

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université de Mentouri – Constantine
Faculté des sciences de l'ingénieur

Département d'informatique

Année :
N° d'ordre :
N° de série :

THESE

Pour l'obtention du diplôme de Doctorat en Sciences

Spécialité : Informatique

Thème

Une approche hybride pour la segmentation d'images Application aux images médicales

Présenté par : Nassima Mezhoud

Soutenue le 22.01.2012 à l'université de Constantine

Devant le jury composé de:

<u>Président :</u>	S .Chikhi	Prof	Université Mentouri Constantine
<u>Rapporteur :</u>	F.Hachouf	Prof	Université Mentouri Constantine
<u>Examineurs :</u>	L.Hamami	Prof	Ecole Nationale Polytechnique Alger
	M .K.Kholladi	Prof	Université Mentouri Constantine
	A.Moussaoui	M.C	Université Farhat Abès de Sétif

Année Universitaire : 2011/2012

REMERCIEMENTS



Je remercie tout d'abord le bon dieu pour m'avoir donné le courage et la santé pour accomplir ce travail.

Je remercie tout d'abord le bon dieu pour m'avoir donnée le courage et la santé pour accomplir ce travail.

Mes vifs remerciements accompagnés de toute ma gratitude vont ensuite à mon Promoteur HACHOUF FELLA professeur à l'université de Constantine, pour ses conseils judicieux, sa grande disponibilité et pour m'avoir suivie et orientée.

Je remercie le président de jury Monsieur. CHIKHI SALIM, professeur à l'université de Constantine qui nous a fait l'honneur de présider le jury.

Je remercie aussi Madame LATIFA HAMAMI, Professeur à Ecole Nationale Polytechnique (E.N.P) Alger, pour l'intérêt qu'elle a bien voulu porter à ce travail en acceptant d'être examinateur.

Je remercie aussi monsieur ABDELOUAHAB MOUSSAOUI, maître de conférence à l'université de Farhat Abès de Sétif e conférence à l'université de Sétif .pour l'intérêt qu'il a bien voulu porter à ce travail en acceptant d'être examinateur.

Je remercie également monsieur KHOLLADI MOHAMED KHEIRREDINE, maître de conférences à l'université de Constantine pour m'avoir fait l'honneur d'évaluer mon travail.

Enfin, que tous ceux qui nous ont aidés et encouragés de près ou de loin dans la concrétisation de ce projet, trouvent ici ma gratitude et mes sincères remerciements.

DÉDICACE



Je dédie ce travail à :

La mémoire de mon père.

Ma chère mère qui a toujours cru en moi,

Mes deux anges enfants : Asae et Abdelwadoud

Mes sœurs, surtout Houda,

Mon frère Tarek

Ma cousine : Atik Leïsa

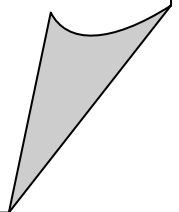
Mon mari : M. abdelhafid

Mon projet de vie : IFTM

Toute la famille Mezhoud

Mes amies, collègues et tous les gens qui m'aiment.

Nassima Mezhoud



RESUME

L'objectif de ce travail est de concevoir une nouvelle méthode hybride pour la segmentation d'images. Cette dernière sera appliquée aux images médicales. Pour ce faire, nous adaptons un modèle flou du clustering appelé ACE : Alternating Cluster Estimation et nous élaborons un schéma d'estimation des équations pour définir la relation existante entre l'ensemble des pixels de l'image et les différents clusters. Afin d'estimer les équations de mise à jour de ce modèle, nous construisons un système d'inférence flou de type Mamdani-Assilian (MA) constitué de cinq couches, chacune correspond à un des modules du système flou. Mis à part le premier module, qui représente un buffer nécessaire pour l'alimentation du réseau, chaque module correspond à une phase d'apprentissage flou. Ensuite nous implémentons les différents modules de ce système flou sur un réseau neuronal et finalement nous utilisons l'algorithme génétique pour entraîner ce réseau. Dans notre modélisation Le réseau neuroflou conçu au complet est un réseau multicouche de type feed-forward. Les entrées du réseau global sont les composantes colorimétriques du pixel. Nous avons choisi l'espace de couleur HSV. La sortie du réseau globale est une valeur numérique représentant un centre de classe. Les résultats obtenus par la méthode proposée étaient satisfaisants et encourageants

Mots clés : Segmentation d'images, images médicales, méthode hybride, réseau de neurones, algorithmes génétiques, espace couleur HSV.



ABSTRACT

The aim of this work is to design a new hybrid method for image segmentation. The latter will be applied to medical images. To do this, we adapt a fuzzy clustering model called ACE: Alternating Cluster Estimation and we develop a scheme for estimating equations to define the relation between the set of pixels of the image and the different clusters. To estimate the equations for updating the model, we construct a fuzzy inference system of Mamdani type-Assilian (MA) consists of five layers, each corresponding to one of the modules of the fuzzy system. Apart from the first module, which is a buffer needed to power the network, each module corresponds to a learning focus. Then we implement the various modules of this system on a fuzzy neural network and finally we use the genetic algorithm to train the network. The modeling in our neuro-fuzzy network is designed to complete a multilayer network-like feed-forward. Entered the global network are the components of the pixel color. We chose the HSV color space. The output of the overall network is a numerical value representing a different class. The results obtained by the proposed method were satisfactory and encouraging

Keywords: image segmentation, medical images, hybrid method, neural networks, genetic algorithms, HSV color space.



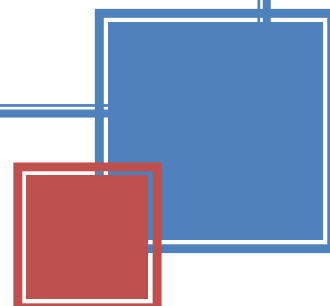
ملخص

الهدف من هذا العمل هو تصميم طريقة جديدة هجينة لتقسيم الصور, هذه الأخيرة ستطبق على صور طبية, من أجل هذا قمنا بتبني نموذج مبهم للتجميع يسمى: تقويم لتجميع تناوبي (ACE) كذلك حضرنا بيان تقويمي للمعادلات من أجل السماح لتحديد العلاقة القائمة بين كل بكسل من الصورة, و مختلف التجمعات. من أجل تقويم و تقدير معادلات التغيير لهذا النموذج, قمنا بإنشاء نظام تأثيري مبهم من نوع مامداني أسيليان الذي يتكون من خمس طبقات كل واحدة توافق أحد مقاييس النظام المبهم. ما عدى المقياس الأول الذي يمثل ممر ضروري لتغذية الشبكة, كل مقياس يوافق مرحلة من مراحل التدريب المبهم. بعد ذلك قمنا ببناء مختلف هذه الوحدات أو المقاييس الخاصة بالنظام المبهم المقترح على شبكة عصبية موافقة له و أخيرا باستعمال خوارزم جيني لتدريب هذه الشبكة. في إطار نموذجتنا للشبكة الخوارزمية المبهم كليتنا هي شبكة متعددة الطبقات من نوع فييد فوورد. مداخل الشبكة الكلية هم المكونات اللونية للبكسل. قمنا بإختيار لون الفضاء HSV. مخارج الشبكة الكلية هي قيمة رقمية تمثل مركز التجمع. النتائج المحصلة من طرف الطريقة المقترحة كانت مرضية و مشجعة.

الكلمات المفتاحية : تقسيم الصور، الصور الطبية، وأسلوب هجين، والشبكات

العصبية والخوارزميات الجينية، ولون الفضاء.

TABLE DES MATIÈRES



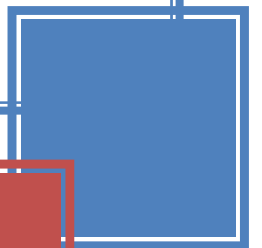
Titres		Page
Table des figures et des tableaux		
INTRODUCTION GENERALE		1
CHAPITRE 1 : Les réseaux de Neurones		
1.	Introduction	4
2.	Définition d'un réseau de neurones	4
3.	Topologie des réseaux de neurones	5
3.1.	Réseau multicouche	5
3.2.	Réseau à connexions locales	5
3.3.	Réseau à connexions récurrentes	6
3.4.	Réseau à connexion complète	6
4.	Types de réseaux de neurones	6
4.1	Réseaux de neurones Feed-Forward	6
4.1.1	Le perceptron de Rosenblatt	7
4.1.2	Le perceptron multicouche (PMC)	8
4.1.3	Adaline (Adaptive Linear Element)	9
4.2	Réseaux feed-back	11
4.2.1	Réseau de Hopfield	11
4.2.2	Les réseaux de neurones compétitifs ou carte auto organisatrices	12
4.2.3	Réseau de Gross Berg	13
5.	Apprentissage	14
5.1	Apprentissage supervisé	14
4.2	Apprentissage semi-supervisé	14
4.3	Apprentissage non supervisé	15
6.	Réseaux de neurones en segmentation d'images	15
7.	Les avantages des réseaux de neurones	16
8.	Les points faibles des réseaux de neurones	17
9.	Applications	17
10.	Conclusion	18
CHAPITRE 2 : Les Algorithmes génétiques		
1.	Introduction	19
2.	Les concepts importants des algorithmes génétiques	19
2.1.	Définition	19
2.2.	Génération	20
2.3.	Population	20
2.4.	Chromosome (ou individu)	21
2.4.1	Codage des chromosomes	21
2.5.	Gène	22
2.6.	Fonction d'adaptation	22

2.7.	Sélection	23
2.7.1	Sélection par rang	23
2.7.2	Probabilité de sélection proportionnelle à l'adaptation	23
2.7.3	Sélection par tournoi	24
2.7.4	Sélection uniforme	24
3.	Opérateurs des algorithmes génétiques	24
3.1.	La reproduction	24
3.2.	La mutation	25
3.3.	Le croisement	26
4.	Fonctionnement de l'algorithme génétique	28
4.1.	Algorithme général	28
4.2.	Cycle complet d'un algorithme génétique	29
4.2.1	La création de la population initiale	29
4.2.2	L'évaluation des individus	29
4.2.3	La création de nouveaux individus	30
4.2.4	Les croisements	30
4.2.5	Les mutations	30
4.2.6	L'insertion des nouveaux individus dans la population	30
4.2.7	Choix du nombre N d'individus à conserver	31
4.2.8	Passage à la génération suivante	31
4.2.9	Réitération du processus	31
5.	Les algorithmes génétiques dans la segmentation des images	32
6.	Les algorithmes génétiques associés aux réseaux de neurones pour la segmentation d'images	33
7.	Domaines d'application des algorithmes génétiques	34
8.	Les avantages des algorithmes génétiques	34
9.	Limitations des algorithmes génétiques	35
10.	Conclusion	35
CHAPITRE 3 : La logique Floue et les modèles flous		
1.	Introduction	36
2.	logique floue	36
3.	La théorie des sous ensembles flous	36
3.1.	Les fonctions d'appartenance	37
3.2.	Les caractéristiques d'un sous ensemble flou	37
3.2.1	Le noyau	37
3.2.2	Le support	38

3.2.3	La hauteur	38
3.2.4	α coupe	38
3.3.	Les opérations ensemblistes	38
3.3.1	La réunion	38
3.3.2	L'intersection	39
3.3.3	Le complément	39
3.4.	Différentes représentations de sous-ensembles flous	40
3.5.	Les relations floues	40
3.6.	Règles floues	41
3.6.1	Définition	41
3.6.2	Variables linguistiques	42
3.6.3	Proposition floue élémentaire	42
3.6.4	Proposition générale floue	42
4.	Système flou	43
4.1.	Fuzzification ou quantification floue	44
4.2.	Inférence	44
4.3.	La défuzzification	44
4.3.1	La méthode du maximum	45
4.3.2	Le centre de gravité	45
5.	Modèles flous	45
5.1.	Définition d'un modèle flou	45
5.2.	Modèle de Mamdani-assilian (MA)	46
5.3.	Le modèle de Takagi-sugeno	47
5.4.	Méthodes de modélisation	48
5.4.1	Modélisation avec évaluation de règles	48
6.	Application de la logique floue pour la segmentation d'images médicales	49
6.1	Représentation des objets par des sous ensembles flous	49
6.1.1	Application et amélioration du FCM	50
6.2	Modélisation des interactions entre différentes structures par sous-ensembles flous.	51
6.2.1	Connectivité flou et objets flous.	52
7.	Caractéristiques, avantages et limitations de la logique floue	53
7.1.	Caractéristiques	53
7.2.	Avantages	54
7.3.	Limitations	54
8.	Conclusion	54
CHAPITRE 4 : Approche hybride proposée		
Partie I : Conception Détaillée de l'approche proposée		
1.	Problématique et solutions	55
2.	Description de l'ACE	57
3.	Méthodologie de la conception	59

3.1.	Adaptation de l'ACE	59
3-2.	Conception d'un schéma d'estimation	60
3.2.1	Construction du modèle initial	60
3.2.2	Conception du réseau neuroflou	63
3.2.3	Conception d'un algorithme d'apprentissage génétique	68
4.	Discussion	72
Partie II : Prsésentation et Analyse des résultats		
1.	Introduction	73
2.	Fonction d'évaluations	73
3.	Résultats Expérimentaux Et Discussion	74
3.1.	Images utilisées	74
3.1.1	Images réelles	74
3.1 .2	Fantômes (Images de synthèse)	74
3.2.	Expériences en niveau de gris	75
3.2.1	IRM Cerveau humain (sain)	75
3.2.2	IRM Cerveau humain (avec une Tumeur)	76
3.2.3	Coupe Axiale du cerveau en T2	78
3.2.4	Coupe Axiale du cerveau en T1	79
3.2.5	Coupe Sagittale du cerveau	80
4.	Expériences en couleur	81
4.1.	Image cérébrale SPECT	81
4.2.	Images cérébrales couleurs de différentes coupes	83
4.3.	Images Couleurs anatomiques	84
5.	Différentes images médicales	85
6.	Conclusion	89
	Conclusion générale	90
	Références Bibliographiques	92
	Liste des Contributions	97

LISTE DES FIGURES ET TABLEAUX



N° Fig	Titres	Page
Chapitre 1 : Réseaux de neurones		
Fig. 1.1.	Définition des couches d'un réseau multicouche	5
Fig. 1.2.	Réseau à connexions locales	5
Fig. 1.3.	Réseau à connexions récurrentes	6
Fig. 1.4.	Réseau à connexions complète	6
Fig. 1.5.	Perceptron calculant la fonction XOR	7
Fig. 1.6.	Description du perceptron multicouche	8
Fig. 1.7.	Schéma d'un modèle de la rétropropagation de l'erreur	9
Fig. 1.8.	Forme générale Adaline	10
Fig. 1.9.	Exemple d'ALN à 8 entrées	10
Fig. 1.10.	Réseau de Hopfield	11
Fig. 1.11.	Mise en correspondance de l'espace d'entrée avec l'espace du réseau	12
Fig. 1.12.	L'architecture d'une carte auto organisatrice (réseau 2D)	13
Fig. 1.13.	l'architecture d'une carte réseau Gross Berg	13
Chapitre 2 : : Les algorithmes génétiques		
Fig. 2.1.	Les cinq niveaux d'organisation d'un algorithme génétique	21
Fig. 2.2.	Illustration schématique du codage des variables réelles	22
Fig. 2.3.	Schéma d'une roulette	24
Fig. 2.4.	Principe de l'opérateur de mutation	25
Fig. 2.5.	Slicing Croisement	27
Fig. 2.6.	Slicing Croisement à 2 points	27
Fig. 2.7.	Croisement binaire en un point	27
Fig. 2.8.	Principe général des algorithmes génétiques	29
Fig. 2.9.	Cycle complet d'un algorithme génétique	32
Chapitre 3 : La logique floue et les modèles flous		
Fig. 3.1.	le noyau, le support, la hauteur d'un sous ensemble flou	38
Fig. 3.2.	Partition floue de l'univers du discours	39
Fig. 3.3.	Ensemble flou « personne petite ou moyenne »	39
Fig. 3.4.	Ensemble flou « personne petite et moyenne »	39
Fig. 3.5.	Ensemble flou « personne non petite »	39
Fig. 3.6.	x approximativement égal à 3	40
Fig. 3.7.	Représentation d'une règle floue.	41
Fig. 3.8.	Variable linguistique (V, X, TV) décrivant la surface d'un appartement	42
Fig. 3.9.	Schéma d'un système floue.	43

Chapitre 4 : Approche Hybride proposée**Partie I : Conception Détaillée de l'approche proposée**

Fig. 4.1.	Description du système.	56
Fig. 4.2.	AO versus ACE	57
Fig. 4.3.	Schéma illustrative des étapes de la conception	59
Fig. 4.4.	Fonctions d'appartenance de l'attribut hue	61
Fig. 4.5.	Fonctions d'appartenance de l'attribut classe.	61
Fig. 4.6.	Fuzzification des entrées	63
Fig. 4.7.	Couche de fuzzification	64
Fig. 4.8.	Conjonction des propositions de l'antécédent	65
Fig. 4.9.	Couche de conjonction	65
Fig. 4.10.	Implication de l'antécédent vers le conséquent	66
Fig. 4.11.	Couche de l'implication	66
Fig. 4.12.	Agrégation et Defuzzification.	67
Fig. 4.13.	Couche d'agrégation et de defuzzification	67
Fig. 4.14.	Structure réseau neuroflou conçu	68
Fig. 4.15.	Codage réel	70
Fig. 4.16.	Croisement multipoint	71
Fig. 4.17.	Mutation	72

Partie II : Présentation et Analyse des résultats

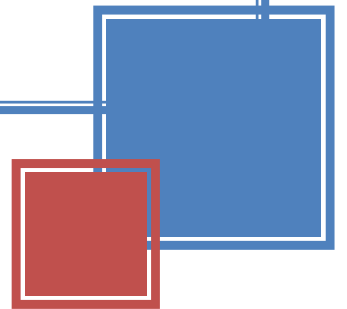
Fig. 4.18.	Exemple de cartes floues du fantôme [Kwan-96]	75
Fig. 4.19.	Segmentation de la coupe axiale du cerveau humain avec des différents paramètres.	76
Fig. 4.20.	Segmentation de la coupe coronale du cerveau avec différents paramètres	77
Fig. 4.21.	Segmentation de la coupe Axiale du cerveau T2 avec différentes classes	79
Fig. 4.22.	Segmentation de la coupe Axiale du cerveau T1 avec différentes classes.	80
Fig.4.23.	Segmentation de la coupe sagittale du cerveau avec différentes classes	81
Fig. 4.24.	Segmentation d'image spect (coupe axiale du cerveau humain)	82
Fig. 4.25.	Segmentation d'Images cérébrales IRM couleurs de différentes coupes	83
Fig. 4.26.	Segmentation d'Image anatomique (Brain X section)	84
Fig. 4.27.	Segmentation d'Image : Cranial_ nerve	84
Fig. 4.28.	Segmentation d'Image : Auditory-pathway	84
Fig. 4.29.	Segmentation d'une Coupe axiale de l'abdominal	87
Fig. 4.30.	Segmentation d'une Coupe sagittale couleur du cerveau	87
Fig. 4.31.	Segmentation d'une Coupe axiale du Pénis	87
Fig. 4.32	Segmentation d'une Coupe sagittale d'IRM en T1 :((méningiome)	88

Fig. 4.33	Segmentation d'une Coupe sagittale d'IRM en T2 (système-ventriculaire)	88
Fig. 4.34	Segmentation d'une Coupe axiale d'IRM au niveau de la base du crane	88
Fig. 4.35	Segmentation d'une Coupe axiale anatomique de la base du crane sur cadavre (Selle turcique)	89

La Liste des tableaux

N° Tab	Désignation	Page
	Chapitre 1 : Réseaux de neurones	
Tab. 1.1.	Différents Types et seuils de fonction de transfert	4
	Chapitre 3 : La logique floue et les modèles flous	
Tab. 3.1.	Différentes représentations de sous-ensembles flous.	40
	Chapitre 4 : Approche Hybride proposée	
Tab 4.1.	Initialisation des centres des clusters ou prototypes	62

INTRODUCTION GÉNÉRALE



- **Contexte de la thèse**

La segmentation est un vaste sujet d'étude et fait partie des grands thèmes de l'imagerie numérique. A ce titre, de nombreuses publications font état de segmentations. Les techniques de segmentation d'images médicales constituent un outil de diagnostic très important pour le médecin. Généralement, les images médicales sont obtenues avec des méthodes d'acquisition différentes. Comment préférer l'une ou l'autre est un débat ouvert qui fait rage dans bien des laboratoires. En effet, pour valider correctement une segmentation d'objets naturels, comme en imagerie médicale, il faut disposer de la vérité terrain ; ce qui est bien difficile dans le cas de la segmentation, car comment définir de façon précise où commencent et où s'arrêtent les objets sur une image ? Il n'y a donc pas une mais des segmentations possibles sur une même image et elles sont bien souvent subjectives. De même, selon ce que nous voulons segmenter, certaines techniques seront plus à même d'y parvenir.

