche. A review of statistical approaches to level set segmentation : integrating color, texture, motion and shape. *International journal of computer vision*, 72(2): 195–215, 2007.
- [Das et al., 2009] Swagatam Das, Ajith Abraham, & Amit Konar. *Metaheuristic clustering*, vol. 178. Springer, 2009.
- [Deb et al., 2002] K. Deb, A. Pratap, S. Agarwal, & T. Meyarivan. A fast and elitist multiobjective genetic algorithm : NSGA-II. *Evolutionary Computation, IEEE Transactions on*, 6(2): 182–197, 2002.

- [Deb, 2001] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*, vol. 16. John Wiley & Sons, 2001.
- [Deneubourg et al., 1983] Jean-Louis Deneubourg, Jacques M Pasteels, & Jean-Claude Verhaeghe. Probabilistic behaviour in ants : a strategy of errors? *Journal of Theoretical Biology*, 105(2): 259–271, 1983.
- [Dorigo, 1992] Marco Dorigo. Optimization, learning and natural algorithms. *Ph. D. Thesis, Politecnico di Milano, Italy*, 1992.
- [Dorigo & Gambardella, 1997] Marco Dorigo & Luca Maria Gambardella. Ant colony system : a cooperative learning approach to the traveling salesman problem. *Evolutionary Computation, IEEE Transactions on*, 1(1): 53–66, 1997.
- [Dorigo et al., 1996] Marco Dorigo, Vittorio Maniezzo, & Alberto Coloni. Ant system : optimization by a colony of cooperating agents. *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on*, 26(1): 29–41, 1996.
- [Dréo et al., 2003] Johann Dréo, Alain Petrowski, Éric D Taillard, & Patrick Siarry. *Métaheuristiques pour l'optimisation difficile*. Eyrolles (Editions), 2003.
- [Edgeworth, 1885] Francis Ysidro Edgeworth. Methods of statistics. *Journal of the Statistical Society of London*, pp. 181–217, 1885.
- [Eigen, 1973] Manfred Eigen. *Ingo Rechenberg Evolutionsstrategie Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. mit einem Nachwort von Manfred Eigen, Friedrich Frommann Verlag, Struttgart-Bad Cannstatt, 1973.
- [El Dor et al., 2014] Abbas El Dor, Julien Lepagnot, Amir Nakib, & Patrick Siarry. PSO-2S optimization algorithm for brain MRI segmentation. In *Genetic and Evolutionary Computing*, pp. 13–22. Springer, 2014.
- [Equihua, 1990] Miguel Equihua. Fuzzy clustering of ecological data. *The Journal of Ecology*, pp. 519–534, 1990.
- [Feng et al., 2005] Du Feng, Shi Wenkang, Chen Liangzhou, Deng Yong, & Zhu Zhenfu. Infrared image segmentation with 2-D maximum entropy method based on particle swarm optimization (PSO). *Pattern Recognition Letters*, 26(5): 597–603, 2005.
- [Fogel et al., 1966] Lawrence J Fogel, Alvin J Owens, & Michael J Walsh. Artificial intelligence through simulated evolution. 1966.
- [Fonseca et al., 1993] Carlos M Fonseca, Peter J Fleming, et al. Genetic Algorithms for Multiob-

- jective Optimization : Formulation, Discussion and Generalization. In *ICGA*, vol. 93, pp. 416–423, 1993.
- [Friedman & Kandel, 1999] Menahem Friedman & Abraham Kandel. *Introduction to pattern recognition : statistical, structural, neural, and fuzzy logic approaches*, vol. 32. World scientific Singapore, 1999.
- [Fritz et al., 2013] Heinrich Fritz, Luis A. García-Escudero, & Agustín Mayo-Iscar. Robust constrained fuzzy clustering. *Information Sciences*, 245(0): 38–52, 2013.
- [Gambardella & Dorigo, 1996] Luca Maria Gambardella & Marco Dorigo. Solving symmetric and asymmetric TSPs by ant colonies. In *International conference on evolutionary computation*, pp. 622–627, 1996.
- [Gath et al., 1983] I Gath, D Lehmann, & E Bar-On. Fuzzy clustering of EEG signal and vigilance performance. *International journal of neuroscience*, 20(3-4): 303–312, 1983.
- [Gath & Geva, 1989] Isak Gath & Amir B. Geva. Unsupervised optimal fuzzy clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(7): 773–780, 1989.
- [Geman & Geman, 1984] Stuart Geman & Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6): 721–741, 1984.
- [Glover, 1986] Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5): 533–549, 1986.
- [Glover, 1989] Fred Glover. Tabu search-part I. *ORSA Journal on computing*, 1(3): 190–206, 1989.
- [Glover, 1990] Fred Glover. Tabu search-part II. *ORSA Journal on computing*, 2(1): 4–32, 1990.
- [Goldberg, 1989] David E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Number 2. Addison-Wesley, Reading, MA, 1989.
- [Goss et al., 1989] Simon Goss, Serge Aron, Jean-Louis Deneubourg, & Jacques Marie Pasteels. Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, 76(12): 579–581, 1989.
- [Granath, 1984] Göran Granath. Application of fuzzy clustering and fuzzy classification to evaluate the provenance of glacial till. *Journal of the International Associa-*

- 
- tion for Mathematical Geology*, 16(3): 283–301, 1984.
- [Graves & Pedrycz, 2010] Daniel Graves & Witold Pedrycz. Kernel-based fuzzy clustering and fuzzy clustering : A comparative experimental study. *Fuzzy sets and systems*, 161(4): 522–543, 2010.
- [Gustafson & Kessel, 1978] Donald E Gustafson & William C Kessel. Fuzzy clustering with a fuzzy covariance matrix. In *Decision and Control including the 17th Symposium on Adaptive Processes, 1978 IEEE Conference on*, vol. 17, pp. 761–766. IEEE, 1978.
- [Handl & Knowles, 2005] Julia Handl & Joshua Knowles. Exploiting the trade-off—the benefits of multiple objectives in data clustering. In *Evolutionary Multi-Criterion Optimization*, pp. 547–560. Springer, 2005.
- [Haralick & Shapiro, 1985] Robert M Haralick & Linda G Shapiro. Image segmentation techniques. In *1985 Technical Symposium East*, pp. 2–9. International Society for Optics and Photonics, 1985.
- [Hegarat-Masclé et al., 2007] Le Hegarat-Masclé, Abdelaziz Kallel, Xavier Descombes, et al. Ant colony optimization for image regularization based on a nonstationary Markov modeling. *Image Processing, IEEE Transactions on*, 16(3): 865–878, 2007.
- [Held et al., 1997] Karsten Held, E Rota Kops, Bernd J Krause, WMIII Wells, Ron Kikinis, & H-W Muller-Gartner. Markov random field segmentation of brain MR images. *Medical Imaging, IEEE Transactions on*, 16(6): 878–886, 1997.
- [Heppner & Grenander, 1990] Frank Heppner & U Grenander. A stochastic nonlinear model for coordinated bird flocks. In *The ubiquity of chaos*, pp. 233–238. AAAS Publications, 1990.
- [Holland, 1975] John H Holland. *Adaptation in natural and artificial systems : An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, 1975.
- [Hongfu, 2002] Chen Guo Zuo Hongfu. 2-D maximum entropy method of image segmentation based on genetic algorithm. *Journal of Computer Aided Design & Computer Graphics*, 6: 008, 2002.
- [Horng, 2011] Ming-Huwi Horng. Multilevel thresholding selection based on the artificial bee colony algorithm for image segmentation. *Expert Systems with Applications*, 38(11): 13785–13791, 2011.
-