Le traitement des images médicales est une discipline nouvelle, riche et variée. L'importance que revêt l'imagerie médicale tient d'abord au fait qu'une image est un concentré d'information bien plus efficace qu'un texte ou qu'une explication verbale. Les techniques de traitement d'images ont permis une prolifération de recherches dans ce domaine ; mémoires, thèses, revues, conférences...etc. Portés sur des thèmes différents, le but est de développer de nouvelles façons de manipulation d'images pour améliorer la qualité du service médical, et d'automatiser au maximum les tâches cliniques.

Actuellement, le diagnostic médical devient de plus en plus facile, grâce au développement des techniques d'imageries médicales. Avec la lancée des méthodes de traitement d'images, la clinique moderne donne des rapports pathologiques crédibles, avec des taux d'erreurs réduits.

Dans ce contexte, notre intérêt est porté sur l'une des étapes critiques du traitement visuel : la segmentation d'images médicales ; l'une des étapes critiques du traitement d'images. Ce processus permet d'engendrer une description compacte de l'image et donc symbolique plus exploitable que l'ensemble des pixels ; elle se résume en une action de découper chaque image source pour générer une image porteuse d'informations plus précises.



- **Problématique**

Une variété de méthodes de segmentation d'images a été proposée ayant toutes, malgré la diversité des moyens qu'elles emploient, pour but le partitionnement d'une image en régions homogènes. Parmi les méthodes proposées, nous nous intéressons particulièrement à la méthode de clustering (regroupement) qui bon nombre de ses algorithmes ont été adaptés pour la segmentation d'images.

L'objectif de notre travail est justement de construire une nouvelle méthode de clustering flou basée sur les modèles, dans le but, de segmenter une image médicale en régions homogènes. Pour ce faire nous suggérons d'adapter un modèle flou basé sur l'estimation simultanée des classes et d'élaborer un schéma d'estimation des équations de ce modèle.


Le modèle global flou que l'on cherche à extraire est de type Mamdani-Assilian. Il est constitué d'un ensemble de règles floues « if-then » desquelles les prémisses et les conclusions estiment, respectivement : l'équation de mise à jour de la matrice des partitions et celle de mise à jour des prototypes du modèle.

Afin d'optimiser ce modèle flou, nous utilisons deux outils d'optimisation: les réseaux neuronaux, pour un besoin de parallélisme et de puissance de calcul qui s'avèrent essentiels dans les traitements d'images médicales et les algorithmes génétiques tel un algorithme d'apprentissage, pour tirer profit de leur capacité d'explorer un grand espace de solutions potentielles en parallèle et de converger vers une solution optimale. Nous pensons que de cette manière nous bénéficierons des apports de la théorie des ensembles flous, des réseaux neuronaux et des algorithmes génétiques, pour rendre notre méthode plus performante.

- **Organisation de la thèse**

La thèse est organisée en quatre chapitres.

Dans le **premier chapitre**, nous définissons d'abord les réseaux de neurones, Après nous donnons les topologies, les types de réseaux existants et les modes d'apprentissage. Ensuite nous citons les avantages, les limitations les et les applications des réseaux de neurones et enfin nous l'avons consacré une partie une partie montrant la contribution de réseaux de neurones en segmentation d'images.



Puis dans **le deuxième chapitre**, nous donnons, une description complète des algorithmes génétiques (Concepts importants, Opérateurs, fonctionnement et cycle de vie) ainsi que ses avantages, limitations et domaines d'application et nous abordons aussi la contribution des algorithmes génétiques en domaine de la segmentation d'images

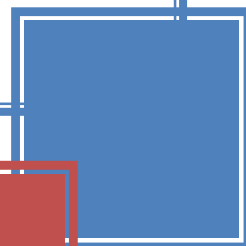
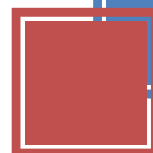
Dans **le troisième chapitre**, en premier lieu, nous introduisons la théorie de la logique floue, suivi par les plus importants de ses concepts de base, ensuite, nous mettons l'accent sur et la notion des règles floues et la notion des modèles flous, et nous énumérons les avantages, les limitations et les domaines d'utilisation de la logique floue, finalement nous présentons aussi la contribution de la logique floue en segmentation d'images.

Le dernier et **le quatrième chapitre** est constitué de deux parties :

- La première introduit la problématique et les raisons de nos différents choix, par la suite, une description de l'algorithme que l'on compte adapter et une méthodologie de conception, sont présentées.
- La deuxième partie contient le fruit de notre travail, les résultats de la segmentation enrichie par évaluation et une analyse de ces derniers.



CHAPITRE 1



1. Introduction

Les neurones sont des automates élémentaires dont le mode de fonctionnement s'inspire de celui des neurones biologiques. Les motivations de ces réseaux se résument en deux principaux : d'une part la modélisation du cerveau, et d'autre part la réalisation d'algorithmes et de machines spécialisées dans des tâches de reconnaissance des formes par exemple, dont les performances pourraient être supérieures à celles des algorithmes et ordinateurs classiques. Le domaine des réseaux de neurones s'est développé durant les années quatre-vingt.

Afin d'éclaircir et bien situé notre approche dans le domaine de réseaux de neurones, nous présentons dans ce chapitre, la topologie, les types et le mode d'apprentissage de ces derniers. Nous donnons aussi la contribution de réseaux de neurones en segmentation d'images ainsi que ses avantages, ses points faibles et leurs applications.

2. Définition d'un réseau de neurones

Un réseau de neurone formel est constitué d'un grand nombre de cellules de base interconnectées, il peut être défini par [HAC 01] :

- Un ensemble de neurone du même type : nœud.
- Un ensemble de connections pondérées entre les nœuds : connaissances.
- Un seuil associé à chaque nœud.
- Une fonction de transfert f qui peut être à seuil, linéaire par morceau, sigmoïde ou dans certaines applications une tangente hyperbolique.


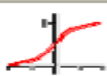

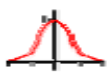
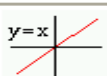
Pas unitaire		$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$
Sigmoïde		$f(x) = \frac{1}{1+e^{-\beta x}}$
Linéaire Seuillée		$f(x) = \begin{cases} 0 & \text{if } x \leq x_{min} \\ mx+b & \text{if } x_{min} < x < x_{max} \\ 1 & \text{if } x \geq x_{max} \end{cases}$
Gaussienne		$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
Identité		$f(x) = x$

Tableau 1.1 : Différents Types et seuils de fonction de transfert [MAS 06].

3. Topologie des réseaux de neurones

La topologie d'un réseau de neurone est la façon dont les neurones sont reliés (d'une manière générale l'architecture locale du réseau). Il n'existe pas une méthode formelle pour déterminer le nombre de neurones ou de couches ni les connexions nécessaires pour résoudre un problème donné (sauf le nombre de neurones de la couche d'entrée et de sortie). [ABD 94].

Nous présentons les différentes topologies [TOU 92] :

3.1 Réseau multicouche

Les neurones sont arrangés par couche. Il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales.

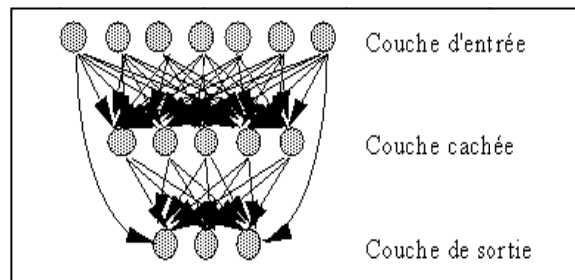


Figure 1.1 Définition des couches d'un réseau multicouche.

3.2 Réseau à connexions locales

Il s'agit d'une structure multicouche, mais qui à l'image de la rétine, conserve une certaine topologie. Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche avale.

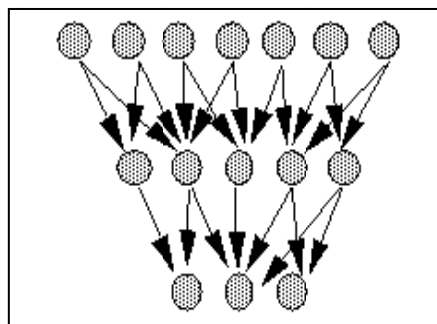


Figure 1.2 Réseau à connexions locales

3.3 Réseau à connexions récurrentes :

Les connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche. Ces connexions sont le plus souvent locales.

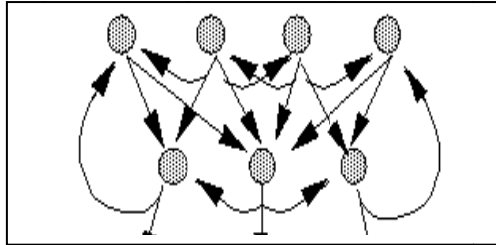


Figure 1.3 Réseau à connexions récurrentes

3.4 Réseau à connexion complète :

C'est la structure d'interconnexion la plus générale (figure.1.5). Chaque neurone est connecté à tous les neurones du réseau (et à lui-même).

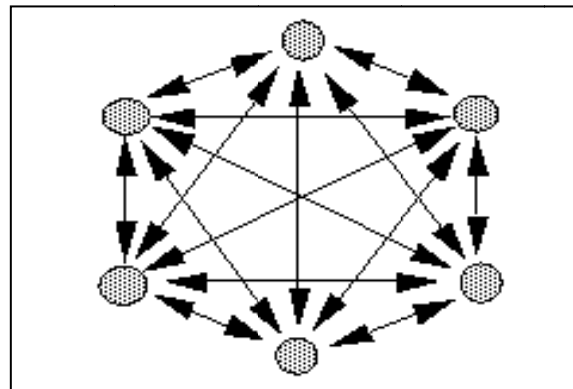


Figure 1.4 Réseau à connexions complète

4. Types de réseaux de neurones

La topologie d'un réseau de neurone influence la manière de propager les informations dans ce réseau, de ce fait, il existe principalement deux types de réseau de neurones: Feed-Forward et Feed-back.

4.1 Réseaux de neurones Feed-Forward

Dans les réseaux de neurones Feed-Forward ou à couches, les sorties de n'importe qu'elle couche ne peuvent être retro propagées vers les couches précédentes, d'où les sorties actuelles ne

sont influencées ni par les sorties futures ni par elles-mêmes. Une seule passe de calcul est effectuée.

4.1.1 Le perceptron de Rosenblatt

Le perceptron peut être considéré comme le premier des réseaux de neurones. Il se voulait un modèle de l'activité perceptive, le but du perceptron est d'associer des configurations des formes ou des stimulus, présentées en entrée à des réponses [ABD 94]. Le perceptron se compose de trois couches

- Les cellules détectrices de phénomènes extérieurs rassemblées dans la rétine où les exemples sont présentés.
- Les cellules d'association intermédiaires (pré-processeurs) : connectées aux cellules de la rétine et aux cellules de décision (sortie). Chacune de ces cellules calcule une fonction F_i fixée à l'avance $F_i : \mathbb{R}^n \rightarrow \mathbb{R}$.
- Les cellules de décision (de sortie) : reçoivent leurs entrées des cellules d'association. Elles représentent la sortie du perceptron, dans le cas du perceptron simple ces dernières sont réduites à une seule. Seules ces dernières cellules renvoient des signaux au pré-processeur jusqu'à ce que les connexions se stabilisent.

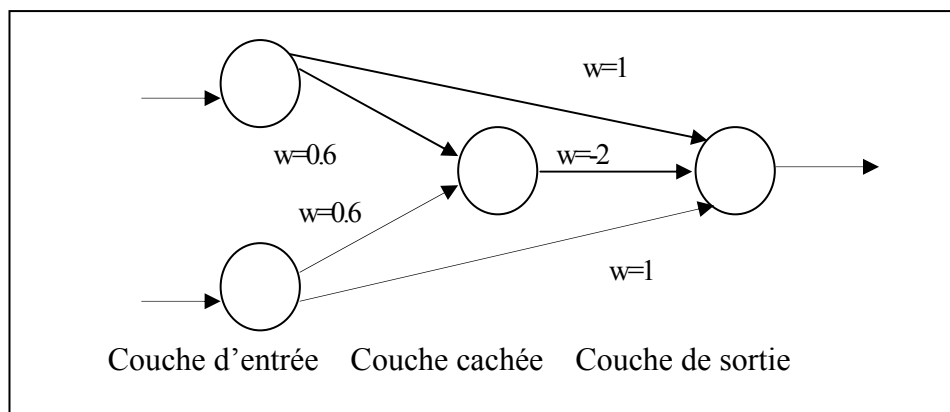


Figure 1.5 Perceptron calculant la fonction XOR

Les neurones d'une même couche ne sont pas connectés entre eux. La fonction de transfert utilisée est une fonction à seuil. Un apprentissage supervisé est utilisé, ce qui signifie qu'on fournit au réseau des exemples à mémoriser. Le perceptron peut apprendre à classifier des exemples par modification des poids des cellules en se basant sur la règle de HEBB [MIC 93].

4.1.2 Le perceptron multicouche (PMC)

Le perceptron multicouche (MLP : multi layer perceptron) est une généralisation du perceptron qui possède qu'une couche. C'est un réseau orienté de neurones artificiels organisés en couches de telle sorte que les neurones de deux couches adjacentes sont connectés complètement, et où l'information se propage dans un seul sens, de la couche d'entrée vers la couche de sortie. Les entrées des neurones de la deuxième couche sont donc les sorties des neurones de la première couche.

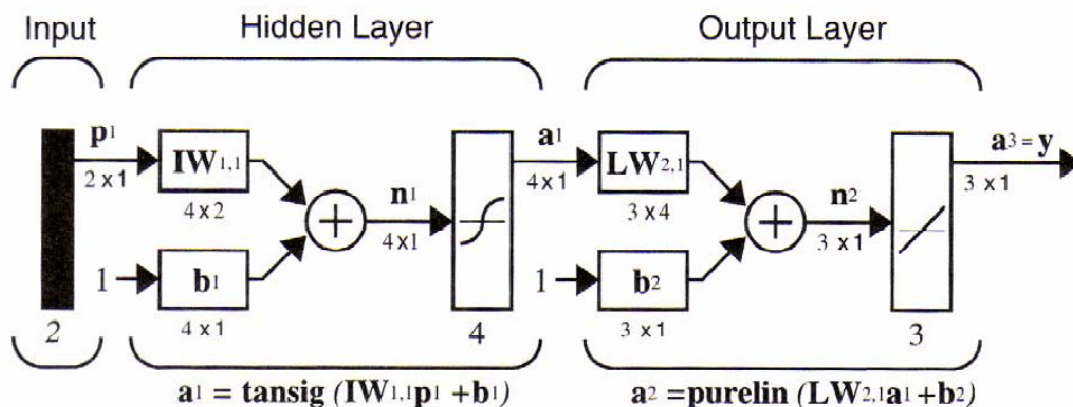


Figure 1.6 Description du perceptron multicouche

La couche d'entrée représente toujours une couche virtuelle associée aux entrées du système. Les neurones de cette couche sont reliés au monde extérieur et reçoivent tous le même vecteur d'entrée. Les sorties des neurones de la dernière couche correspondent toujours aux sorties du système (fournit les résultats). Enfin les neurones de la couche cachée n'ont aucun lien avec l'extérieur.

Dans le cas général un perceptron multi couche peut posséder plusieurs couches cachées et plusieurs neurones par couche.

Les neurones sont reliés entre eux par des connexions pondérées. Ce sont les poids de ces connexions qui gouvernent le fonctionnement du réseau et programment une application de l'espace des entrées vers l'espace des sorties à l'aide d'une transformation non linéaire.

La modification des poids du PMC s'effectue par l'algorithme de rétropropagation.

▪ La rétro propagation ou algorithme d'apprentissage de « Backpropagation » [MAS O6]

La rétro propagation a été créée en généralisant la loi d'apprentissage de Widrow-Hoff à des réseaux de neurones multicouches constitués de fonctions de transfert différentiables. Les vecteurs d'entrées et les vecteurs cibles correspondant sont utilisés pour apprendre le réseau.

Les réseaux de neurones constitués de biais et de fonctions de transfert « sigmoïdale » et une couche de sortie constituée de fonctions de transfert linéaires sont capables d'approximer n'importe quelle fonction possédant un nombre fini de discontinuités. La règle delta impose toujours $\Delta W = -\alpha \frac{\partial F}{\partial W}$. La difficulté réside toujours dans le calcul de $\frac{\partial F}{\partial W}$.

L'algorithme de rétropropagation est la généralisation de la règle de Widrow et Hoff pour un réseau multicouche qui consiste à minimiser la fonction d'erreur quadratique. Il utilise les dérivations composées et ne présente aucune difficulté particulière. Le principe de cet algorithme se résume en deux tapes : étape de propagation et une d'adaptation de poids.

La première étape consiste à propager vers l'avant les activations ou les états des neurones jusqu'à obtenir une sortie calculée par le réseau.

La seconde étape compare la sortie calculée et la sortie désirée. On modifie alors les poids de telle sorte qu'à la prochaine itération l'erreur obtenue entre la sortie calculée et la sortie désirée soit minimisée. Nous présentons un schéma de la rétropropagation de l'erreur.

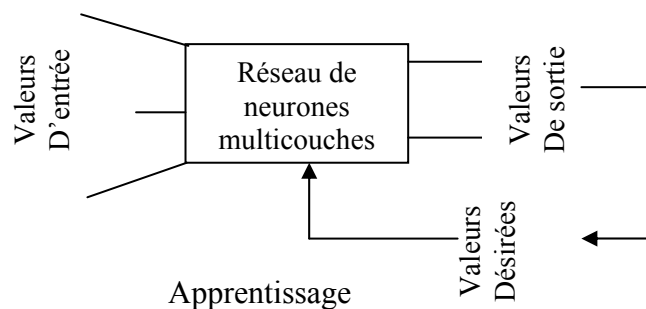


Figure 1.7 Schéma d'un modèle de la rétropropagation de l'erreur

4.1.3 Adaline (Adaptive Linear Element)

Proposé par Widrow Hoff, il est composé d'un seul neurone, utilisé dans les décisions binaires et dans l'association d'un modèle d'entrée à une sortie : il est capable de reconnaître une lettre A par exemple quelle que soit sa taille et son orientation.

La forme générale d'Adaline est présentée par la figure suivante :

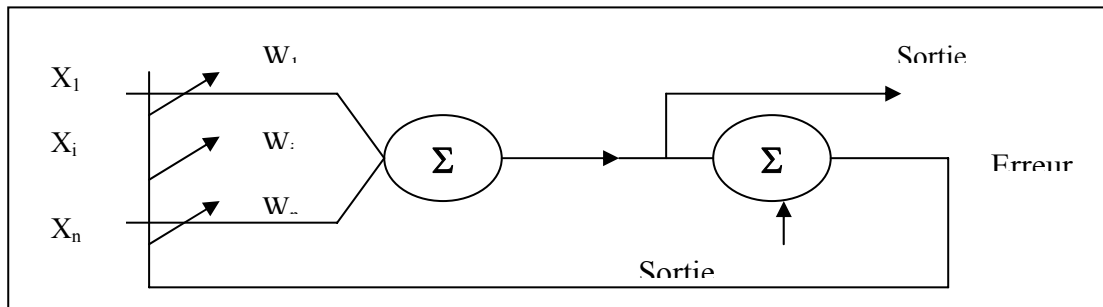


Figure 1.8 Forme générale Adaline

Dans ce modèle, les poids sont déterminés par la méthode du quadratique, il utilise l'apprentissage supervisé à l'aide de la règle LMS « Least Mean Square » qui n'est qu'une variante de la règle d'apprentissage Delta normalisée par Windrow et Hoff, cette dernière consiste à minimiser la fonction d'erreur quadratique [GRI 02].

a. Réseaux de neurones logiques (Adaptive Logical Network) :

L'apparition des réseaux de neurones logiques (ALN) remonte au milieu des années 70, il s'agit également de réseaux fonctionnant en apprentissage supervisé, cependant la structure des réseaux logiques est complètement différente de celle des réseaux utilisant des techniques de rétro propagation du gradient.

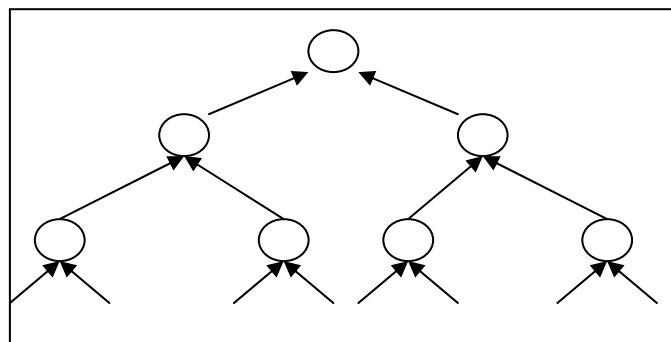


Figure 1.9 Exemple d'ALN à 8 entrées

Un neurone dans un ALN est une porte logique à deux entrées (gauche et droite) booléennes et une sortie booléenne, cette porte peut être une porte OU une porte ET, une porte droite ou une porte gauche. Un réseau de neurone logique est un arbre binaire où chaque nœud est un neurone élémentaire. [ELA 87].