- [Huang et al., 2012] Hsin-Chien Huang, Yung-Yu Chuang, & Chu-Song Chen. Multiple kernel fuzzy clustering. *Fuzzy Systems, IEEE Transactions on*, 20(1): 120–134, 2012.
- [Huntsberger et al., 1985] TL Huntsberger, CL Jacobs, & Robert L Cannon. Iterative fuzzy image segmentation. *Pattern recognition*, 18(2): 131–138, 1985.
- [Ishibuchi & Murata, 1998] Hisao Ishibuchi & Tadahiko Murata. A multi-objective genetic local search algorithm and its application to flowshop scheduling. *Systems, Man, and Cybernetics, Part C : Applications and Reviews, IEEE Transactions on*, 28(3): 392–403, 1998.
- [Izakian & Abraham, 2011] Hesam Izakian & Ajith Abraham. Fuzzy c-means and fuzzy swarm for fuzzy clustering problem. *Expert Systems with Applications*, 38(3): 1835–1838, 2011.
- [Ji et al., 2014] Zexuan Ji, Guo Cao, & Quansen Sun. A fuzzy clustering algorithm with robust spatially constraint for brain MR image segmentation. In *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*, pp. 202–209. IEEE, 2014.
- [JoonKim et al., 1998] Hang JoonKim, Eun YiKim, Jin WookKim, & Se HyunPark. MRF model based image segmentation using hierarchical distributed genetic algorithm. *Electronics Letters*, 34(25): 2394–2395, 1998.
- [Kang & Zhang, 2009] Jiayin Kang & Wenjuan Zhang. Fingerprint image segmentation using modified fuzzy c-means algorithm. In *Bioinformatics and Biomedical Engineering, 2009. ICBBE 2009. 3rd International Conference on*, pp. 1–4. IEEE, 2009.
- [Kanungo et al., 2002] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, & Angela Y Wu. An efficient k-means clustering algorithm : Analysis and implementation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7): 881–892, 2002.
- [Kapur et al., 1985] JN Kapur, Prasanna K Sahoo, & Andrew KC 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.
- [Karasulu & Korukoglu, 2011] Bahadir Karasulu & Serdar Korukoglu. A simulated annealing-based optimal threshold determining method in edge-based segmentation of grayscale images. *Applied Soft Computing*, 11(2): 2246–2259, 2011.
- [Kass et al., 1988] Michael Kass, Andrew Witkin, & Demetri Terzopoulos. Snakes : Active contour models. *International journal of computer vision*, 1(4): 321–331,

- 1988.
- [Kennedy & Eberhart, 1995] J Kennedy & R Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, pp. 1942–1948. IEEE, 1995.
- [Khodja, 1997] Lotfi Khodja. *Contribution à la classification floue non supervisée*. PhD thesis, Université de Savoie, 1997.
- [Kim et al., 2000] EY Kim, SH Park, & HJ Kim. A genetic algorithm-based segmentation of Markov random field modeled images. *Signal Processing Letters, IEEE*, 7 (11): 301–303, 2000.
- [Kirkpatrick et al., 1983] Scott Kirkpatrick, C Daniel Gelatt, Mario P Vecchi, et al. Optimization by simulated annealing. *Science*, 220(4598): 671–680, 1983.
- [Koza, 1992] John R Koza. *Genetic programming : on the programming of computers by means of natural selection*, vol. 1. MIT press, 1992.
- [Krinidis & Chatzis, 2010] Stelios Krinidis & Vassilios Chatzis. A robust fuzzy local information c-means clustering algorithm. *Image Processing, IEEE Transactions on*, 19(5): 1328–1337, 2010.
- [Krishnapuram & Keller, 1993] Raghuram Krishnapuram & James M Keller. A possibilistic approach to clustering. *Fuzzy Systems, IEEE Transactions on*, 1(2): 98–110, 1993.
- [Lai & Tseng, 2004] Chih-Chin Lai & Din-Chang Tseng. A hybrid approach using Gaussian smoothing and genetic algorithm for multilevel thresholding. *International Journal of Hybrid Intelligent Systems*, 1(3): 143–152, 2004.
- [Lakshmanan & Derin, 1989] Sridhar Lakshmanan & Haluk Derin. Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(8): 799–813, 1989.
- [Lepagnot et al., 2011] Julien Lepagnot, Amir Nakib, Hamouche Oulhadj, & Patrick Siarry. Brain cine MRI segmentation based on a multiagent algorithm for dynamic continuous optimization. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pp. 1695–1702. IEEE, 2011.
- [Liew & Yan, 2003] AW-C Liew & Hong Yan. An adaptive spatial fuzzy clustering algorithm for 3-D MR image segmentation. *Medical Imaging, IEEE Transactions on*, 22