L'apprentissage dans les réseaux logiques se fait par la modification des fonctions des portes logiques.

4.2 Réseaux feed- back

Dans ce type de réseaux, chaque cellule est reliée à tous les autres et possède même un retour sur elle-même. Les signaux peuvent être propagés en avant ou rétro-propagés.

4.2.1 Réseau de Hopfield

Ces réseaux sont des réseaux récurrents, un peu plus complexes que les perceptrons multicouches. Chaque cellule est connectée à toutes les autres et les changements de valeurs de cellules s'enchaînent en cascade jusqu'à un état stable. Ces réseaux sont bien adaptés à la reconnaissance de formes [MAS 06].

Il est typique d'un réseau totalement connecté ce réseau est surtout utilisé au tant que mémoire associative. Ce réseau est formé de N neurones ne pouvant envoyer et recevoir que des valeurs binaires $\{1, -1\}$. La fonction de transfert de chaque neurone est une fonction à seuil [MIC 93]. Chaque neurone est relié à tous ses voisins par des poids mais n'agit pas sur lui-même. L'algorithme régissant le réseau est basé sur une fonction de LYAPOUNOV qui sert à étudier l'équilibre dans des systèmes complexes [MAS 95]. La première phase consiste à initialiser les connexions entre neurones : il s'agit de mémoriser M exemples. On présente une forme au réseau, on itère jusqu'à ce que le réseau se stabilise.

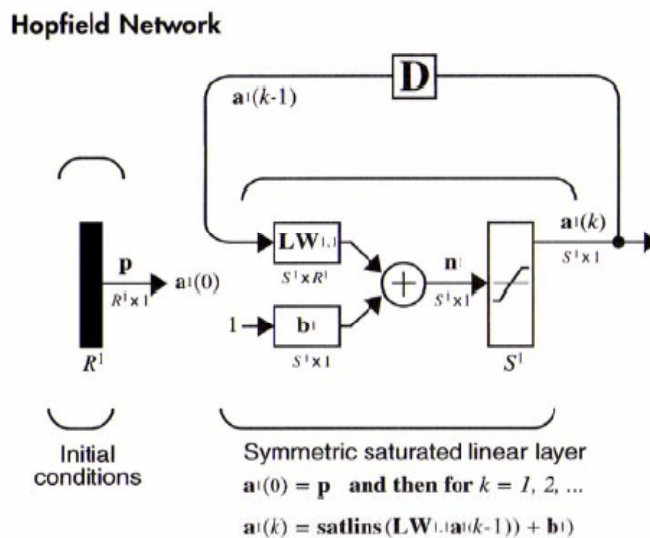


Figure 1.10 Réseau de Hopfield

4.2.2 Les réseaux de neurones compétitifs ou cartes auto organisatrices

Ces réseaux peuvent détecter des régularités et des corrélations dans les données d'entrées et peuvent adapter leurs réponses en fonction de ces entrées.

Les neurones de ces réseaux peuvent reconnaître des groupes dans les vecteurs d'entrée. Les cartes auto organisatrices peuvent apprendre à reconnaître des groupes de vecteurs similaires, ce qui entraîne que les neurones physiquement proches les uns des autres dans la couche de neurones à répondre à ces vecteurs d'entrée similaires. ²

La LVQ méthode (Learning Vector Quantization) permet d'entraîner les réseaux de neurones compétitifs de manière supervisée. Comme la classification se fait par le calcul de distance entre les poids des vecteurs d'entrée, il n'y a pas de mécanisme strict qui permettent de définir si les vecteurs d'entrés sont dans la mêmes classe ou non. Celles-ci peuvent être imposées par l'utilisateur [MAS O6].

a. Les cartes auto organisatrices de KOHONEN

Ce modèle auto organisatrice appartient à la classe des réseaux à compétition. Les neurones de la couche de sortie entrent en compétition, un seul neurone de sortie est active pour une entrée donnée. Cette compétition entre les neurones est réalisée grâce à des connexions latérales inhibitrices. Ces cartes s'organisent par rapport aux exemples d'entrée présentés en respectant les contraintes topologiques de l'espace d'entrée. Il y a une mise en correspondance de l'espace d'entrée avec l'espace du réseau ; les zones voisines de l'espace d'entrée sont voisines sur la carte auto organisatrice [TOU 92].

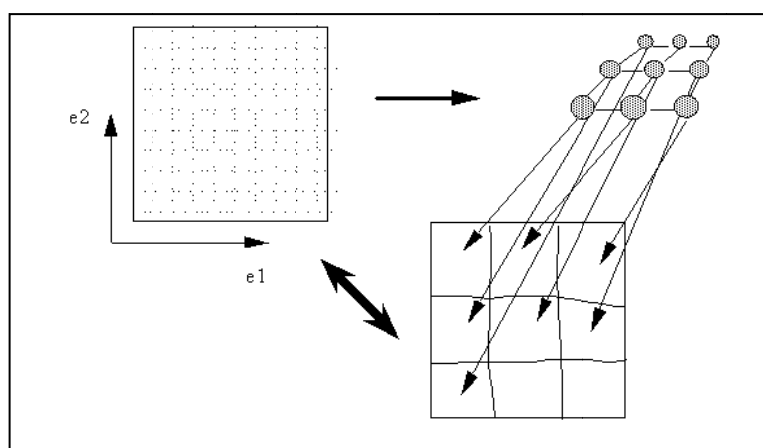


Figure 1.11 Mise en correspondance de l'espace d'entrée avec l'espace du réseau

Après avoir initialiser les connexions de manière aléatoire, on calcule la distance euclidienne qui s'étend entre l'entrée et chaque neurone de sortie, puis on choisit le neurone de distance minimale, il ne reste qu'à modifier les poids des connexions des neurones appartenant à son voisinage et d'itérer le processus « suivant une règle de HEBB » après avoir réduit les voisinages [MIC 93],[MAS 95].

Ces réseaux sont très utilisés pour l'analyse de données. Ils permettent de cartographier en deux dimensions et de distinguer des groupes dans des ensembles de données [MAS 06].

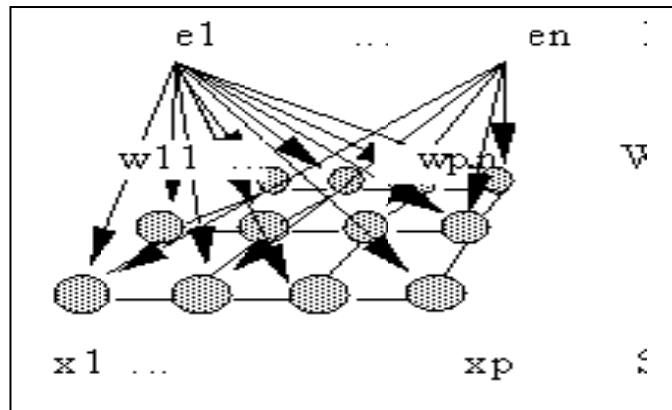


Figure 1.12 l'architecture d'une carte auto organisatrice

4.2.3 Réseau Gross B3r2

La topologie du réseau est formée de deux couches, une d'entrée et une de sortie [HAT 91], où les neurones sont entièrement connectés dans un plan. La première forme donnée en entrée est choisie comme un représentant des formes à mémoriser. On compare ensuite une seconde forme à la première à l'aide d'une distance qui reste à définir. Si cette distance est inférieure à un certain seuil, cette seconde forme n'est qu'un autre représentant de la même catégorie, alors que dans le cas contraire, on crée une nouvelle catégorie.

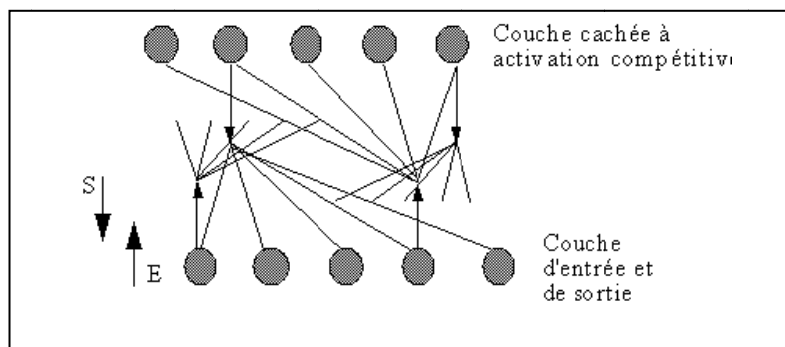


Figure 1.14 l'architecture d'une carte réseau Gross Berg

Bien évidemment, plus le nombre d'entrées est important, plus le nombre de catégories augmentera et modifiera la topologie du réseau, puisqu'une catégorie correspond à un nœud dans la couche de sortie comme le modèle HOPFIELD, les données sont binaires, mais l'existence de deux couches permet le Feed-back des sorties vers les entrées. Il faut noter aussi que les neurones correspondant à la couche de sortie sont connectés entre eux, permettant d'implémenter ainsi un modèle d'inhibition latérale entre deux neurones.

L'intérêt de ce modèle repose sur sa faculté de s'auto organiser ainsi qu'à l'apprentissage non supervisé (sans exemples). Il donne de mauvais résultats si les formes sont trop bruitées.

5. Apprentissage

Le fait de définir une architecture spécifique d'un réseau est insuffisant pour résoudre un problème donné. Il faut déterminer une configuration de poids adaptée au problème, c'est le rôle du processus d'apprentissage. [1]. A l'issue de l'apprentissage, les poids sont fixés : c'est alors la phase d'utilisation. [TOU 92].

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré, il peut être séparé en trois familles :

- Apprentissage supervisé
- Apprentissage non supérieur
- Apprentissage semi supervisé

5.1 Apprentissage supervisé

Il consiste à apprendre une association particulière entre deux ensembles de vecteurs, on présente au réseau des vecteurs d'entrée en lui imposant, comme des contraintes, les vecteurs de sortie que l'on désire lui faire associer à ces entrées.

5.2 Apprentissage semi-supervisé

Cette forme d'apprentissage s'apparente à la précédente par la présence du maître mais la valeur de l'erreur en sortie n'est plus disponible. La seule information disponible est un signal de l'échec ou simplement de l'erreur, c'est une information très pauvre.

5.3 Apprentissage non supervisé

Il consiste à découper l'ensemble des vecteurs d'entrée en classes d'équivalence sans qu'il soit nécessaire de donner au réseau neuronal les noms des classes pour chaque exemple. La séparation en classes d'équivalence s'opère par mesure de ressemblance entre les vecteurs proposés la contrainte que l'on doit imposer au réseau est le nombre de classes d'équivalence.

Pour ces trois types d'apprentissage, il y a également un choix traditionnel entre :

- L'apprentissage << off-line >> : toutes les données sont dans une base d'exemples d'apprentissage qui sont traités simultanément ;
- L'apprentissage << on-line >> : Les exemples sont présentés les uns après les autres au fur et à mesure de leur disponibilité.

6. Réseaux de neurones en segmentation d'images

Classiquement, en segmentation d'images médicales, les réseaux de neurones sont utilisés comme classifieurs. Les poids synaptiques sont déterminés par apprentissage sur une base d'images dont le résultat de segmentation est connu. On parle alors de réseau de neurones supervisé. Souvent, les neurones d'entrées sont les différentes images IRM disponibles et les neurones de sorties nous donnent alors les différentes classes recherchées [HUL 03]. Il est en outre possible d'introduire des informations a priori en plus des volumes et donc de donner plus de robustesse à cette classification. Mais l'inconvénient majeur de cette méthode par réseaux de neurones est l'étape d'apprentissage qui demande une intervention manuelle pour donner cette vérité terrain dont le réseau a besoin pour calculer les poids synaptiques [JER 08].

Dans [Bou 10] les auteurs segmentent des images et détectent les contours ainsi que le suivi d'objets par réseaux de neurones impulsif sur une base de 12 images couleur de cytologie de tumeurs bronchiques présentant chacune des dizaines à des centaines de cellules. Donc, pour une modélisation plus réaliste, il prend en compte les aspects temporels en utilisant les neurones impulsif.

La référence [BER 09] présente un système de traitement des images scanographiques et IRM cérébrale qui est basée sur la méthode « split and merge » pour l'étape de segmentation et l'application du réseau de neurone de type Perceptron Multicouches (PMC) pour la détection des anomalies dans ces images. L'activation du MLP, signifie la simulation de l'image d'entrée (présentée par un ensemble de régions) par MLP qui est déjà créée auparavant. Ce réseau contient :

- Une couche d'entrée: qui compte 5 neurones représentant les attributs choisis de chaque région

de l'image. Les attributs extraits à partir des régions de l'image, après la phase de segmentation sont: la surface, la variance, le niveau de gris moyen, l'élongation, la compacité. La couche cachée : contient 50 neurones. Ce nombre est choisie après plusieurs tests afin d'aboutir à une meilleure généralisation. Dans [LIL 05], les auteurs mettent en œuvre de nouvelles approches pour la segmentation acoustique de la parole et expose une nouvelle méthode basée sur la fusion de données pour l'apprentissage du système hybride MMC/PMC proposé. Dans [TAL 04], les auteurs proposent une méthode de classification non supervisée des images couleurs, basée sur l'utilisation des cartes auto-adaptatives de Kohonen et de la morphologie mathématique. Cette approche consiste en un premier temps à représenter la carte de Kohonen par la fdp sous jacente à l'échantillon d'observations représentatif des pixels d'une image dans l'espace 3D des composantes couleurs. Dans [SAT10], les auteurs développent une méthode améliorée de la segmentation à l'aide d'un réseau de neurone en utilisant la logique floue. le but est de détecter les divers tissus du cerveau tels que la matière blanche, grise et le liquide céphalo-rachidien ainsi que la tumeur. Dans [SEN 09], les auteurs présentent une étude sur la théorie de la détection de contours des images en utilisant une approche basée sur La logique floue, algorithme génétiques et les réseaux de neurones. Auto-organisation de Kohonen fonctionnalité carte (SOFM) est un outil puissant pour la classification [SUN 05]. Ji et le parc ait proposé un algorithme pour la segmentation des bassins versants basée sur SOM dans [MAU 07]. Cette méthode trouve le bassin versant de la composante de luminance de l'image couleur [DIN 07] [DIN 07]. La méthode s'explique comme suit. Elle se compose des réseaux de neurones indépendants un pour le plan de saturation et l'intensité [JAN 96] [MAE 08]. Le réseau de neurones est constitué de trois couches qui sont : couche d'entrée, couche cachée, et la couche de sortie.

7. Les avantages des réseaux de neurones

- *Souplesse* : les réseaux de neurones sont capables de traiter une gamme très étendue de problèmes. Leur résultat peut être une régression, une classification ou encore une analyse de clusters (clustering neuronal).
- *Bonne résolution* : ils donnent de bons résultats même dans des domaines complexes car ils sont beaucoup plus puissants que les statistiques ou les arbres de décisions.
- *Bonne adaptation* : une fois que les données sont codées, ils traitent aussi bien des variables continues qu'énumératives.

- *Parallélisme massif* : les réseaux de neurones sont constitués d'unités de calcul qui peuvent opérer d'une manière parallèle. La plupart des implémentations des réseaux de neurones peuvent être facilement converties d'une version séquentielle à une version parallèle.
- *Outils disponibles* : il existe de nombreux produits sur le marché intégrant la technique des réseaux de neurones (4thought, saxon, Neural connections, Clémentine, Intelligent Miner, SAS...).

8. Les points faibles des réseaux de neurones [NAK 98]

- *Codage des entrées* : toutes les entrées d'un réseau de neurone doivent se trouver dans un intervalle défini en général entre 0 et 1, ce qui entraîne des transformations qui impliquent des traitements supplémentaires, et risque de fausser les résultats.
- *Lisibilité* : les réseaux de neurones ne fournissent pas l'explication de leur résultat. Ce qui peut être gênant si on cherche à comprendre un phénomène.
- *Détermination de la taille* : afin que l'échantillon fournisse de bons résultats, sa taille doit être calculée en fonction du nombre d'entrée, du nombre des couches et du taux de connexion ce qui entraîne une augmentation du nombre d'exemple qui ne sont pas toujours disponibles.
- *Non optimalité* : plusieurs optima locaux ne donnent pas toujours la solution optimale en totalité.
- *Performance* : le nombre de calculs à effectuer pour définir un réseau optimal peut être très consommateur de puissance, ce qui peut donner de mauvaises performances à cette technique.

9. Applications

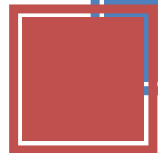
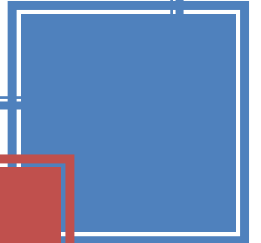
l'étude des réseaux de neurones est une voie prometteuse de l'Intelligence Artificielle, qui a des applications dans de nombreux domaines [RAC 11], [FAB 10].
 Industrie : contrôle qualité, diagnostic de panne, corrélations entre les données fournies par différents capteurs, analyse de signature ou d'écriture manuscrite...
 - Finance : prévision et modélisation du marché (cours de monnaies...), sélection d'investissements, Prévision du coût de la vie , attribution de crédits...

- Télécommunications et informatique : analyse du signal, élimination du bruit, reconnaissance de formes (bruits, images, paroles), compression de données...
- Environnement : évaluation des risques, analyse chimique, prévisions et modélisation météorologiques, gestion des ressources...
- Aérospatial : pilotage automatique, simulation du vol...
 - Automobile : système de guidage automatique,...
 - Défense : guidage de missile, suivi de cible, reconnaissance du visage, radar, sonar, lidar, traitement du signal, compression de données, suppression du bruit...
 - Electronique : prédiction de la séquence d'un code, vision machine, synthétiseur vocal, modèle non linéaire,...

10. Conclusion

Nous avons présenté dans cette partie des notions de base sur le neurone formel et le neurone biologique, ceci dans le but de bien connaître les caractéristiques d'un neurone qui nous intéresse, de même le principe de son fonctionnement. Par la suite nous avons exposé les différents types, les différentes topologies et les différents modes d'apprentissage existants des réseaux de neurones ainsi que les différents travaux utilisant ces réseaux en segmentation d'images, enfin, nous avons cité ses avantages, ses points faibles et leurs applications.

CHAPITRE 2



1. Introduction

Les algorithmes génétiques sont construits en s'inspirant du mécanisme de sélection naturelle de Darwin et des méthodes de combinaison de Mendel, pour résoudre des problèmes et affronter des situations difficiles tels que situation imprévue, environnement inconnu, contraintes changeantes induites par l'environnement.

Cette méthode travaille sur une population de nombreuses solutions potentielles, toutes différentes. Elle élimine les éléments les plus faibles pour favoriser la conservation et la reproduction des individus les plus "performants" (les plus "justes", les mieux adaptés). La recombinaison (reproduction par hybridation génétique) des individus les plus forts permet de donner naissance à des individus encore meilleurs d'une génération à la suivante. [2].

Pour résumer, Lerman et Nugouenet (1995) distinguent 4 principaux points qui font la différence fondamentale entre ces algorithmes et les autres méthodes :

1. Les algorithmes génétiques utilisent un codage des paramètres, et non les paramètres eux-mêmes.
2. Les algorithmes génétiques travaillent sur une population de points, au lieu d'un point unique.
3. Les algorithmes génétiques n'utilisent que les valeurs de la fonction étudiée, pas sa dérivée, ou une autre connaissance auxiliaire.
4. Les algorithmes génétiques utilisent des règles de transition probabilistes, et non déterministes.

2. Les concepts importants des algorithmes génétiques.

2.1 Définition

Les algorithmes génétiques (AGs) sont des algorithmes d'optimisation stochastique fondés sur les mécanismes de la sélection naturelle et de la génétique. Leur fonctionnement est extrêmement simple. On part avec une population de solutions potentielles (*chromosomes*) initiales arbitrairement choisies. On évalue leur performance (*fitness*) relative. Sur la base de ces performances on crée une nouvelle population de solutions potentielles en utilisant des opérateurs évolutionnaires simples : la sélection, le croisement et la mutation. On recommence ce cycle jusqu'à ce que l'on trouve une solution satisfaisante.

Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Pour l'utiliser, on doit disposer des cinq éléments suivants [3] :

1. *Un principe de codage de l'élément de la population.* Cette étape associe à chacun des points de l'espace d'état une structure de données. Elle se place généralement après une phase de modélisation mathématique du problème traité. La qualité du codage des données conditionne le succès des algorithmes génétiques. Le codage binaire a été très utilisé à l'origine. Les codages réels sont désormais largement utilisés, notamment dans les domaines applicatifs pour l'optimisation de problèmes à variables réelles.
2. *Un mécanisme de génération de la population initiale.* Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. Dans le cas où l'on ne connaît rien du problème à résoudre, il est essentiel que la population initiale soit répartie sur tout le domaine de recherche.
3. *Une fonction à optimiser.* Celle-ci retourne une valeur de \mathbb{R}^+ appelée *fitness* ou fonction d'évaluation de l'individu.
4. *Des opérateurs* permettant de diversifier la population au cours des générations et d'explorer l'espace d'état. L'opérateur de croisement recompose les gènes d'individus existant dans la population, l'opérateur de mutation a pour but de garantir l'exploration de l'espace d'états.
5. *Des paramètres de dimensionnement :* Taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation.