- (9): 1063–1075, 2003.
- [Liu et al., 2014] Cong Liu, Aimin Zhou, Qian Zhang, & Guixu Zhang. Adaptive image segmentation by using mean-shift and evolutionary optimisation. *Image Processing, IET*, 8(6): 327–333, 2014.
- [Lourenço et al., 2001] Helena R Lourenço, Olivier C Martin, & Thomas Stutzle. Iterated local search. *arXiv preprint math/0102188*, 2001.
- [Lu & Zhou, 2007] Xiaodong Lu & Jun Zhou. Image segmentation based on Markov random field with ant colony system. In *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pp. 1793–1797. IEEE, 2007.
- [MacEachern & Manku, 1998] Leonard A MacEachern & Tajinder Manku. Genetic algorithms for active contour optimization. In *Circuits and Systems, 1998. ISCAS'98. Proceedings of the 1998 IEEE International Symposium on*, vol. 4, pp. 229–232. IEEE, 1998.
- [Maitra & Chatterjee, 2008] Madhubanti Maitra & Amitava Chatterjee. A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding. *Expert Systems with Applications*, 34(2): 1341–1350, 2008.
- [Makropoulos et al., 2012] Antonios Makropoulos, Christian Ledig, Paul Aljabar, Ahmed Serag, Joseph V Hajnal, A David Edwards, Serena J Counsell, & Daniel Rueckert. Automatic tissue and structural segmentation of neonatal brain MRI using Expectation-Maximization. *MICCAI Grand Challenge : Neonatal Brain Segmentation*, 9: 2012, 2012.
- [Malisia & Tizhoosh, 2006] Alice R Malisia & Hamid R Tizhoosh. Image thresholding using ant colony optimization. In *Computer and Robot Vision, 2006. The 3rd Canadian Conference on*, pp. 26–26. IEEE, 2006.
- [Manousakas et al., 1998] IN Manousakas, PE Undrill, GG Cameron, & TW 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.
- [Maulik & Bandyopadhyay, 2000] Ujjwal Maulik & Sanghamitra Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern recognition*, 33(9): 1455–1465, 2000.

- [McInerney & Terzopoulos, 1996] Tim McInerney & Demetri Terzopoulos. Deformable models in medical image analysis : a survey. *Medical image analysis*, 1(2): 91–108, 1996.
- [McInerney & Terzopoulos, 2000] Tim McInerney & Demetri Terzopoulos. T-snakes : Topology adaptive snakes. *Medical image analysis*, 4(2): 73–91, 2000.
- [Merheb, 2007] Charbel Merheb. *Optimisation de la reconstruction complète 3D en tomographie par émission de positons du petit animal par modélisation monte-carlo de la matrice système*. PhD thesis, Université Paris Sud, 2007.
- [Metropolis et al., 1953] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, & Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6): 1087–1092, 1953.
- [Monmarché, 2000] Nicolas Monmarché. *Algorithmes de fourmis artificielles : applications à la classification et à l'optimisation*. PhD thesis, École Polytechnique Tours, 2000.
- [Nakib, 2008] Amir Nakib. *Conception de métaheuristiques d'optimisation pour la segmentation d'images. Application à des images biomédicales*. PhD thesis, Université de Paris 12, 2008.
- [Nakib et al., 2007] Amir Nakib, Hamouche Oulhadj, & Patrick Siarry. Image histogram thresholding based on multiobjective optimization. *Signal Processing*, 87(11): 2516–2534, 2007.
- [Nakib et al., 2008] Amir Nakib, Hamouche Oulhadj, & Patrick Siarry. Non-supervised image segmentation based on multiobjective optimization. *Pattern Recognition Letters*, 29(2): 161–172, 2008.
- [Ng & Wong, 2002] Michael K Ng & Joyce C Wong. Clustering categorical data sets using tabu search techniques. *Pattern Recognition*, 35(12): 2783–2790, 2002.
- [Niknam & Amiri, 2010] Taher Niknam & Babak Amiri. An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. *Applied Soft Computing*, 10(1): 183–197, 2010.
- [Niknam et al., 2009] Taher Niknam, Babak Amiri, Javad Olamaei, & Ali Arefi. An efficient hybrid evolutionary optimization algorithm based on PSO and SA for clustering. *Journal of Zhejiang University Science A*, 10(4): 512–519, 2009.
- [Otsu, 1975] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296): 23–27, 1975.

- [Ouadfel & Batouche, 2003] Salima Ouadfel & Mohamed Batouche. Ant colony system with local search for Markov random field image segmentation. In *Image Processing, 2003. ICIIP 2003. Proceedings. 2003 International Conference on*, vol. 1, pp. I–133. IEEE, 2003.
- [Pareto, 1896] Vilfredo Pareto. *Cours d'Economie Politique*. Droz, Genève, 1896.
- [Parsopoulos et al., 2004] Konstantinos E Parsopoulos, Dimitris K Tasoulis, Michael N Vrahatis, et al. Multiobjective optimization using parallel vector evaluated particle swarm optimization. In *Proceedings of the IASTED international conference on artificial intelligence and applications (AIA 2004)*, vol. 2, pp. 823–828, 2004.
- [Pham, 2002] Dzung L Pham. Fuzzy clustering with spatial constraints. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 2, pp. II–65. IEEE, 2002.
- [Pham & Prince, 1999] Dzung L Pham & Jerry L Prince. An adaptive fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities. *Pattern Recognition Letters*, 20(1): 57–68, 1999.
- [Pham et al., 2000] Dzung L Pham, Chenyang Xu, & Jerry L Prince. Current methods in medical image segmentation 1. *Annual review of biomedical engineering*, 2(1): 315–337, 2000.
- [Reynolds, 1987] Craig W Reynolds. Flocks, herds and schools : A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4): 25–34, 1987.
- [Ritzel et al., 1994] Brian J Ritzel, J Wayland Eheart, & S Ranjithan. Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. *Water Resources Research*, 30(5): 1589–1603, 1994.
- [Sathya & Kayalvizhi, 2010] PD Sathya & R Kayalvizhi. PSO-based Tsallis thresholding selection procedure for image segmentation. *International Journal of Computer Applications*, 5(4): 39–46, 2010.
- [Schaffer, 1985] J David Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the 1st international Conference on Genetic Algorithms*, pp. 93–100. L. Erlbaum Associates Inc., 1985.
- [Sethian, 1996] James A Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4): 1591–1595, 1996.

- [Sezgin & Sankur, 2004] Mehmet Sezgin & Bulent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13: 146–168, 2004.
- [Siarry (Ed.), 2014] Patrick Siarry (Ed.). *Métaheuristiques : Recuit simulé, recherche avec tabous, recherche à voisinages variables, méthodes GRASP, algorithmes évolutionnaires, fourmis artificielles, essais particuliers et autres méthodes d’optimisation*. Editions Eyrolles, 2014.
- [Snyder et al., 1990] Wesley Snyder, Griff Bilbro, Ambalavaner Logenthiran, & Sarah Rajala. Optimal thresholding—a new approach. *Pattern Recognition Letters*, 11(12): 803–809, 1990.
- [Srinivas & Deb, 1994] Nidamarthi Srinivas & Kalyanmoy Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2(3): 221–248, 1994.
- [Storvik, 1994] Geir Storvik. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(10): 976–986, 1994.
- [Szilagyí et al., 2003] László Szilagyí, Zoltán Benyo, Sándor M Szilagyí, & HS Adam. MR brain image segmentation using an enhanced fuzzy c-means algorithm. In *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE*, vol. 1, pp. 724–726. IEEE, 2003.
- [Talbi, 2009] El-Ghazali Talbi. *Metaheuristics : from design to implementation*, vol. 74. John Wiley & Sons, 2009.
- [Tang et al., 2007] Liqun Tang, Kejun Wang, Guangsheng Feng, & Yonghua Li. An image segmentation algorithm based on the simulated annealing and improved snake model. In *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*, pp. 3876–3881. IEEE, 2007.
- [Tao et al., 2003] Wen-Bing Tao, Jin-Wen Tian, & Jian Liu. Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm. *Pattern Recognition Letters*, 24(16): 3069–3078, 2003.
- [Toennies, 2012] Klaus D Toennies. *Guide to Medical Image Analysis : Methods and Algorithms*. London : Springer London, 2012.
- [Tolias & Panas, 1998] Yannis A Tolias & Stavros M Panas. Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions.