2.2 Génération

Une génération est une population à un instant (ou à une itération). Les algorithmes génétiques faisant évoluer les populations créent une nouvelle génération d'individus plus adaptée.

2.3 Population

Une population est un ensemble de chromosomes codant un ensemble de solutions potentielles du problème à résoudre à l'aide d'un mécanisme approprié du codage.

La taille de la population initiale est également laissée à l'appréciation du programmeur. Il n'est souvent pas nécessaire d'utiliser des populations démesurées. Une taille de 100 ou 150

individus s'avèrera souvent amplement suffisante, tant pour la qualité des solutions trouvées que pour le temps d'exécution de notre algorithme. Evidemment, ce n'est qu'un ordre de grandeur. Ensuite, on peut de modifier la taille de la population initiale en fonction du problème à résoudre si les solutions trouvées ne conviennent pas [KHA 05]

2.4 Chromosome (ou individu)

Nous appelons une séquence (chromosome, individu) A de longueur $l(A)$.

une suite $A = fa_{i1}; a_{i2}; \dots$; avec $\forall i \in [1, l] a_i \in V = \{0, 1\}$.

Un chromosome est donc une chaîne de caractères qui représente une solution possible dans l'espace des solutions. [2]. Chaque chromosome comprend un nombre de gènes égal aux variables à optimiser.

2.4.1 Codage des chromosomes

Le mécanisme de codage est fondamental pour la structure des algorithmes génétiques, il dépend de la nature des variables du problème à coder [SOU 04]. Chaque paramètre d'une solution est assimilé à un gène, toutes les valeurs qu'il peut prendre sont les allèles de ce gène. On doit trouver une manière de coder chaque allèle différent de façon unique (établir une bijection entre l'allèle "réel" et sa représentation codée). On peut par exemple choisir de regrouper les paramètres similaires dans un même chromosome (chromosome à un seul brin) et chaque gène sera repérable par sa position : son locus sur le chromosome en question. Chaque individu est représenté par un ensemble de chromosomes, et une population est un ensemble d'individus.

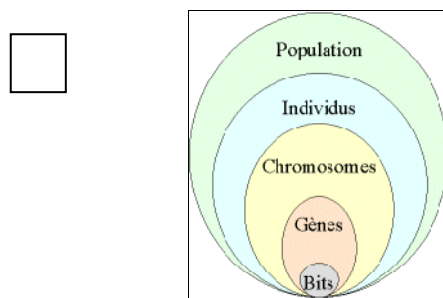


Figure 2.1: Les cinq niveaux d'organisation d'un algorithme génétique

Il y a trois principaux types de codage utilisables, et on peut passer de l'un à l'autre relativement facilement :

- a) **le codage binaire** : c'est le plus utilisé. Chaque gène dispose du même alphabet binaire $\{0, 1\}$, un gène est alors représenté par un entier long (32 bits), les chromosomes qui sont des suites de gènes sont représentés par des tableaux de gènes et les individus de notre espace de recherche sont représentés par des tableaux de chromosomes.
- b) **le codage réel** : cela peut-être utile notamment dans le cas où l'on recherche le maximum d'une fonction réelle.

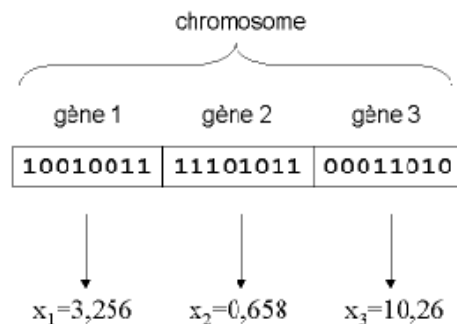


Figure 2.2 : Illustration schématique du codage des variables réelles.

- c) **le codage Gray** : dans le cas d'un codage binaire on utilise souvent la "distance de Hamming" comme mesure de la dis similarité entre deux éléments de population. Cette mesure compte les différences de bits de même rang de ces deux séquences. Et c'est la que le codage binaire commence à montrer ses limites. En effet, deux éléments voisins en terme de distance de Hamming ne codent pas nécessairement deux éléments proches dans l'espace de recherche. Cet inconvénient peut être évité en utilisant un "codage de Gray" : le codage Gray est un codage qui a comme propriété qu'entre un élément n et un élément $n + 1$, donc voisin dans l'espace de recherche, un seul bit diffère.

2.5 Gène

Dans les algorithmes génétiques un gène est une caractéristique (trait ou détecteur) défini par une valeur de caractéristique (Allèle) et une position dans le chromosome (Locus) [GOL 94]

2.6 Fonction d'adaptation (Fitness d'une séquence)

La *fonction d'adaptation* ou la *fonction d'évaluation* ou encor la *fonction fitness*, fournit une mesure relative à un ensemble particulier de paramètres qui codent le chromosome, c à d si on considère que x_1, x_2, \dots, x_n des caractères ou paramètres qui constituent un chromosome i et

f la fonction d'adaptation de ce chromosome, alors $f(x_1, x_2, \dots, x_p)$ donne une valeur f_i pour le chromosome i . La valeur f_i dite valeur d'adaptation, représente toujours une performance de ce chromosome, c à d une valeur qui distingue les chromosomes qui représentent les solutions les plus proche de la solution optimale à ceux qui représentent les mauvaises solutions du problème. [WHI 93] [MAN 95].

Nous appelons donc *fitness* d'une séquence toute valeur positive que nous noterons $f(A)$, où f est typiquement appelée fonction de fitness. Les chromosomes ou individus les plus adaptés possèdent des valeurs d'adaptation importantes.

2.7 Sélection

La sélection est le choix des individus pour la reproduction. Cet opérateur est chargé de définir quels seront les individus de P qui vont être dupliqués dans la nouvelle population P' et vont servir de parents (application de l'opérateur de croisement). Soit n le nombre d'individus de P , on doit en sélectionner $n/2$ (l'opérateur de croisement nous permet de repasser à n individus).

L'opérateur de la sélection peut-être le plus important puisqu'il permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement reliée à son efficacité relative au sein de la population. [SOU 04]. Il existe plusieurs techniques de sélection. Voici les plus utilisées [4]. :

2.7.1 Sélection par rang.

Cette technique de sélection choisit toujours les individus possédant les meilleurs scores d'adaptation, le hasard n'entre donc pas dans ce mode de sélection. En fait, si n individus constituent la population, la sélection appliquée consiste à conserver les k meilleurs individus (au sens de la fitness) suivant une probabilité qui dépend du rang (et pas de la fitness).

2.7.2 Probabilité de sélection proportionnelle à l'adaptation :

(Appelé aussi *roulette* ou roue de la *fortune*), Pour chaque individu, la probabilité d'être sélectionné est proportionnelle à son adaptation au problème. Afin de sélectionner un individu, on utilise le principe de la roue de la fortune biaisée. Cette roue est une roue de la fortune classique sur laquelle chaque individu est représenté par une portion proportionnelle à son adaptation. On effectue ensuite un tirage au sort homogène sur cette roue.

2.7.3 Sélection par tournoi

Cette technique utilise la sélection proportionnelle sur des paires d'individus, puis choisit parmi ces paires l'individu qui a le meilleur score d'adaptation.

2.7.4 Sélection uniforme

La sélection se fait aléatoirement, uniformément et sans intervention de la valeur d'adaptation. Chaque individu a donc une probabilité $1/P$ d'être sélectionné, où P est le nombre total d'individus dans la population.

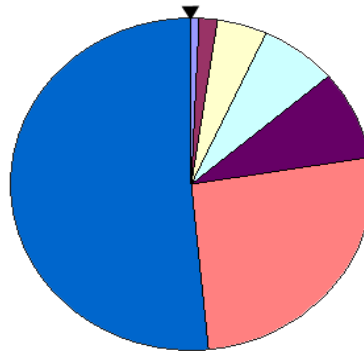


Figure 2.3 : Schéma d'une roulette

3. Opérateurs des algorithmes génétiques

Les algorithmes génétiques utilisent trois mécanismes (opérateurs) pour opérer sur des populations de chaînes (chromosome) : la reproduction la mutation et le croisement ou recombinaison (crossover). Ces opérateurs sont appliqués aux générations de chaînes successives afin de créer de nouvelles populations de chaînes. Ils sont la simplicité elle-même, ne mettant en jeu des procédures aussi peu complexes que la génération de nombres aléatoires, la copie de chaînes ; cependant malgré leur simplicité la performance de l'optimisation qui en résulte est impressionnante. [GOL 94].

3.1 La reproduction

La reproduction est un procédé dans lequel on utilise une fonction de fitness (fonction d'adaptation ou d'adéquation) pour assurer des duplications d'individus qui satisfont certains critères, seuls les mieux performants seront conservés. Intuitivement cette fonction d'adaptation peut être envisagée comme une mesure que l'on souhaite optimiser. Dupliquer

des chaînes (individus) en fonction des valeurs de leur fonction d'adaptation revient à donner aux chaînes dont la valeur est optimale une probabilité plus élevée à contribuer à la génération suivante.

3.2 La mutation

La mutation modifie aléatoirement, avec une certaine probabilité, la valeur d'un gène ou de plusieurs gènes d'un chromosome ou individu. Elle joue le rôle de bruit et empêche l'évolution de se figer. Elle permet de diversifier les solutions en alternant de façon non déterministe un individu et d'assurer une recherche aussi bien globale que locale.

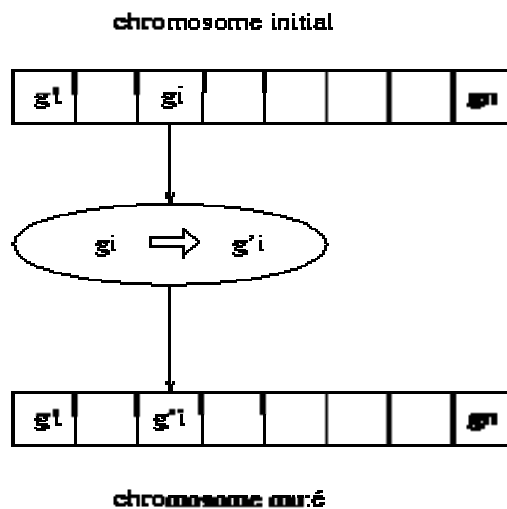


Figure 2.4 : Principe de l'opérateur de mutation

Pour les problèmes discrets, l'opérateur de mutation consiste généralement à tirer aléatoirement un gène dans le chromosome et à le remplacer par une valeur aléatoire. Si la notion de distance existe, cette valeur peut être choisie dans le voisinage de la valeur initiale [3].

Pour résumer l'opérateur dispose de 4 grands avantages [SOU 04] :

- Il garantit la diversité de la population, ce qui est primordial pour les algorithmes génétiques.
- Il permet d'éviter un phénomène connu sous le nom de *dérive génétique*. On parle de dérive génétique quand certains gènes favorisés par le hasard se répandent au détriment des autres et sont ainsi présents au même endroit sur tous les chromosomes. Le fait que l'opérateur de mutation puisse entraîner de manière aléatoire des changements au niveau de n'importe quel locus permet d'éviter l'installation de cette situation défavorable.

- Il permet de limiter les risques d'une convergence prématurée causée par exemple par une méthode de sélection élitiste imposant à la population une pression sélective trop forte. En effet, dans le cas d'une convergence prématurée on se retrouve avec une population dont tous les individus sont identiques mais ne sont que des optimums locaux. Tous les individus étant identiques, le croisement ne changera rien à la situation. En effet, l'échange d'informations par crossover entre des individus strictement identiques est bien sûr totalement sans conséquences; on aura beau choisir la méthode de croisement qu'on veut on se retrouvera toujours à échanger des portions de chromosomes identiques et la population n'évoluera pas. L'évolution se retrouvant bloquée on n'attendra jamais l'optimum global. La mutation entraînant des inversions de bits de manière aléatoire permet de réintroduire des différences entre les individus et donc de nous extirper de cette situation. Il est quand même utile de garder à l'esprit que ceci n'est pas une solution "miracle" et qu'il est bien entendu plus intelligent de ne pas utiliser de méthodes de sélection connues pour entraîner ce type de problème.
- La mutation permet d'atteindre la propriété d'ergodicité. L'ergodicité est une propriété garantissant que chaque point de l'espace de recherche puisse être atteint. En effet, une mutation pouvant intervenir de manière aléatoire au niveau de n'importe quel locus, on a la certitude mathématique que n'importe quel permutation de notre chaîne de bits peut apparaître au sein de la population et donc que tout point de l'espace de recherche peut être atteint. Grâce à cette propriété on est donc sûr de pouvoir atteindre l'optimum global.

On notera que la mutation règle donc le problème exposé à la fin du Section sur le croisement [SOU 04].

3.3 Croisement

Le croisement a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes. Classiquement, les croisements sont envisagés avec deux parents et génèrent deux enfants. Initialement, le croisement associé au codage par chaînes de bits est le croisement à découpage de chromosomes (slicing crossover). Pour effectuer ce type de croisement sur des chromosomes constitués de M gènes, on tire aléatoirement une position dans chacun des parents. On échange ensuite les deux sous-chaînes terminales de chacun des deux chromosomes, ce qui produit deux enfants C_1 et C_2 [15] (cf. Figure 2.5).

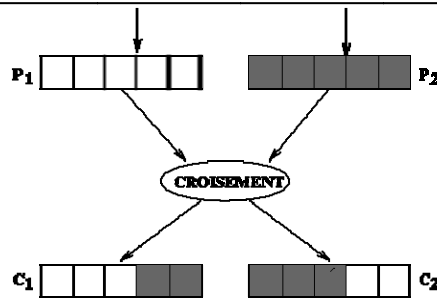


Figure 2.5 Slicing Croisement.

On peut étendre ce principe en découpant le chromosome non pas en 2 sous-chaînes mais en 3, 4, etc. . (Cf. Figure 2.6).

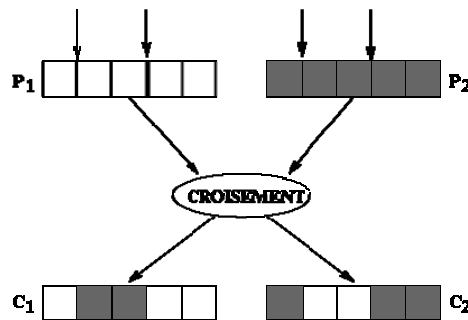
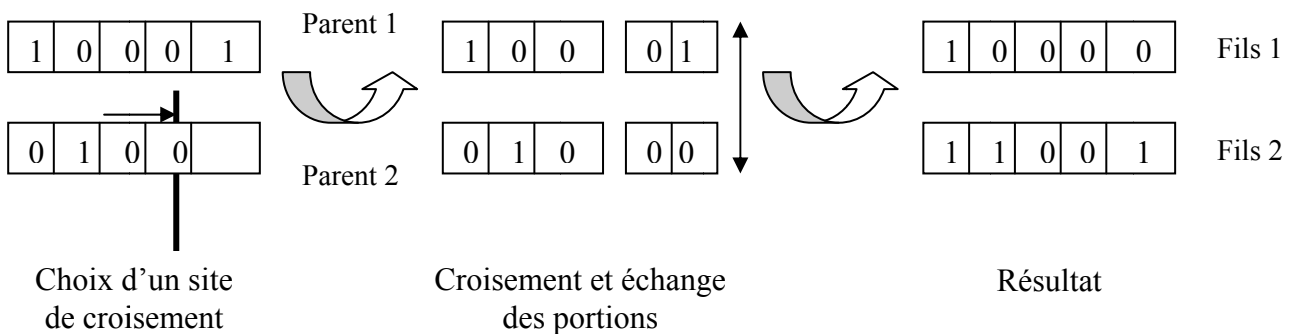


Figure 2.6 Slicing Croisement à 2 points

Le croisement donc permet la création de nouveaux individus selon un processus fort simple qui consiste à sélectionner deux chromosomes aléatoirement (parents) ensuite choisir un point identique pour les deux chromosomes pour les coupées, et intervertit les deux premières portions ou les derniers des deux chromosomes coupés. De ce fait on parvient à avoir une génération de plus en plus adaptée avec des qualités meilleures car seuls les individus suffisamment adéquats vivent longtemps pour se reproduire [JEA 91].

Le croisement peut être effectué en un point ou en plusieurs points. Nous présentons un exemple de croisement en un point.



Cite de Figure 2.7 Croisement binaire en un point.

4. Fonctionnement de l'algorithme génétique

L'algorithme génétique et par une autre approche itérative recherche une autre génération à partir de l'ancienne, jusqu'à l'obtention d'une génération qui est mieux adaptées au problème à résoudre. Nous présentons l'algorithme général.

4.1 Algorithme général

Les AGs sont alors basés sur les phases suivantes :

1. **Initialisation.** Une population initiale de N chromosomes est tirée aléatoirement.
2. **Évaluation.** Chaque chromosome est décodé, puis évalué.
3. **Sélection.** Création d'une nouvelle population de N chromosomes par l'utilisation d'une méthode de sélection appropriée.
4. **Reproduction.** Possibilité de croisement et mutation au sein de la nouvelle population.
5. **Retour** à la phase d'évaluation jusqu'à l'arrêt de l'algorithme.

Initialisation : générer un ensemble de solutions initiales.

Boucle principale :

I. Evolution

- **Sélection** : choisir avec une probabilité proportionnelle à leur qualité une liste d'individus (chromosomes).
- **Reproduction** : générer à partir de cette liste de nouveaux individus à l'aide des opérateurs génétiques.
- **Remplacement** : Eliminer avec une probabilité inversement proportionnelle à leur qualité certains individus.

II. Réactualisation de la meilleure solution

III. Aller à 1) tant que nombre de générations est inférieur à la valeur prédéterminée.

L'algorithme décrit précédemment représente l'algorithme de base, mais bien d'autres variantes existent. Chaque utilisateur conçoit en général son propre algorithme génétique suivant sa propre application en ajoutant quelques caractéristiques qui en améliorent l'efficacité tout en conservant les principes de base.

Le principe général du fonctionnement d'un algorithme génétique est représenté sur la figure 5.8 : on commence par générer une population d'individus de façon aléatoire. Pour passer d'une génération k à la génération $k+1$, les trois opérations suivantes sont répétées pour tous les éléments de la population k . Des couples de parents $P1$ et $P2$ sont sélectionnés en fonction de leurs adaptations. L'opérateur de croisement leur est appliqué avec une probabilité P_c

(généralement autour de 0.6) et génère des couples d'enfants C1 et C2. D'autres éléments P sont sélectionnés en fonction de leur adaptation. L'opérateur de mutation leur est appliqué avec la probabilité P_m (P_m est généralement très inférieur à P_c) et génère des individus mutés P0. Le niveau d'adaptation des enfants (C1, C2) et des individus mutés P0 sont ensuite évalués avant insertion dans la nouvelle population. Différents critères d'arrêt de l'algorithme peuvent être choisis [JEA 05]

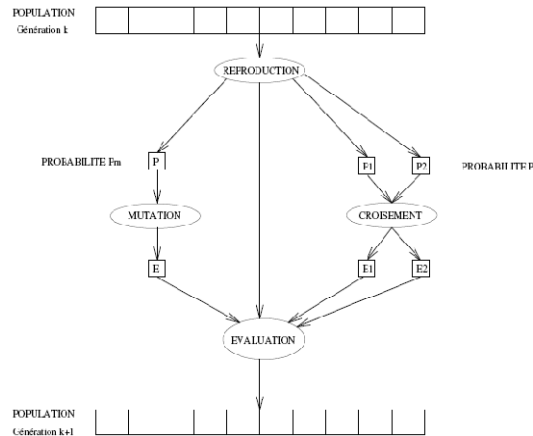


Figure 2.8 Principe général des algorithmes génétiques

4.2 Cycle complet d'un algorithme génétique

4.2.1 La création de la population initiale : Pour démarrer un algorithme génétique, il faut lui fournir une population à faire évoluer. La manière dont le programmeur va créer chacun des individus de cette population est entièrement libre. Il suffit que tous les individus créés soient *de la forme d'une solution potentielle*, et il n'est nullement besoin de songer à créer des bons individus. Ils doivent juste fournir une réponse, même mauvaise, au problème posé.

4.2.2 L'évaluation des individus : Une fois que la population initiale a été créée, nous allons en sortir les individus les plus prometteurs, ceux qui vont participer à l'amélioration de notre population. Nous allons donc attribuer une 'note' ou un indice de qualité à chacun de nos individus. La méthode d'évaluation des individus est laissée au programmeur en fonction du problème qu'il a à optimiser ou à résoudre. Cette étape intermédiaire d'évaluation peut même devenir une étape importante du processus d'amélioration de notre population. En effet, les différents individus ne sont pas toujours comparables, il n'est pas toujours possible de dire qu'un individu est meilleur ou moins bon qu'un autre. C'est le cas des problèmes multicritères, lorsqu'une solution dépend de plusieurs paramètres. Vous pourrez toujours dire qu'un

nombre est supérieur ou non à un autre, mais vous ne pourrez pas dire si un vecteur est supérieur ou non à un autre. La notion de supériorité pour les vecteurs n'existe pas. Vous pouvez comparer leur norme, mais pas les vecteurs eux-mêmes.

- 4.2.3 La création de nouveaux individus :** Nous allons maintenant enrichir notre population en croisant des individus. Nous allons essayer de prendre des morceaux de solution de certains individus et d'autres morceaux d'autres individus pour créer de nouveaux individus qui, on l'espère, seront de meilleures solutions à notre problème. Pour ce faire, il faut choisir une des méthodes de sélection citée au par avant.
- 4.2.4 Les croisements :** permettent de simuler des reproductions d'individus dans le but d'en créer de nouveaux. Il est tout à fait possible de faire des croisements aléatoires. Toutefois, une solution largement utilisée est d'effectuer des croisements multipoints.
- 4.2.5 Les mutations :** Une autre solution que le croisement pour créer de nouveaux individus est de modifier ceux déjà existants. Une fois de plus, le hasard va nous être d'une grande utilité. Il peut s'avérer efficace de modifier aléatoirement quelques individus de notre population en en modifiant un gène ou un autre. Rien ne nous dit que l'individu muté sera meilleur ou moins bon, mais il apportera des possibilités supplémentaires qui pourraient bien être utiles pour la création de bonnes solutions. De même que pour les croisements, il n'est pas recommandé de faire muter tous les individus. Il est possible de faire muter un individu de la manière qu'il vous plaira. Une seule contrainte : l'individu muté doit être de la forme d'une solution potentielle. Généralement, on ne modifie qu'un gène pour passer d'une solution à une autre solution de forme similaire mais qui peut avoir une évaluation totalement différente.
- 4.2.6 L'insertion des nouveaux individus dans la population :** Une fois que nous avons créé de nouveaux individus que ce soit par croisement ou par mutation, il nous faut sélectionner ceux qui vont continuer à participer à l'amélioration de notre population. Une fois encore, libre au programmeur de choisir ceux qu'il souhaite conserver. Il est possible de refaire une étape d'évaluation des individus nouvellement créés. De même qu'il est possible de conserver tous les nouveaux individus en plus de notre population. Il n'est toutefois pas recommandé de ne conserver que les nouveaux individus et d'oublier la population de travail. En effet, rien ne nous dit que les nouveaux individus sont meilleurs que les individus de départ. Une méthode relativement efficace consiste à insérer les nouveaux individus dans la population, à

trier cette population selon l'évaluation de ses membres, et à ne conserver que les N meilleurs individus.

- 4.2.7 Choix du nombre N d'individus à conserver :** Le nombre d'individus N à conserver est à choisir avec soin. En prenant un N trop faible, la prochaine itération de l'algorithme se fera avec une population plus petite et elle deviendra de plus en plus petite au fil des générations - elle pourrait même disparaître. En prenant un N de plus en plus grand, nous prenons le risque de voir exploser le temps de traitement puisque la population de chaque génération sera plus grande. Une méthode efficace est de toujours garder la même taille de population d'une génération à l'autre, ainsi il est possible de dérouler l'algorithme sur un grand nombre de générations.
- 4.2.8 Passage à la génération suivante :** Une fois la nouvelle population obtenue, vous pouvez recommencer le processus d'amélioration des individus pour obtenir une nouvelle population et ainsi de suite ...
- 4.2.9 Réitération du processus :** Le programmeur a l'opportunité d'évaluer les individus de sa population avant et/ou après les phases de création d'individus. En effet, il peut s'avérer pertinent de les évaluer avant de les insérer dans la future population, de même qu'il peut être utile de les réévaluer au début de la génération suivante, si par exemple la méthode d'évaluation dépend de la taille de la population (qui a très bien pu changer). Ainsi on peut être amené à évaluer deux fois par génération chacun des individus. Le nombre de génération est aussi laissé à l'appréciation du programmeur. Généralement il n'est pas possible de trouver des solutions convenables en moins de 10 générations et au bout de 500 générations, les solutions n'évoluent plus. la recherche d'un optimum multicritère, il est rare d'obtenir une solution unique. En effet, les réponses optimales à notre problème peuvent provenir de valeurs largement différentes tant au niveau des données d'entrée (les gènes) que des données de sortie (les critères d'évaluation). Ainsi, l'on considère généralement que l'on obtient un ensemble de solutions optimales, dites "co-dominantes" : toute amélioration d'une solution selon l'un des critères entraînera obligatoirement une dégradation de l'évaluation des autres critères

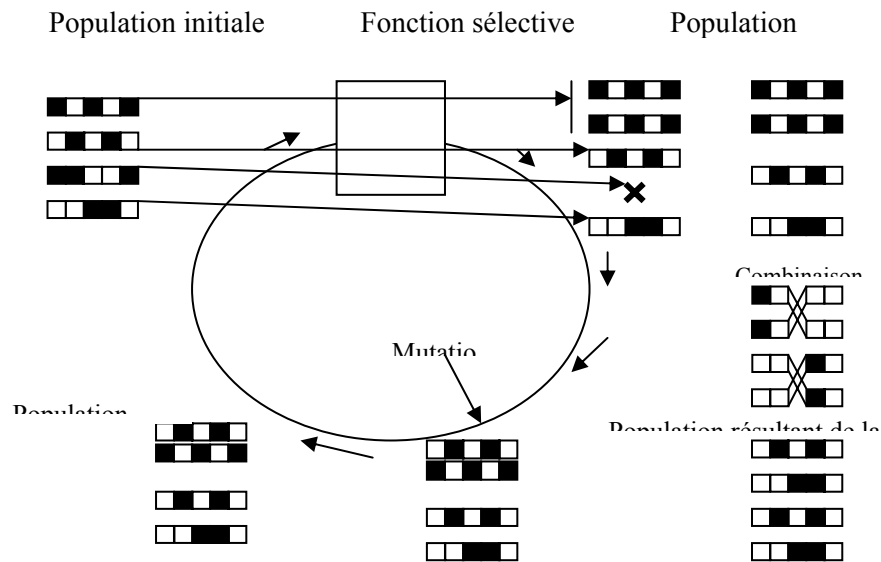


Figure 2.9 Cycle complet d'un algorithme génétique

5. Les algorithmes génétiques dans la segmentation des images [NAD 04]

S. Bhandarkar et auteurs considèrent le problème de la détection de contours comme un problème d'optimisation d'une fonction objective de coût en utilisant les AGs. Cette fonction initialement proposée par Tan et al [1991 ; 1989] utilise les informations sur les contours et les valeurs des niveaux de gris des régions. En effet, Tan et al ont défini le problème de la détection de contours comme étant un problème de minimisation de coût. Il a essayé de surmonter les imperfections des techniques existantes (Gonzalez, 1987) en formulant une définition d'un contour qui est assez générale pour inclure la plupart des types de contours. Leur travail a permis également d'améliorer les techniques existantes en considérant explicitement la structure locale du contour avec ses pixels voisins. Par la minimisation de la fonction coût, les auteurs exigent que les contours détectés soient fins, continus, longs et d'une très bonne localisation. En effet, l'image de contours est obtenue par le calcul du coût qui est fonction de la dissemblance (ou dissimilitude) de la région, de la continuité, de la fragmentation et de la largeur des structures locales de contours. L'algorithme proposé est basé sur une technique simple de recherche locale, il compare localement deux configurations de contours qui sont différentes uniquement en un point, et de sélectionner la meilleure.

Bhandarkar et al [1994 ; 1999] ont proposé d'associer à leur approche les AGs pour la comparaison des structures de contours valides et tous les points contours (considérés sur un

voisinage de 3x3) de l'image en utilisant une fonction fitness basée sur les facteurs de coûts. Aussi, Bhanu et al [1991 ; 1995] ont construit un système de segmentation de scènes d'extérieur en couleur qui optimise, par AG, la segmentation de chacune des images qui lui sont présentées tout en gardant une mémoire de son expérience passée dans une base de règles du type 'condition/action'. Le système repose sur l'utilisation d'un AG de segmentation paramétrable et sur une fonction d'évaluation de la qualité des segmentations réalisées. La partie 'action' de chacune des règles code sous forme binaire, un jeu de paramètres de l'algorithme de segmentation. La partie 'condition' stocke les caractéristiques d'une image ainsi que l'heure et les conditions climatiques au moment de son acquisition. Lorsqu'une image est présentée au système, elle active les règles porteuses de parties 'condition' correspondant à cette image. Les parties 'action' de ces règles sont alors soumises à un AG dont l'objectif est de générer une paramétrisation conduisant à une segmentation de qualité satisfaisante, au sens de la fonction retenue a priori. Le nouveau jeu de paramètres ainsi obtenu est alors stocké, avec les caractéristiques de l'image, dans une règle de la base du système.

Dans l'approche développée par Chun et Yang [1996], ce ne sont pas des configurations de contours qui évoluent mais des configurations de régions. Une partition initiale en un grand nombre de régions de taille éventuellement réduite est initialement obtenue par un algorithme de clustering. Un AG est ensuite utilisé pour améliorer cette segmentation en simulant des opérations itérées de fusion et de division des régions obtenues. Un chromosome porte autant de gènes qu'il y a de régions initiales et les allèles sont les labels affectés aux régions dans la segmentation pour laquelle il code. Un chromosome dans lequel des allèles des gènes i et j sont identiques spécifie une fusion des deux régions. Une mutation qui modifie l'un de ces deux allèles génère une séparation des deux régions.

P. Andrey [1997] a proposé l'algorithme de relaxation sélectionniste qui est un algorithme génétique non supervisé de segmentation des images à plusieurs niveaux de gris qui nous fournit pour une même exécution l'image de contours et l'image de régions.

6. Les algorithmes génétiques associés aux réseaux de neurones pour la segmentation

Dokur et Olmez [2003] ont proposé un réseau de neurones quantifié QNN pour la segmentation des images à résonance magnétique MR et des images topographiques CT (Computer Topography). Les éléments du vecteur caractéristique sont formés des intensités des pixels voisins du point considéré. QNN est une structure d'un réseau de neurones conçu par

apprentissage avec les AGs. Chaque noeud de la première couche du QNN forme un hyperplan HP dans l'espace d'entrée. Il y a une contrainte sur les HPs dans les réseaux QNN. Le HP est représenté uniquement par un paramètre dans un espace d'entrée à N dimension. Les AGs sont utilisés pour trouver les valeurs optimales des paramètres des noeuds de ces réseaux QNN. Une comparaison pour la segmentation des images MR et des images CT en utilisant le perceptron multicouche et le réseau de kohonen a été donnée. Il a été observé que le QNN présente de meilleures performances de classification en utilisant peu de noeuds avec un temps d'apprentissage court.

Matsui et Kosugi [1999] proposent une méthode de segmentation des images par les réseaux de neurones avec une sélection des paramètres basée sur les AGs. Ces derniers sont utilisés pour la sélection des combinaisons optimales des paramètres d'entrée du réseau de neurones classificateur. L'AG proposé est basé sur l'évaluation d'une fonction objective, qui permet d'estimer la combinaison optimale des paramètres des indices rapidement sans tester les classificateurs actuels. Les deux types de réseaux de neurones utilisés sont : le réseau de neurones par rétro propagation et le réseau de neurones à compteur propagation modifié (modified counter-propagation network).

7. Domaines d'application des algorithmes génétiques

D'une façon générale, Les applications des algorithmes génétiques sont multiples peuvent être utilisés pour :

- L'optimisation des fonctions numériques difficiles (discontinues, multimodales, bruitées).
- Le traitement d'images.
- Le contrôle des systèmes industriel
- Apprentissage des réseaux de neurones...etc.

8. Les avantages des algorithmes génétiques [NAK98]

- Equilibre entre exploration et exploitation de l'espace de recherche.
- Ils sont performants pour optimiser les réseaux de neurones.
- Ils peuvent traiter une large gamme de types de données.
- Les résultats sont explicites et faciles à analyser.
- L'utilisation minimale de l'information à priori.
- Ils s'adaptent bien aux problèmes d'optimisation.
- Le parallélisme.

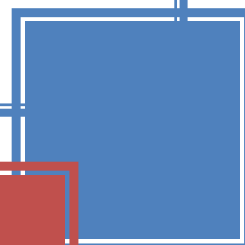
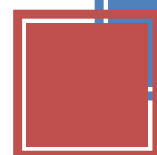
9. Limitations des algorithmes génétiques [NAK98] [FER98] [5]

- L'influence de certains paramètres sur le résultat final (probabilité de croisement et de mutation).
- *Le temps de calcul* : ils nécessitent de nombreux calculs, en particulier au niveau de la fonction d'évaluation, (taille du chromosome très grande).
- Ils sont le plus souvent *difficiles à mettre en œuvre* : des paramètres comme la taille de la population ou le taux de mutation sont parfois difficiles à déterminer. Or le succès de l'évolution en dépend et plusieurs essais sont donc nécessaires, ce qui limite encore l'efficacité de l'algorithme. En outre, choisir une bonne fonction d'évaluation. Celle-ci doit prendre en compte les bons paramètres du problème. Elle doit donc être choisie avec soin.
- Il faut aussi noter l'impossibilité d'être *assuré*, même après un nombre important de générations, que la solution trouvée soit la meilleure. On peut seulement être sûr que l'on s'est approché de la solution optimale (pour les paramètres et la fonction d'évaluation choisie), sans la certitude de l'avoir atteinte.
- Un autre problème important est celui des *optima locaux*. En effet, lorsqu'une population évolue, il se peut que certains individus qui à un instant occupent une place importante au sein de cette population deviennent majoritaires. À ce moment, il se peut que la population converge vers cet individu et s'écarte ainsi d'individus plus intéressants mais trop éloignés de l'individu vers lequel on converge. Pour vaincre ce problème, il existe différentes méthodes comme l'ajout de quelques individus générés aléatoirement à chaque génération, des méthodes de sélection différentes de la méthode classique ...

10. Conclusion

Dans ce chapitre, une description des notions de bases reliées aux algorithmes génétiques est présentée, notamment les opérateurs : sélection, mutation, croisement et reproduction ainsi que le fonctionnement de base de l'algorithme. Les AGs ont fait leurs preuves concernant la résolution de certains problèmes d'optimisation complexes. De grandes classes d'applications sont envisagées à ce jour par utilisation des AGs. Nous avons présenté différents travaux effectués dans la segmentation des images ainsi que la combinaison des algorithmes génétiques avec les réseaux de neurones pour la segmentation. À la fin, nous avons discuté les apports, les limitations et les domaines d'application des Algorithmes génétiques.

CHAPITRE 3



1. Introduction

La plupart des problèmes rencontrés sont modélisables mathématiquement. Mais ces modèles nécessitent des hypothèses parfois trop restrictives, rendant délicate l'application au monde réel. Les problèmes du monde réel doivent tenir compte d'informations imprécises et incertaines.

La logique floue généralise la logique classique avec des variables logiques et des formules logiques prenant des degrés de valeur de vérité quelconques entre 0 (faux) et 1 (vrai) inclusivement. La logique classique avec ses valeurs de vérité booléennes de 0 et 1 est considérée comme un cas particulier de la logique floue [BEN 07].

2. La logique floue

Le terme d'ensemble flou apparaît pour la première fois en 1965 lorsque le professeur Lotfi A. Zadeh, de l'université de Berkeley aux USA, publie un article intitulé « Ensembles flous » (Fuzzy sets). Il a réalisé depuis de nombreuses avancées théoriques majeures dans le domaine et a été rapidement accompagné par de nombreux chercheurs développant des travaux théoriques. La logique floue s'appuie sur la théorie mathématique des ensembles flous. C'est une théorie formelle et mathématique dans le sens où Zadeh, en partant du concept de fonction d'appartenance pour modéliser la définition d'un sous-ensemble d'un univers donné, a élaboré un modèle complet de propriétés et de définitions formelles. Il a aussi montré que cette théorie des sous-ensembles flous se réduit effectivement à la théorie des sous-ensembles classiques dans le cas où les fonctions d'appartenance considérées prennent des valeurs binaires ($\{0,1\}$).

3. La théorie des sous ensembles flous

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

Les sous-ensembles flous (ou parties floues) ont été introduits afin de modéliser la représentation humaine des connaissances, et ainsi améliorer les performances des systèmes de décision qui utilisent cette modélisation. Les sous-ensembles flous sont utilisés soit pour modéliser l'incertitude et l'imprécision, soit pour représenter des informations précises sous forme lexicale assimilable par un système expert [5][STR 05].

3.1 Les fonctions d'appartenance

On peut donner un coefficient de confiance à l'affirmation « X appartient à un ensemble A », par exemple : le coefficient d'appréciation de « l'air à la température égale à 30°C est chaud » vaut 0.6, ce qui signifie que cette température correspond à « plutôt chaude ». On peut pour toute température, donc pour tout X, définir ce coefficient directement à X.

Cette propriété se présente facilement par une fonction dite d'appartenance $\mu_A(X)$ à valeurs dans $[0,1]$, la notion signifie « coefficient d'appartenance de X à l'ensemble caractérisé par A », l'argument X se rattache à la variable linguistique et l'indice A désigne l'ensemble concerné.

De la même manière, une variable Y appartiendra à un ensemble B avec une fonction d'appartenance notée $\mu_B(Y)$, par exemple « le vent est fort ». On peut associer X et Y dans une même phase, par exemple l'ensemble C : « l'air est chaud et le vent est fort ». La variable Z définie par : « air chaud et vent fort » correspond à l'intersection de « air est chaud » et de « vent est fort ». L'ensemble C correspond à l'intersection des ensembles A et B. la valeur de $\mu_C(Z)$ se déduit des valeurs de $\mu_A(X)$ et $\mu_B(Y)$. Il existe diverses solutions pour traduire mathématiquement le problème.

Un fait certain aura une fonction d'appartenance égale à 1 pour le point de fonctionnement considéré. Un fait incertain aura une fonction d'appartenance inférieure ou égale à 1. Lorsque le fait certain correspond à l'énoncée de la valeur d'une variable $X=X_0$ on aura $\mu_{X_0}(X_0) = 1$ pour $X=X_0$ et $\mu_{X_0}(X) = 0$ pour X différent de X_0 : on a un singleton. Un fait incertain tel que x à peu près égale à X_0 : aura une fonction d'appartenance en forme de triangle. L'affirmation de X à peu près compris entre X_1 et X_2 correspond à une fonction trapézoïdale [9]. Les fonctions d'appartenance peuvent avoir diverses formes selon leur définition : triangulaire, trapézoïdale, Gaussienne, Sigmoides...

3.2 Les caractéristiques d'un sous ensemble flou

Un ensemble flou F de l'univers X est caractérisé par :

3.2.1 Le noyau : Noté $\text{noy}(F)$, qui représente l'ensemble des éléments de X pour lesquels la fonction d'appartenance $\mu_F(x)$ vaut 1 :

$$\text{noy}(F) = \{x \in X / \mu_F(x) = 1\} \quad 3.1$$



3.2.2 Le support : Noté $\text{supp}(F)$, qui représente l'ensemble des éléments de X appartenant, même très peu, à F , c.-à-d., ayant $\mu_F(x)$ qui n'est pas nulle :

$$\text{supp}(F) = \{x \in X / \mu_F(x) \neq 0\} \quad 3.2$$

3.2.3 La hauteur: Notée $h(F)$, qui représente la plus grande valeur prise par sa fonction d'appartenance :

$$h(F) = \sup \mu_F(x) , x \in X \quad 3.3$$

3.2.4 α -coupe: Qui représente l'ensemble contenant les éléments ayant un degré d'appartenance Supérieur ou égal à a : [BNE 05]

$$\alpha\text{-coupe}(F) = \{x \in X / \mu_F(x) \geq a\} \quad 3.4$$

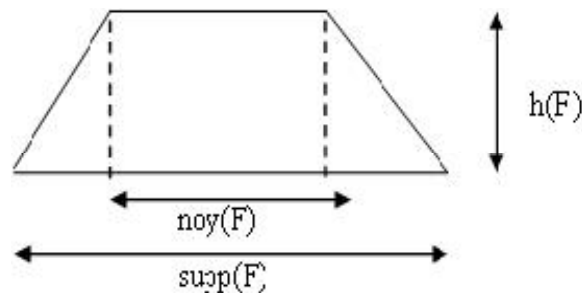


Figure 3.1 : le noyau, le support, la hauteur d'un sous ensemble flou

3.3 Les opérations ensemblistes

Il existe de nombreuses variantes dans ces opérateurs. Cependant, les plus répandus sont ceux dits « de Zadeh » décrits ci-dessous. Leur utilisation sera reprise dans l'exemple didactique d'utilisation d'une base de règles floues.

Dans ce qui suit, le degré de vérité d'une proposition A sera noté $\mu(A)$.

3.3.1 La réunion

L'opérateur logique correspondant à l'union d'ensembles est le OU. Le degré de vérité de la proposition « A OU B » est le maximum des degrés de vérité de A et de B .

Exemple 1 :

- A est l'ensemble flou des personnes petites.
- B est l'ensemble flou des personnes moyennes.