- Systems, Man and Cybernetics, Part A : Systems and Humans, IEEE Transactions on*, 28(3): 359–369, 1998.
- [Tsai et al., 2014] Chun-Wei Tsai, Wei-Cheng Huang, & Ming-Chao Chiang. Recent development of metaheuristics for clustering. In *Mobile, Ubiquitous, and Intelligent Computing*, pp. 629–636. Springer, 2014.
- [Tseng et al., 2009] Chun-Chieh Tseng, Jer-Guang Hsieh, & Jyh-Horng Jeng. Active contour model via multi-population particle swarm optimization. *Expert Systems with Applications*, 36(3): 5348–5352, 2009.
- [Tseng & Lai, 1999] Din-Chang Tseng & Chih-Ching Lai. A genetic algorithm for MRF-based segmentation of multi-spectral textured images. *Pattern Recognition Letters*, 20(14): 1499–1510, 1999.
- [Wang et al., 2006] Xiao-Nian Wang, Yuan-Jing Feng, & Zu-Ren Feng. Ant colony optimization with active contour models for image segmentation. *Control Theory & Applications*, 4: 004, 2006.
- [Wu et al., 2003] Zhong-dong Wu, Wei-xin Xie, & Jian-ping Yu. Fuzzy c-means clustering algorithm based on kernel method. In *Computational Intelligence and Multimedia Applications, 2003. ICCIMA 2003. Proceedings. Fifth International Conference on*, pp. 49–54. IEEE, 2003.
- [Yin, 1999] Peng-Yeng Yin. A fast scheme for optimal thresholding using genetic algorithms. *Signal Processing*, 72(2): 85–95, 1999.
- [Yin, 2000] Peng-Yeng Yin. A tabu search approach to polygonal approximation of digital curves. *International Journal of Pattern Recognition and Artificial Intelligence*, 14(02): 243–255, 2000.
- [Yin, 2007] Peng-Yeng Yin. Multilevel minimum cross entropy threshold selection based on particle swarm optimization. *Applied Mathematics and Computation*, 184(2): 503–513, 2007.
- [Zadeh, 1965] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3): 338–353, 1965.
- [Zhang & Chen, 2002] Daoqiang Zhang & Songcan Chen. Fuzzy clustering using kernel method. In *The 2002 International Conference on Control and Automation, 2002. ICCA, 2002*. Citeseer, 2002.
- [Zhang et al., 2008] Liangpei Zhang, Yindi Zhao, Bo Huang, & Pingxiang Li. Texture feature fusion with neighborhood oscillating tabu search for high resolution image classification. *Photogrammetric Engineering & Remote Sensing*, 74(3):

323–331, 2008.

- [Zhang, 1996] Yu Jin Zhang. A survey on evaluation methods for image segmentation. *Pattern recognition*, 29(8): 1335–1346, 1996.
- [Zhang, 2006] Yu-Jin Zhang. *Advances in image and video segmentation*. IGI Global, 2006.
- [Zhao et al., 2008] Xin Zhao, Myung-Eun Lee, & Soo-Hyung Kim. Improved image thresholding using ant colony optimization algorithm. In *Advanced Language Processing and Web Information Technology, 2008. ALPIT'08. International Conference on*, pp. 210–215. IEEE, 2008.