L'ensemble des personnes petites OU moyennes est un ensemble flou de fonction d'appartenance :

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \quad \forall x \in U \quad 3.5$$



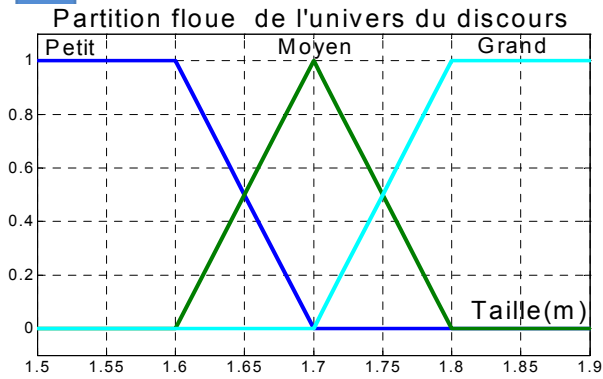


Figure 3.2 : Partition floue de l'univers du discours

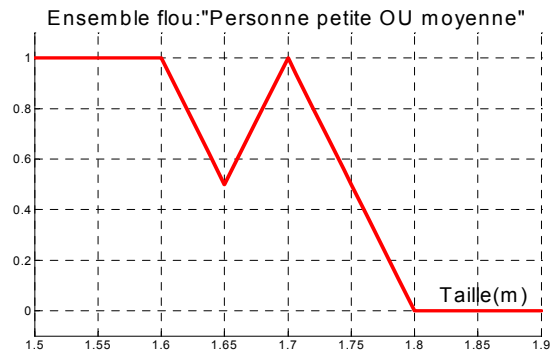


Figure 3.3 : Ensemble flou « la réunion »

3.3.2 L'intersection

L'opérateur logique correspondant à l'intersection d'ensembles est le ET. Le degré de vérité de la proposition « A ET B » est le minimum des degrés de vérité de A et de B.

Exemple 2 :

-A est l'ensemble flou des personnes petites. B est l'ensemble flou des personnes moyennes. L'ensemble des personnes petites ET moyennes est un ensemble flou de fonction d'appartenance :

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \quad \forall x \in U$$

3.6

3.3.3. Le complément

L'opérateur logique correspondant au complément d'un ensemble est la négation

Exemple 3 :

A est l'ensemble flou des personnes petites. L'ensemble des personnes NON petites est un ensemble flou de fonction d'appartenance : [CHE 06].

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad \forall x \in U$$

3.7

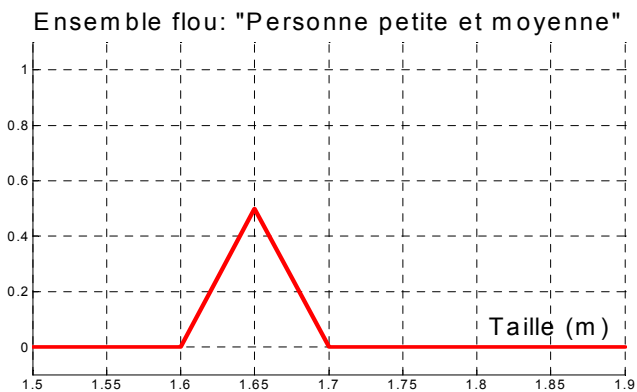


Figure 3.4 : Ensemble flou « l'intersection »
« complément »

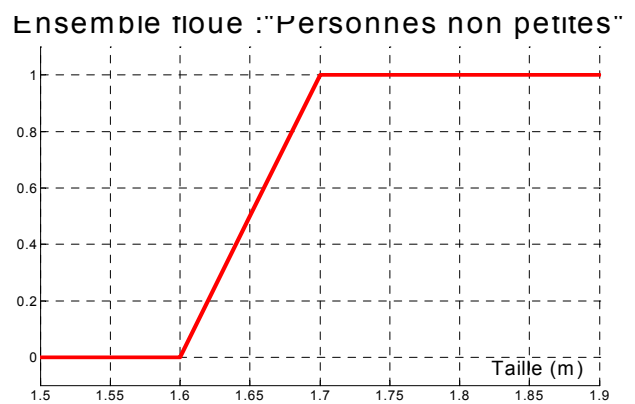


Figure 3.5 : Ensemble flou



3.4 Différentes représentations de sous-sembles flous

Appellation	ET	OU	NON
Zadeh	$\min(x, y)$	$\text{Max}(x, y)$	$1-x$
Pro balistique	xy	$x \div y - xy$	$1-x$
Lukasiewicz	$\text{Max}(x+y-1, 0)$	$\min(x+y, 1)$	$1-x$
Ha... ($\beta > 0$)	$xy / (\beta + (1 - \beta)(x+y \cdot xy))$	$(x \div y + xy - (1 - \beta)xy) / (1 - (1 - \beta)xy)$	$1-x$
Weber	x si $y=1$, y si $x=1$	x si $y=0$, y si $x=0$ 1 sinon	$1-x$

Tableau 3.1 : Différentes représentations de sous-sembles flous.

3.5 Les relations floues

Le concept de relation floue généralise celui de relation classique. Il met en évidence des liaisons imprécises ou graduelles entre les éléments d'un ou plusieurs ensembles [LAM00].

Une relation floue R sur les univers de références X_1, X_2, \dots, X_n est définie comme un ensemble flou du produit cartésien $X_1 \times X_2 \times \dots \times X_n$ ayant la fonction d'appartenance μ_R .

Exemple :

La relation floue R : "approximativement égal à" peut être définie sur $\mathbf{R} \times \mathbf{R}$ par la fonction :

$$\mu_R(x, y) = \frac{1}{1+(x-y)^2} \tag{3.8}$$

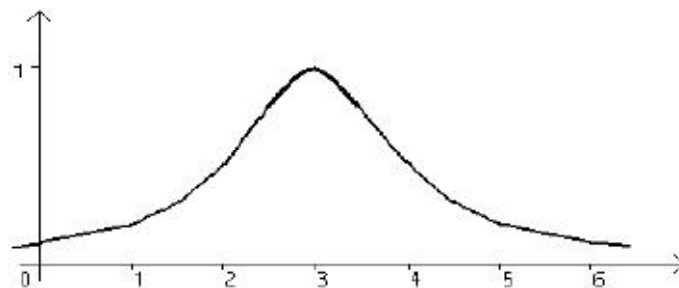


Figure 3.6 : x approximativement égal à 3

La différence entre une relation floue et une relation classique (exacte) est que pour la première, toute valeur d'appartenance dans l'intervalle $[0,1]$ est permise alors que pour la seconde, seules les valeurs 0 et 1 sont permises [BEZ 93].

Une relation floue R sur X est dite :

- symétrique si : $\forall (x, y) \in X \times X, \mu_R = \mu_R (y, x)$.
- antisymétrique si : $\forall (x, y) \in X \times X, (\mu_R (x, y) > 0 \text{ et } \mu_R (y, x) > 0) \Rightarrow x = y$.
- réflexive si : $\forall x \in X, \mu_R (x, x) = 1$.
- transitive si : $R \circ R \subseteq R$ ie : $\forall (x, y) \in X \times X, \mu_{R \circ R} (x, y) \leq \mu_R (x, y)$.

Remarque :

- Une relation de similarité est une relation réflexive, transitive et symétrique. Une relation de similarité permet de modéliser la ressemblance et la proximité.
- Une relation d'ordre floue est une relation réflexive, transitive et antisymétrique. Une relation d'ordre exprime la notion de préférence et d'antériorité.

3.6 Règles floues

Les règles floues sont beaucoup plus faciles à exprimer car elles sont très proches du langage naturel.

3.6.1 Définition

Une règle floue est une proposition floue de la forme : « Si P alors Q » utilisant une implication entre deux propositions floues quelconques P et Q .

Exemple

Règle : « Si la surface est grande »

« Alors prix est élevé »

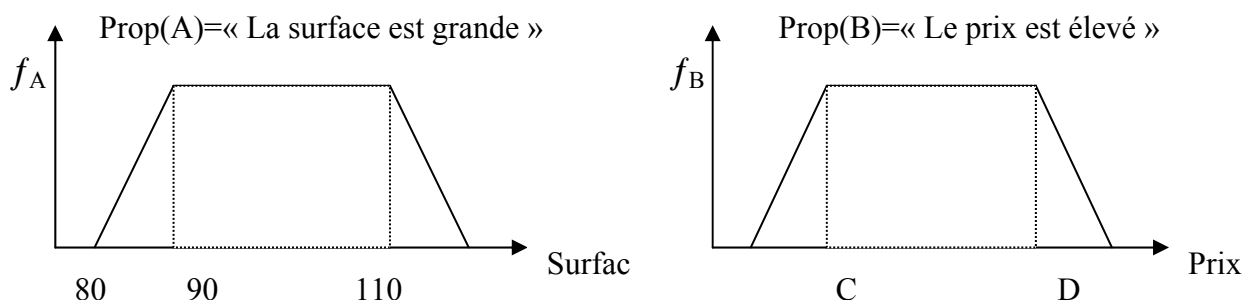


Figure 3.7: Représentation d'une règle floue.

3.6.2 Variables linguistiques

Une variable linguistique est un triplet (V, X, T) dans lequel V est une variable définie sur un ensemble de références X , l'ensemble $T_V = \{A_1, A_2, A_n\}$, fini ou infini contient des sous-ensembles flous normalisés de X , utilisables pour caractériser V .

Fonction d'appartenance

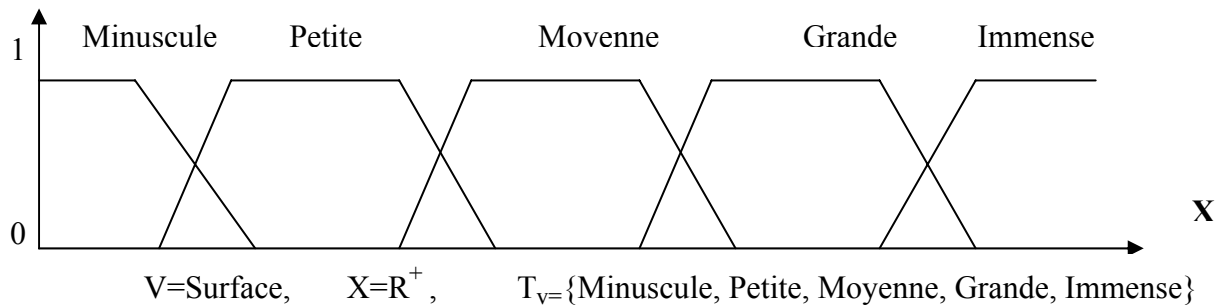


Figure 3.8 : Variable linguistique (V, X, T_V) décrivant la surface d'un appartement

Pour utiliser dans le même cadre, connaissance numérique et connaissance symbolique et si l'attribut A peut subir des variations graduelles, liées à un environnement imprécis, on utilise des expressions qui sont toujours de la forme « V est A » mais pour laquelle la variable est associée à une variable linguistique. On se limite alors à des descriptions de la forme « Le temps est beau », « La robe est chère ».

Etant donnée un ensemble flou L de variables linguistiques (V, X, T_V) de L pour la qualification « V est A » où A est une caractérisation floue de T_V .

3.6.4 Proposition générale floue

Une proposition générale floue est obtenue par l'utilisation conjointe de propositions floues élémentaires « V est A », « W est B » pour des variables V, W, \dots supposée non interactives. La plus simple s'exprime comme la conjointe de propositions floues élémentaires « V est (A et W est B) » où V et W sont définies sur des ensembles de références X et Y (*exemple* : le nombre de page est petit et le prix est élevé) et elle est associée au produit cartésien $A \times B$ caractérisant (V, W) sur l'ensemble $X \times Y$. Sa valeur de vérité est définie par $\min(f_A(x), f_B(y))$. Généralement, on peut construire des propositions floues par conjonction, disjonction ou implication portant sur des propositions floues quelconques.

4. Système flou

Généralement, un système à base de règles floues se compose de trois modules successifs [BER 95] :

- Un module de fuzzification et calcul des propositions élémentaires.
- Un module d'inférence floue (base de règles et moteur d'inférence).

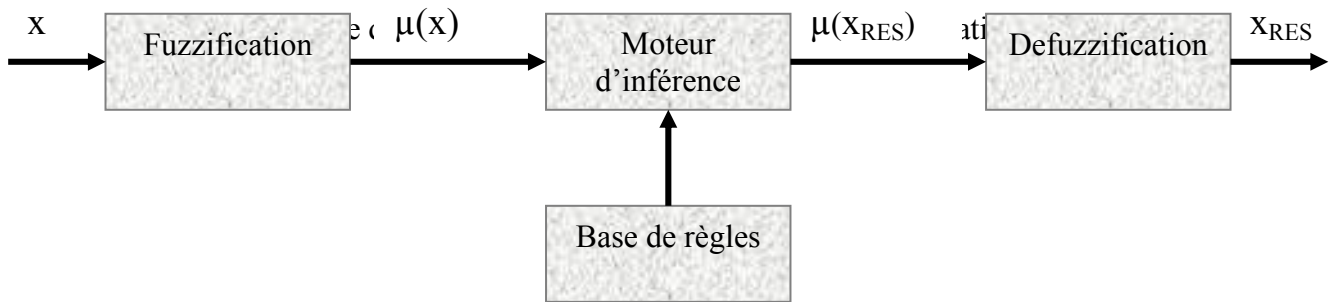


Figure 3.9 : Schéma d'un système floue.

Où x représente le vecteur des entrées, x_{RES} celui des commandes, $\mu(x)$ et $\mu(x_{RES})$ les fonctions d'appartenance correspondantes.

➤ **Le premier module** traite les entrées du système, on définit tout d'abord un univers de discours, un partitionnement de cet univers en classes pour chaque entrée, et des fonctions d'appartenance pour chacune de ces entrées (exemple : pression grande, petite, faible). La première étape, est appelée fuzzification, consiste à attribuer à la valeur réelle de chaque entrée, au temps t , sa fonction d'appartenance à chacune des classes préalablement définies, donc à transformer l'entrée réelle en un sous-ensemble flou.

➤ **Le deuxième module** consiste en l'application de règles de type « si l'écart de température est grand, diminué le débit du fuel ». Ces règles permettent de passer d'un degré d'appartenance d'une grandeur réglant aux degrés d'appartenance d'une commande. Ce module est constitué d'une base de règles et d'un moteur d'inférence qui permet le calcul.

➤ **Le troisième et le dernier module** décrit l'étape de defuzzification qui est la transformation inverse de la première. Il permet de passer d'un degré d'appartenance d'une commande à la détermination de la valeur à donner à cette commande.

4.1. Fuzzification ou quantification floue

Cette première étape consiste à déterminer le degré d'appartenance de chaque variable d'entre a chaque état. Celui-ci est déterminé à l'aide des fonctions d'appartenance définies dans le système.

Pour fuzzifier, il faut donner:

- L'univers du discours i.e.: Plage de variations possibles de l'entrée considérée.
- Une partition en classe floue de cet univers.
- Les fonctions d'appartenances de chacune de ces classes.

Remarque :

- Il faut fuzzifier les entrées et les sorties du processus flou.
- La fuzzification des variables est une phase délicate du processus mis en œuvre par la logique floue. Elle est souvent réalisée de manière itérative et requiert de l'expérience [ELK11]

4.2. Inférence

Opération logique par laquelle on admet une proposition en vertu de sa liaison avec d'autres propositions tenues pour vraies, par exemple :

Si (X est A) Alors (Y est B) 3.9

- La variable floue X appartient à la classe floue A avec un degré de validité $\mu(x_0)$.
- La variable floue Y appartient à la classe floue B à un degré qui dépend du degré de validité $\mu(x_0)$ de la prémisse.

Pour l'inférence :

- Détermination du degré d'appartenance de chacune des conditions des règles.
- Activation de la règle, détermination de la conséquence (min).
- Agrégation des règles (max).
- La méthode choisie à peu d'influence sur le résultat (trois méthodes d'inférence : méthode de Mamdani, méthode de Larsen et méthode de Takagi-Segeno)

4.3. La défuzzification

L'objectif de la défuzzification est de transformer un ensemble flou en une valeur exacte. Soit C un ensemble flou, et defuzz l'opérateur de défuzzification :

$Z_U = \text{defuzz}(C)$, est une valeur précise.

Les opérateurs de défuzzification sont nombreux, citons par exemple :

4.3.1. La méthode du maximum

Elle consiste à choisir comme solution défuzzifiée l'abscisse du maximum de la Fonction d'appartenance des solutions.

Si plusieurs points conviennent, on peut par exemple utiliser une variante, la méthode de la moyenne des maxima, qui consiste à prendre comme solution la moyenne des abscisses des maxima.

$$Z_U = \frac{\int_G w \cdot dw}{\int_G dw} \quad , \quad \text{où } G = \{g \in Z / \mu_c(g) = \max_{z \in Z} (\mu_c(z))\} \quad 3.10$$

Ou

$$Z_U = \frac{1}{N} \sum_{i=1}^n z_i \quad , \quad \text{où } z_i \in Z \text{ et } \mu_i(z_i) = \max_{z_j \in Z} (\mu_c(z_j)) \quad 3.11$$

L'avantage de cette méthode est qu'elle ne nécessite pas une grande puissance de calcul.

4.3.2. Le centre de gravité

Cette méthode est la plus souvent utilisée et donne généralement les meilleurs résultats. Elle consiste à prendre comme solution l'abscisse du centre de gravité des solutions.

$$Z_U = \frac{\int_Z w \cdot \mu_c(w) \cdot dw}{\int_Z \mu_c(w) \cdot dw} \quad \text{Ou} \quad Z_U = \frac{\int_Z w_i \mu_c(w_i)}{\int_Z \mu_c(w_i)} \quad 3.12$$

Par comparaison avec la méthode du maximum, elle exige une plus grande puissance de calcul [LAC 98].

5. Modèles flous

La description des systèmes suffisamment complexes est faite au moyen d'un traitement approprié de l'information, basé sur des règles « If-Then », de certaine admission d'incertitude et/ou d'imperfection et d'imitation des mécanismes d'apprentissage (appliquer avec succès par l'être humain). [GRI 02].

5.1 Définition d'un modèle flou

Un modèle flou est un ensemble \mathfrak{R} , de règles floues reproduisant approximativement la relation existant entre les données en entrée et les données en sortie d'un système.

Un modèle flou \mathfrak{R} est dit utile, si les données en entrée et les données en sortie sont représentatives et si $\mathfrak{R}(x_K) \approx y_k$, tel que $K=1..N$. [RUN 99].

La modélisation floue de systèmes se divise typiquement en deux catégories qui diffèrent dans leurs capacités de représenter différents types d'informations : le modèle de Mamdani-Assilian(MA) ou le modèle linguistique, basé sur l'expérience et le modèle de Takagi-Sugeno(TS), plus approprié pour une approche basée sur les données.

Pour chaque modèle on présente la structure des règles, la méthode d'inférence et de defuzzification.

5.2 Modèle de mamdani-assilian (MA)

Le modèle MA a été introduit comme un moyen pour capturer les connaissances qualitatives (ou semi-qualitatives) disponibles, en forme de règles « If-Then » [GRI 02].

Etant données :

L'ensembles en entrée $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^p$, l'ensemble en sortie $Y = \{y_1, y_2, \dots, y_k\} \in \mathbb{R}^q$, l'ensemble $Z = \{z_1, z_2, \dots, z_n\} \subset \mathbb{R}^{p+q}$ tel que z_i est une concaténation entre un vecteur d'entrée et un vecteur de sortie, et le nombre de règles c . Le modèle MA peut est défini comme suit [RUN 99] :

Chaque règle R_i peut être vue comme une relation floue, elle est de la forme :

$$R_i : \text{If } \bigwedge_{\ell=1..p} \mu_{i\ell}(x^{(\ell)}) \text{ Then } v_{im}(y^{(m)}) = \mu_{i(p+m)}(y^{(m)}); \forall m=1, \dots, q.$$

3.13

- $x = (x^{(1)}, x^{(2)}, \dots, x^{(p)})$: est le vecteur en entrée du système.
- $Y = (y^{(1)}, y^{(2)}, \dots, y^{(q)})$: est le vecteur en sortie du système.
- $\mu_{i\ell}, v_{im} : \mathbb{R} \rightarrow [0,1]$; tel que : $i=1, \dots, c$; $\ell=1, \dots, p+q$; $m=1, \dots, q$.
- $\mu_{i\ell}(x^{(\ell)})$: est appelée la fonction d'appartenance précédente; $i=1, \dots, c$; $\ell=1, \dots, p$.
- $\mu_{i(p+m)}(y^{(m)})$: est appelée la fonction d'appartenance conséquente; $i=1, \dots, c$; $m=1, \dots, q$.

- $\wedge : [0,1]^2 \rightarrow [0,1]$ est une T-norme (Conjonction floue).

Chaque $v_{i m} (y^{(m)})$, $i=1, \dots, c$; $m=1, \dots, q$; représente la sortie floue de R_i pour la composante de sortie y^ℓ , elle est calculée comme :

$$v_{i m} (y^{(m)}) = \mu_{i(p+m)}(y^{(m)}) \wedge \bigwedge_{\ell=1..p} \mu_{i \ell} (x^{(\ell)}) , m=1, \dots, q. \quad 3.14$$

Les composantes en sortie de tout le système sont calculées comme :

$$y^{(m)} = d (\bigvee_{i=1, \dots, c} v_{i m} (y^{(m)})) , m=1, \dots, q. \quad 3.15$$

- $d : (\{\mu : \mathbb{R} \rightarrow [0,1]\}) \rightarrow \mathbb{R}$ est une defuzzification.
- $\vee : [0,1]^2 \rightarrow [0,1]$ est une T-conorme (Disjonction floue).

Le modèle flou MA est un modèle qui a une large acceptation, intuitif et exprime bien les données en entrée fournies par l'être humain (généralement l'expert). [MAT 01].

5.3 Le modèle de Takagi-sugeno :

Le modèle TS est basé sur la méthode de raisonnement TS qui a été proposée par Takagi et sugeno en 1985 [TAK 85]. Ce type de modèle est formé par des règles logiques qui ont un antécédent flou et un conséquent qui est une fonction concrète des variables qui interviennent dans l'antécédent. Il est défini comme suit [RUN 99]:

Chaque règle est de la forme :

$$R_i: \text{If } \bigwedge_{\ell=1..p} \mu_{i \ell} (x^{(\ell)}) \text{ Then } y = f_i (x) , \text{ tel que } , f_i : \mathbb{R} \rightarrow \mathbb{R} , i=1, \dots, c. \quad 3.16$$

Les composantes en sortie de tout le système sont calculées comme:

$$y^{(m)} = \frac{\sum_{i=1}^c \bigwedge_{\ell=1}^p (\mu_{i \ell} (x^{(\ell)})) f_i^*(x)}{\sum_{i=1}^c \bigwedge_{\ell=1}^p \mu_{i \ell} (x^{(\ell)})} \quad 3.17$$

Modèle peut être appliqué pour les systèmes statiques et dynamiques multi entrée et multi sortie, permet une application relativement facile des techniques d'identification à partir des données, convenables pour les analyses mathématiques et fonctionne mieux avec les techniques adaptatives et celle d'optimisation. [MAT 01].

5.4 Méthodes de modélisation

La modélisation floue est un apprentissage supervisé car pour chaque vecteur x_k en entrée le vecteur de sortie y_k est connu.

Deux méthodes de modélisation floue sont distinguées, modélisation avec évaluation de règles et modélisation sans évaluation de règles.

Nous présentons la méthode de modélisation avec évaluation de règles [RUN 99].

5.4.1 Modélisation avec évaluation de règles

Pour générer un modèle flou \mathfrak{R} à partir d'un ensemble de données d'entrée et de sortie on commence par définir une règle de base initiale R_0 , ensuite l'évaluee en utilisant l'ensemble de données X en entrée et enfin comparer la sortie du système avec l'ensemble de données en sortie désiré Y . En fonction de la différence observée entre le résultat du système et le résultat attendu, \mathfrak{R} est mis à jour jusqu'à l'obtention d'un modèle avec une exactitude suffisante. L'approche d'évaluation de règle utilise généralement les algorithmes génétiques ou les réseaux de neurones.

a. Les réseaux de neurones

Pour utiliser un réseau de neurone, toutes les composantes du système floues : fonction d'appartenance, règles et defuzzification sont organisées en topologie d'un réseau de neurone. Durant l'apprentissage le réseau de neurone représentant la structure du système flou est évalué en utilisant l'ensemble de données en entrée X et les résultats en sortie sont comparés avec l'ensemble de données en sortie Y désiré. Les paramètres du réseau de neurone sont ajustés en utilisant un algorithme d'apprentissage. Après l'apprentissage le réseau de neurone est restauré en un système flou. [RUN 99]

b. les algorithmes génétiques

Pour appliquer les algorithmes génétiques, un ensemble de chaîne de chromosomes est utilisé pour représenter les différents candidats pour le modèle flou. Chaque chaîne de

chromosomes contient des informations décrivant un modèle entier c.à.d. les paramètres de la fonction d'appartenance et des informations concernant la structure de la base de règles.

Dans chaque génération tous les modèles candidats de la population courante sont évalués en utilisant l'ensemble de données en entrée X et les valeurs de sortie de ces candidats sont comparées avec l'ensemble de données Y en sortie désirée.

En fonction d'une erreur d'approximation calculée pour chaque modèle, les meilleurs candidats sont sélectionnés pour la reproduction de la prochaine génération.

6. Applications de la logique floue pour la segmentation d'images médicales

Les outils disponibles pour mener à bien la tâche de segmentation sont si nombreux que la plupart des méthodes combinent plusieurs approches, et qu'il y a de multiples façons de les classer. Nous donnons dans cette partie. Quelques exemples de techniques où le rôle de la logique floue nous a semblé intéressant. Ils sont organisés en quatre groupes. Le premier concerne les approches qui cherchent à représenter directement les objets de l'image par des sous-ensembles flous, et utilisent principalement la notion de fonction d'appartenance aux sous-ensembles flous pour faire de la classification, mais exploitent peu les opérateurs et le raisonnement approximatif. Les méthodes et applications du deuxième groupe sont plus sophistiquées. Utilisent les applications de la logique floue pour la segmentation d'images médicales. Ces mêmes fonctions d'appartenance, mais les combinent ou effectuent des opérations dessus et éventuellement raisonnent à partir d'elles pour prendre une décision. Le troisième groupe utilise également le raisonnement approximatif, mais le replace dans un cadre spatial, en exprimant toutes les connaissances utilisées dans le référentiel des objets de l'image à traiter. Enfin, les applications du dernier groupe concernent plus particulièrement la commande floue.[CYB 05].

6.1 Représentation des objets par des sous-ensembles flous

Les applications portent en général sur la classification de tissus, ou la classification en deux catégories : sain et pathologique. Dans ce contexte, les opérateurs de la théorie des sous-ensembles flous et le raisonnement approximatif sont peu exploités, puisque la décision revient juste à attribuer une classe par pixel ou voxel. Ce qui peut être fait directement en observant les degrés d'appartenance. Un des premiers algorithmes visant à représenter des connaissances imparfaites par des sous-ensembles flous et appliqué dans le domaine médical est l'algorithme des C-moyennes floues (ou FCM, pour Fuzzy C-means), Cet algorithme prend directement en compte le fait que les images elles-mêmes peuvent être considérées comme des scènes floues, où les objets apparaissent avec des frontières imprécises, en grande partie a

cause des limitations de la résolution des systèmes d'acquisition. Le FCM réalise une classification des données telles que les classes sont représentées par des sous-ensembles flous. Il construit donc leur fonction d'appartenance en calculant le degré d'appartenance de chaque voxel aux différentes classes possibles. Les degrés d'appartenance peuvent être vus comme le degré de similarité entre le niveau de gris du voxel courant et le prototype associé à la classe, également appelé centroïde. L'avantage principal de cette méthode est son caractère non supervisé, qui ne nécessite pas de base d'apprentissage. Cependant, les fonctions d'appartenance calculées peuvent avoir une forme difficile à adapter pour certaines applications car elles ne décroissent pas en fonction de la distance à l'objet représenté. De plus, le résultat est fortement dépendant de l'initialisation. Des extensions ont donc été proposées, comme FCM approximé qui remplace certains calculs par une recherche dans des tables. Cet algorithme, ainsi que le FCM traditionnel, ont été appliqués à la segmentation d'IRM cérébrales 2D multi modalités (pondérations T1, T2 et densité de proton) et comparés aux performances d'un réseau de neurones qui propose une classification supervisée [HAL 92]. L'étude vise à la fois à déterminer les classes, matière blanche (MB), matière grise (MG) et liquide céphalo-rachidien (LCR) et à classer les tissus sains et pathologiques. Pour cela, elle a été menée sur 12 sujets dont 8 sains et 4 présentant une tumeur. Les critères de comparaison sont la robustesse à l'initialisation de la logique floue, la qualité visuelle des résultats de segmentation et le temps de calcul. Le réseau de neurones est moins sensible à l'initialisation et plus rapide que les méthodes FCM et FCM approximés, cependant, il nécessite un apprentissage, donc plus d'interaction avec un opérateur. Les résultats des différentes méthodes sont sensiblement équivalents, sauf dans quelques cas pathologiques. L'article ne conclut pas sur la supériorité d'une méthode mais suggère que les méthodes FCM et FCM approximés sont prometteuses.

6.1.1 Applications et améliorations du FCM

Elles ont été largement utilisées et améliorées par la suite. Par exemple, elles peuvent servir à distinguer des coupes IRM normales de celles présentant une tumeur et à faire une estimation préliminaire de la localisation du cerveau dans un système de segmentation de tumeurs [CLA 98]. Elles sont également employées pour définir les classes, tissu sain et endommagé dans le contexte de la segmentation de polypes, du colon sur des coupes scannées [YAO 04]. Les résultats de la classification sont ensuite utilisés pour faire évoluer un modèle déformable vers le contour des polypes. Parmi les améliorations du FCM, une version appelée AFCM pour FCM adapté propose de prendre en compte les artefacts dus aux inhomogénéités de champ dans la méthode de segmentation [PHA 99]. L'AFCM modifie la fonction de cout

minimisée dans l'algorithme FCM pour y introduire un gain modélisant la variation du champ dans le volume IRM, et est implémenté avec une stratégie multi-échelle pour réduire le temps de calcul. Il est testé sur des données 3D réelles multimodales et sur la base de données Brainweb [COL 98], en pondération T1, pour la classification en 3 classes : MB, MG et LCR. L'algorithme est en effet robuste aux inhomogénéités de champ, contrairement au FCM et aux autres méthodes testées, et donne des résultats prometteurs. En revanche, il n'inclut pas d'information spatiale, et tend, comme le FCM, à rechercher des classes de même forme et de même taille, car il n'utilise pas la matrice de covariance des tissus. Les résultats intéressants de cette méthode ont conduit à la réutiliser dans un contexte plus applicatif, comme première étape de la segmentation de la surface médiane du cortex [PHA 99]. L'approche proposée se décompose en trois étapes dans lesquelles la théorie des sous-ensembles flous intervient. Il s'agit tout d'abord de réaliser la traditionnelle classification MB, MG et LCR du cerveau par l'algorithme AFCM. Puis l'iso-surface correspondant au degré d'appartenance 0.5 de la matière blanche est extraite et typologiquement corrigée. En effet, cette surface est déformée pour évoluer vers la surface médiane du cortex, au moyen de forces définies grâce aux fonctions d'appartenance des sous-ensembles flous obtenus par la classification AFCM. Dans cette dernière étape de la méthode, l'intérêt de l'interprétation linguistique des sous-ensembles flous permet de définir une force de pression pour pousser la surface vers l'extérieur jusqu'à ce qu'elle entre dans la matière grise. L'estimation de la frontière de la Matière grise se fait alors aisément à l'aide des fonctions d'appartenance. L'ensemble de ces étapes, valide sur 6 sujets, fournit finalement une approche modulaire permettant d'obtenir une surface médiane du cortex topologiquement correcte. Cette méthode présente cependant une certaine sensibilité au bruit, nécessite quelques interactions manuelles et peut produire de mauvais résultats dans les zones. Ces limitations ont été traitées dans [PHA 02] et la méthode améliorée finalement incorporée dans un outil global de reconstruction corticale [HAN 04].

6.2 Modélisation des interactions entre différentes structures par sous-ensembles flous

Ce paragraphe donne quelques exemples d'utilisation de la théorie des sous-ensembles flous qui poussent un peu plus loin la tâche de la classification des tissus. En particulier, les classes ne recouvrent plus nécessairement des structures anatomiques réelles, mais peuvent représenter des mélanges de tissus ou des objets composés de plusieurs structures. Des notions plus avancées de la théorie des sous-ensembles flous, comme celle de la relation floue, sont également utilisées.

Une autre approche consiste à utiliser conjointement des méthodes probabilistes et la théorie des sous-ensembles flous, afin de tirer parti de leur complémentarité [RUA 02]. Il s'agit dans ces travaux d'effectuer une classification des tissus du cerveau par le biais d'un champ de Markov en tenant compte de l'effet de volume partiel. La matière blanche (MB), la matière grise (MG) et le liquide céphalo-rachidien (LCR) sont considérés comme des classes pures, par opposition aux mélanges MB/MG et MG/LCR en particulier. Contrairement à la plupart des méthodes de classification markoviennes, l'approche proposée ne modélise pas les intensités de l'image observée comme la somme pondérée des intensités des classes pures et d'un bruit, mais comme la somme pondérée de variables aléatoires représentant les classes pures, ce qui permet de prendre en compte la variabilité de ces classes. D'autre part, la connaissance anatomique du cerveau est implicitement intégrée au processus, par le choix du nombre de classes. La méthode est testée sur des données réelles sur lesquelles elle donne de meilleurs résultats que le FCM. Cette méthode est également appliquée en première étape d'un algorithme modulaire de segmentation conjointe de structures internes du cerveau [XUE 01]. Les fonctions d'appartenance aux classes de tissus obtenues sont alors utilisées à plusieurs reprises dans l'algorithme, en combinaison avec la notion de région d'intérêt floue (ROI floue). Les ROIs sont utilisés conjointement avec les classes obtenues par champ de Markov ou dans un algorithme génétique, puis les résultats sont affinés avec un algorithme de croissance de régions. Cette méthode utilise avantageusement la théorie des sous-ensembles flous pour représenter de manière implicite les formes des structures d'intérêt. De plus, la notion de distance et d'interaction entre ces structures est introduite, toujours par le biais des fonctions d'appartenance. En revanche, les très nombreuses étapes de l'algorithme multiplient les risques d'erreurs et d'imprécisions, et peuvent conduire à un temps d'exécution long.

6.2.1 Connectivité floue et objets flous

Une autre série de travaux est partie de l'idée de représenter des objets flous en fonction de deux concepts : la composition hiérarchique des objets (un objet peut en contenir plusieurs) et la notion d'ensemble d'objets [UDU 96]. Cette notion se conçoit bien lorsqu'on considère par exemple un volume IRM cérébral. Ces définitions sont donc directement fondées sur l'interaction entre les différentes structures de l'image. Toutes ces notions constituent un cadre théorique dans lequel plusieurs types de décisions peuvent être prises, ce qui permet aux auteurs de proposer des algorithmes de segmentation et classification, testés sur un scanner du genou, un volume angiographique et des IRM multimodales. L'application à la segmentation de lésions de sclérose en plaques sur ces volumes IRM fait l'objet d'un autre article [UDU 97], et la méthode a aussi été utilisée pour la séparation des veines et des artères en angiographie

[LEI 01]. La théorie proposée dans ces travaux est solide, mais les algorithmes présentent quelques inconvénients majeurs. En particulier, l'initialisation doit être faite manuellement et les résultats de segmentation des lésions doivent être également valides manuellement par un opérateur. De plus, le calcul de certains paramètres nécessaires à la définition de l'affinité floue, tels que la moyenne et la variance des classes des objets, n'est pas automatisé. En fin, la définition d'un objet est obligatoirement liée à un seuillage au sens de la connectivité, qui doit être déterminé ad hoc. Cette dernière limitation est corrigée dans une version ultérieure de la méthode [UDU 02], où la notion de connectivité floue est remplacée par celle de connectivité floue relative. Ce qui signifie que la méthode prend en compte les différents objets de l'image pour définir l'affinité floue, et les met en concurrence dans les algorithmes de segmentation, qui sont de ce fait plus robustes et fournissent de meilleurs résultats.

6.3 Relations spatiales définies dans le référentiel des images

Les exemples précédents montrent que la théorie des sous-ensembles flous permet d'utiliser des informations qui conduisent à segmenter des structures qui auraient été difficiles à identifier à partir d'une image seule, même à l'aide d'un atlas ou d'un modèle de forme. Les travaux [Udu 02], [XUE 01] s'intéressent particulièrement aux interactions entre les différentes structures anatomiques. Ces interactions peuvent être modélisées sous la forme de relations spatiales. Elles s'avèrent alors utiles pour segmenter des structures sur lesquelles on a une connaissance anatomique, tout en tenant compte de leur variabilité, ce que ne fait pas nécessairement une méthode basée sur le recalage, qui met en œuvre une approche plus globale [BLOC 03]. Une approche plus poussée consiste à segmenter des structures de manière hiérarchisée, en partant de celles qui sont les plus faciles à identifier, puis en traitant progressivement les autres, grâce aux relations spatiales [BLOC 03]. Le résultat est donc constitué d'un ensemble de structures, et non d'une seule cible, et l'approche constitue un cadre méthodologique générique, qui peut s'appliquer à d'autres organes. En prolongation de ces travaux, les relations spatiales ont été introduites non seulement dans un processus de localisation hiérarchique des structures, mais également comme contrainte d'évolution d'un modèle déformable, de façon à affiner localement les résultats de la segmentation [COL 04].

7. Caractéristiques, avantages et limitations de la logique floue

7.1 Caractéristiques

- La connaissance est structurée en règles linguistiques.
- Les calculs sont numériques.

7.2 Avantages

- Facilité de construction et d'interprétation des règles : les règles sont formulées de manière naturelle par les experts, comme en symbolique.
- Interprétation numérique entre les règles : lorsque deux règles sont activées en même temps car leurs prémisses sont en partie vérifiées, alors la conclusion proposée du système peut prendre une valeur intermédiaire entre les conclusions proposées par celle-ci.
- Intégration de la connaissance à priori.
- Robustesse vis à vis des incertitudes.

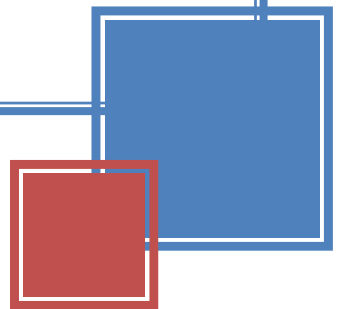
7.3 Limitations

- Construction manuelle de règle suivant l'intuition de l'opérateur ou les connaissances de l'expert.
- Optimisation manuelle des fonctions d'appartenance : nombre de termes linguistiques, choix de la forme des fonctions d'appartenance (triangle, trapèze,...), position du centre, largeur.

8. Conclusion

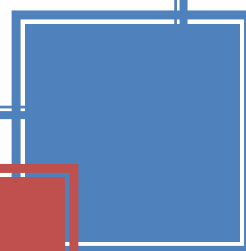
La théorie des sous-ensembles flous a été largement utilisée en traitement d'images médicales en raison de sa capacité à représenter des connaissances imprécises. En revanche, La logique floue ouvre des possibilités remarquables de codification des connaissances des experts. Cependant, les applications utilisant la logique floue ne sont pas fondamentalement plus performantes. Elles sont tout simplement plus faciles à réaliser et à utiliser : l'utilisation faite par la logique floue d'expressions du langage courant permet au système flou de rester compréhensible pour les personnes non expertes. C'est ainsi que des machines complexe peuvent devenir plus convivial grâce à l'utilisation de la logique floue. Malheureusement la manipulation de règles non précises peut générer un nombre d'erreurs non négligeable. La mise en place d'un système floue nécessite donc une attention particulière lors de la phase de test de manière à détecter les éventuelles aberrations du système.

CHAPITRE 4



PARTIE 1 :

CONCEPTION DÉTAILLÉE DE L'APPROCHE PROPOSÉE



1. Problématique et solutions

Le clustering, en général, est le regroupement des objets dans des *clusters* (groupes) selon des critères de similarité. Il inclut plusieurs méthodes : par partitionnement, hiérarchique, basée sur les modèles etc ... Nombreux de ces algorithmes ont été adaptés pour la segmentation d'images. Nous citons comme exemple : FCM (Fuzzy C-Means), PCM (Probabilistic C-Means), K-nearest etc...

Au niveau de la segmentation, les méthodes de clustering sont utilisées pour diviser une image en régions homogènes en maximisant la similarité intra classes et en minimisant la similarité interclasses. Il est clair que ce processus est suffisamment complexe à décrire notamment la relation existant entre les pixels et les centres des clusters. Donc nous jugeons que la méthode de clustering flou basée sur les modèles est le bon moyen pour définir cette relation.

Toutefois les modèles de clustering flous basés sur la *fonction objective*, dont l'optimisation guidée par deux conditions de l'extremum local, (une pour mettre à jour la matrice de partition, et une autre pour mettre à jour les prototypes), sont des modèles où la *fonction d'appartenance* obtenue par l'extension de la condition de leur *fonction objective* est restreinte à des formes particulières; déterminées par l'équation de mise à jour des partitions dérivée à partir de leur fonction objective. Or l'utilisateur peut être intéressé par le fait de choisir la forme de la fonction d'appartenance la plus appropriée à son application. Pour remédier à cette restriction, nous proposons d'*adapter* pour la segmentation d'images médicales, un nouveau *modèle* ou *paradigme* de *clustering flou*, ne se basant pas sur l'optimisation d'une fonction objective, appelé ACE (Alternating Cluster Estimation). Nous proposons également de concevoir un schéma d'estimation des équations de mise à jour afin d'extraire un modèle flou global définissant la relation entre les entrées et les sorties de notre système.

Le modèle que l'on propose d'adapter libère l'utilisateur de la contrainte du choix des équations de mise à jour devant correspondre à un modèle de *fonction objective*, ceci en lui offrant la possibilité de choisir celles qu'il estime *appropriées* à son application. Le *schéma d'estimation* proposé pour l'extraction des modèles flous est un système d'inférence *flou* de type *Mamdani-Assilian*, implémenté sur un *réseau neuronal* entraîné par un *algorithme génétique*. Ce choix de combiner les *réseaux neuronaux* avec les *algorithmes génétiques* se justifie par deux raisons ; la première est que l'utilisation des *réseaux neuronaux* présente deux inconvénients majeurs qui peuvent être contournés par les *algorithmes génétiques* et la deuxième est que

chacun des *réseaux de neurones* et des *algorithmes génétiques* a des caractéristiques inhérentes qui se révèlent utiles dans notre travail. Nous nous étalons sur les deux raisons qui nous ont incités à unir les deux outils d'optimisation et les apports des deux outils et les limitations des *réseaux neuronaux*. Les *principaux* avantages, car il y en a d'autres, des *réseaux neuronaux* résident dans leur capacité d'effectuer des traitements *parallèles* et dans leur *puissance* de calcul, deux choses qui s'avèrent un besoin essentiel pour effectuer un traitement sur une image comme par exemple la segmentation. Les *algorithmes génétiques* de leur côté offrent la possibilité d'explorer *en parallèle* un espace considérable de *solutions potentielles* pour donner à la fin la meilleure des solutions. C'est à dire la *solution optimale*. L'inconvénient principal des *réseaux neuronaux* réside dans le fait qu'ils soient incapables de donner, *toujours*, une solution *optimale* car la plupart du temps ils convergent vers un minimum local qui n'est pas forcément une solution optimale. Afin de bien cerner cette limitation nous utilisons *l'algorithme génétique* comme *algorithme d'apprentissage* du réseau puisque ce dernier finit toujours par choisir le meilleur des candidats c'est à dire la meilleure solution. En plus de cet inconvénient, l'utilisation des *réseaux neuronaux* impose le choix des fonctions de calcul qui doivent être dérivables, ce qui n'est pas toujours le cas. Encore une fois *l'algorithme génétique* résout ce problème car il ne nécessite pas les calculs de dérivées.

Le schéma suivant fournit une description globale de notre système :

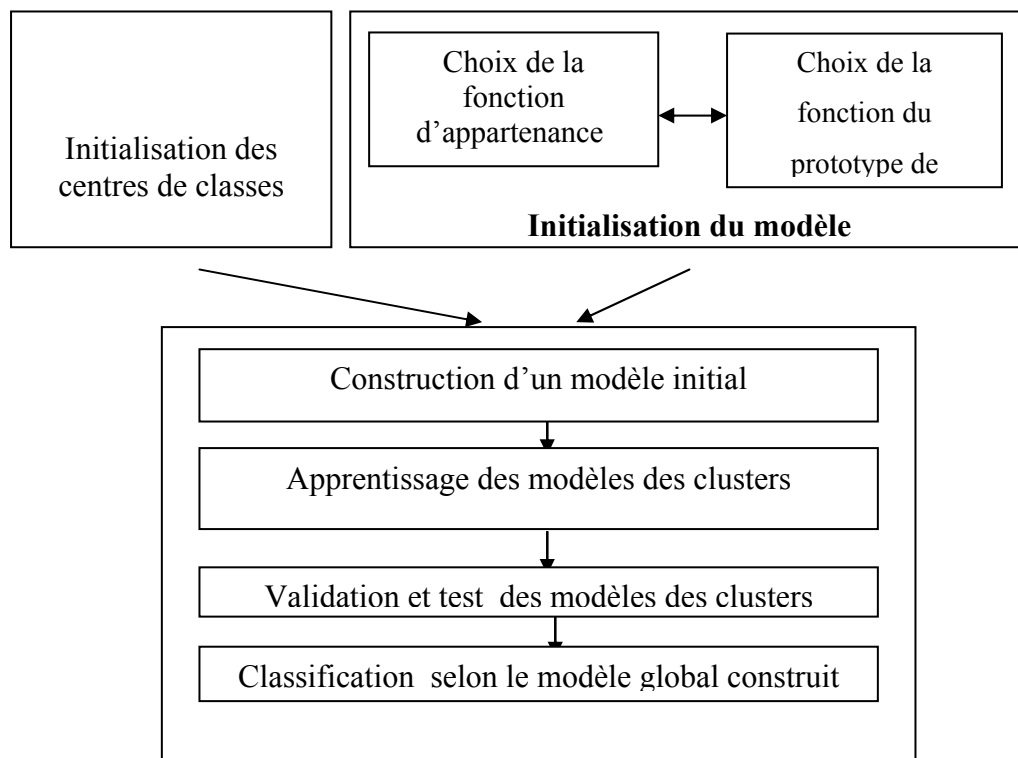


Figure 4.1 Description du système.

Dans ce qui suit, nous présentons une description du modèle ACE. En suite, nous exposons la conception de notre système, puis sa réalisation. Finalement nous concluons avec une analyse des résultats de la segmentation effectuée sur différentes images médicales.

2. Description de l'ACE

Les modèles notés $M(U, V)$ basés sur l'optimisation d'une *fonction objective* sont abandonnés en faveur d'un modèle plus général, défini par l'architecture de l'algorithme AO et les *équations* de mise à jour de la matrice de partition U et du vecteur des prototypes V , choisies par l'utilisateur. Les instances de notre modèle peuvent ou non optimiser une *fonction objective* particulière. Et dans le cas où l'utilisateur choisit des équations de mise à jour ne correspondant pas à un modèle de *fonction objective*, cas sujet de notre intérêt, les clusters et les centres de clusters sont estimés par une mise à jour alternative des prototypes et des partitions. Par conséquent le modèle est appelé ACE : Alternating Cluster Estimation.

Afin de bien assimiler l'avantage qu'offre le modèle ACE, nous estimons qu'une comparaison avec les modèles basés sur une fonction objective, que nous résumons dans le schéma ci-dessous est utile.

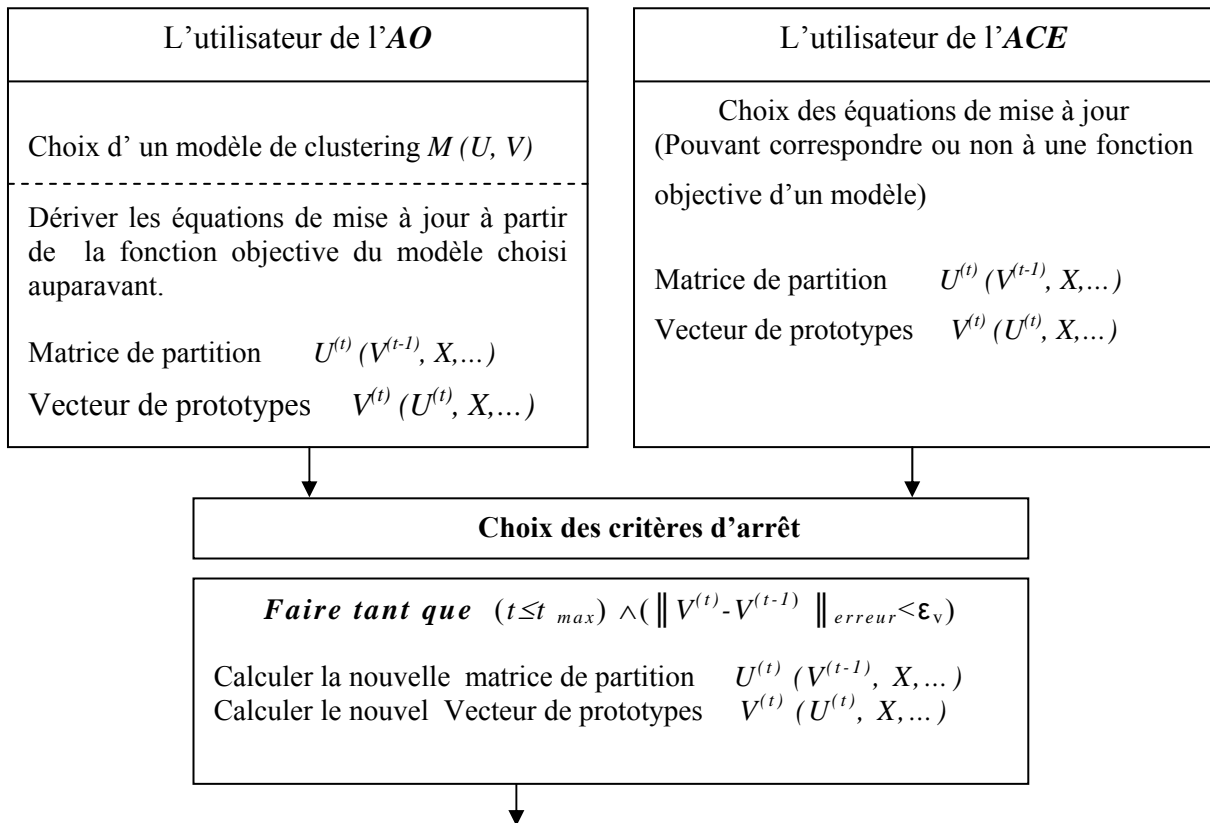


Figure 4.2: AO versus ACE

Le schéma de la figure 4.2 montre que L'ACE a la même architecture itérative que l'algorithme AO. En effet, initialement les prototypes sont initialisés aléatoirement ensuite à chaque itération d'optimisation la matrice de partition U et le vecteur de prototype V sont mis à jour. L'algorithme se termine lorsque les critères d'arrêts sont satisfaits.

En revanche une différence cruciale existe dans le fait que, l'utilisateur de l'AO choisit un modèle $M(U, V)$ qui la forme des équations de mise à jour. Tandis que, l'utilisateur de l'ACE abandonne le modèle $M(U, V)$ et choisit par contre des équations de mise à jour qui n'optimisent pas forcément une fonction objective. Dans le cas où le choix de l'utilisateur est porté sur des équations correspondant à un modèle, le clustering par l'AO devient un cas spécial de l'ACE.

Le modèle ACE permet d'explorer l'ensemble des fonctions d'appartenance existantes et celui des fonctions des prototypes Il choisit n'importe quelle fonction d'appartenance avec n'importe quelle fonction de prototype ou encore mieux, utiliser dans la même implémentation différents types de fonctions d'appartenance et de prototype ; ceci pour discerner la combinaison la plus appropriée à notre problème (segmenter une image).

Dans ce cadre, l'utilisateur de l'ACE peut imaginer qu'il a à sa disposition une barre d'outil de fonctions d'appartenance et une autre de fonctions de prototypes. il n'a qu'à choisir chaque équation à part, sans aucune contrainte.

Avant d'exposer notre méthodologie de conception, nous tenons à expliciter les points suivants :

- Nous choisissons le mode **HSV** pour représenter les caractéristiques des pixels de l'image couleur à segmenter. Ceci pour deux raisons, la première est qu'il présente de nombreux avantages, la deuxième raison est qu'il s'avère efficace selon les expériences mentionnées dans [CHE 01].
- La méthode de clustering que l'on propose de concevoir est une méthode supervisée du moment qu'elle est basée sur la modélisation d'une relation entre les entrées et les sorties du système. Puisque nous utilisons un réseau neuronal pour implémenter cette modélisation alors le mode d'apprentissage de ce réseau neuronal est implicitement supervisé lui aussi.

3. Méthodologie de la conception

La conception de notre système comporte deux étapes : *adaptation* de l'ACE pour la segmentation d'images et *conception* d'un schéma d'estimation des clusters. Nous présentons un schéma qui résume ces deux étapes :

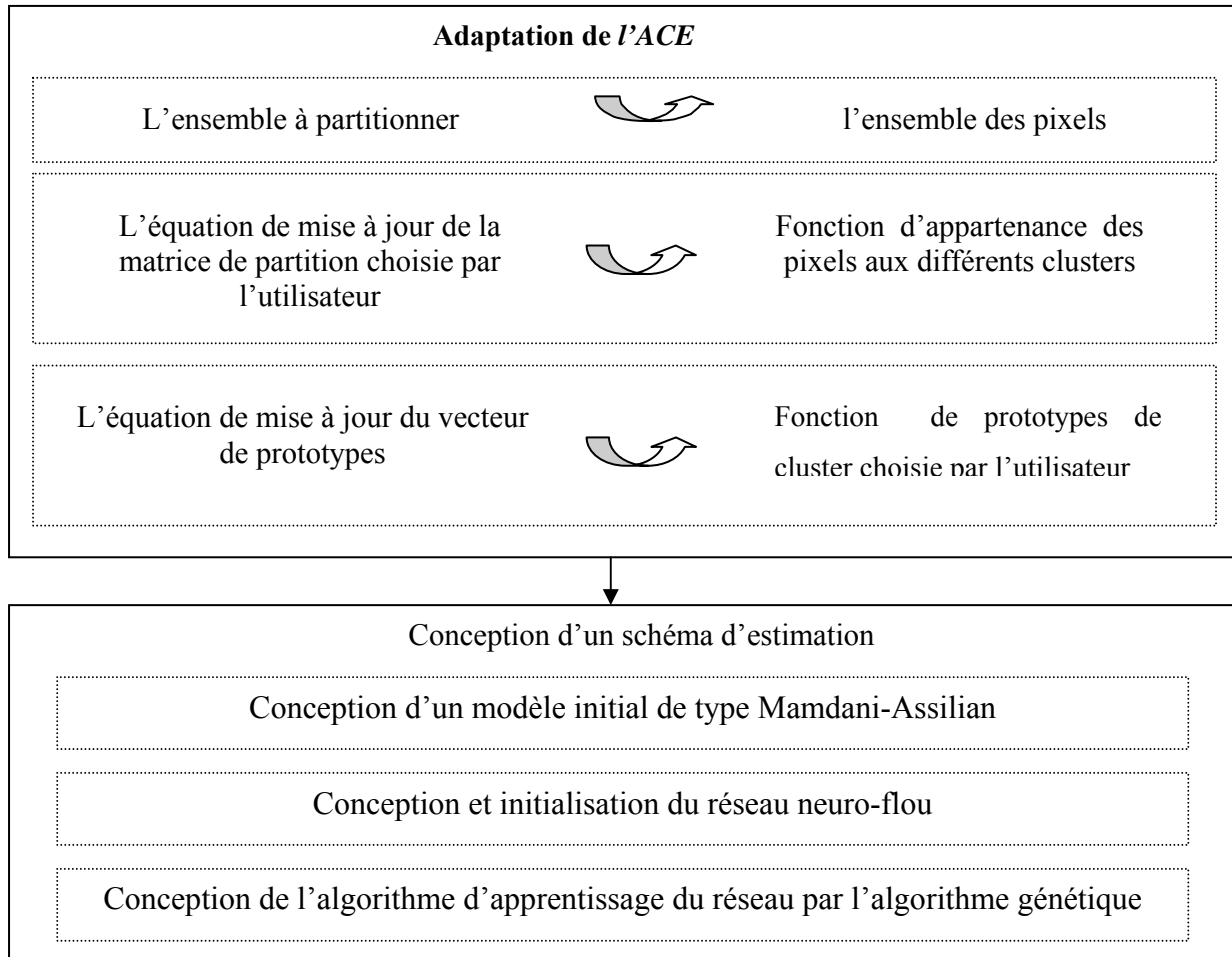


Figure 4.3 Schéma illustratif des étapes de la conception

3.1 Adaptation de l'ACE

Le clustering par l'ACE en général est l'opération de partitionnement d'un ensemble d'objets en sous-ensembles d'objets, tel que, chaque sous-ensemble respecte un critère d'homogénéité. Dans la segmentation d'image, et par projection, comme indiqué dans la figure 4.3 l'ensemble des objets sensés être partitionnés est l'ensemble des pixels. L'équation de mise à jour de la matrice de partition choisie par l'utilisateur est une fonction d'appartenance des pixels aux clusters. L'équation de mise à jour du vecteur des prototypes est la fonction d'estimation du vecteur des prototypes représentant les clusters. Parmi les prototypes existant, nous choisissons le prototype point (centre de cluster).

Tout le long de notre conception nous avons utilisé comme exemple illustratif une instance du modèle ACE :

L'équation de mise à jour de la matrice de partition est une fonction d'appartenance **exponentielle** et l'équation de mise à jour du vecteur du prototype est la fonction de defuzzification du centre de gravité (COD).

3.2 Conception d'un schéma d'estimation

Afin d'estimer les équations de mise à jour de l'ACE, nous construisons un système d'inférence flou de type Mamdani-Assilian (MA) constitué de cinq modules : *Fuzzification* des variables d'entrée, *application des opérateurs flous* (AND ou OR) dans l'antécédent, *l'implication* de l'antécédent vers le conséquent, *l'agrégation* des conséquents des règles et la *defuzzification*. Ensuite nous implémentons les différents modules de ce système flou sur un réseau neuronal et finalement nous utilisons l'algorithme génétique pour entraîner le réseau.

Dans les sections suivantes, illustrons chacune des étapes précédentes.

3.2.1 Construction du modèle initial

A ce niveau nous construisons la base de règles initiale de type Mamdani-Assilian qui décrit initialement la relation existant entre l'espace des entrées (pixels) et l'espace des sorties (les centres de clusters ou prototypes), dans notre système. Notre base de règles initiale est une collection de règles floues « if-then », ayant la forme suivante:

$$\mathbf{R}_i : \text{If } \bigwedge_{\ell=1 \dots p} \mu_{i \ell} (X^{(\ell)}) \text{ Then } v_{i m} (Y^{(m)}) = \mu_{i (p+m)} (Y^{(m)}); \quad \forall m=1, \dots, q, i=1, \dots, c \quad (1)$$

Définir toutes les variables. La partie gauche de la règle est appelée l'antécédent et la partie droite est appelée le conséquent.

Dans la base des règles initiale, chaque cluster est décrit par une règle (modèle local), par conséquent le nombre de règles est déterminé selon le nombre de clusters spécifiés par l'utilisateur. Dans (1), l'antécédent et le conséquent sont des propositions floues avec des variables linguistiques ou attributs associés à des termes linguistiques définissant des sous-ensembles flous. De ce fait il est nécessaire de déterminer les variables linguistiques (attributs) ainsi que les termes linguistiques auxquels chaque variable est associée.

A. Spécification des variables linguistiques de l'antécédent et le conséquent

A1. Variables linguistiques de l'antécédent

Dans notre modélisation [MEZ 11], les variables linguistiques de l'antécédent sont les composantes colorimétriques du pixel. Puisque nous avons choisi l'espace des caractéristiques

HSV, alors *Hue* (la teinte), *Saturation* (la saturation) et *Value* (l'intensité), deviennent les variables linguistiques de notre modèle. Chacune des variables précédentes est associée à un certain nombre de termes linguistiques. Nous fixons ce nombre par exemple à trois.

D'après la définition de la variables linguistique présentée dans le chapitre de la logique floue, nous spécifions nos variables comme :

Variable 1 : (*Hue*, [0,1], T_{hue}) tel que : $T_{hue} = \{ Bash, Moyenh, Elevéh \}$.

Variable2: (*Saturation*, [0,1], $T_{saturation}$) tel que: $T_{saturation} = \{ Bass, Moyens, Élevés \}$.

Variable 3 : (*Value*, [0,1], T_{value}), tel que : $T_{value} = \{ Basv, Moyenv, Élevév \}$.

Le schéma suivant présente la fonction d'appartenance de la variable linguistique (attribut) : *Hue*

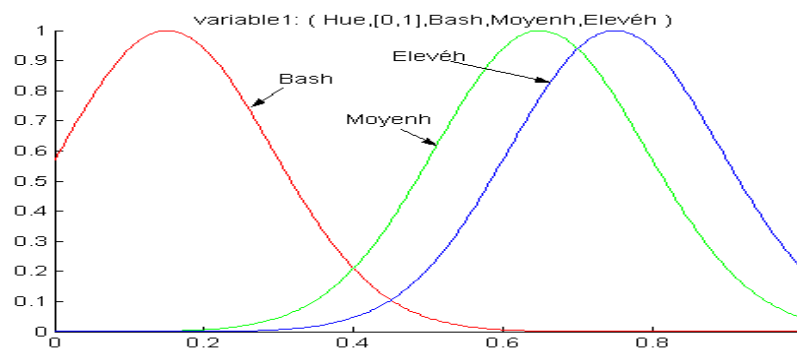


Figure 4.4 Fonctions d'appartenance de l'attribut *hue*.

A2. Variable linguistique du conséquent

Les termes linguistiques de l'attribut du conséquent sont construits à partir des centres des clusters initiaux. Par conséquent leur nombre est le même que celui des clusters.

Variable du conséquent : (*Classe*, $[0,1]^3$, T_{classe}), tel que : $T_{classe} = \{ classe1, classe2, classe3 \}$.

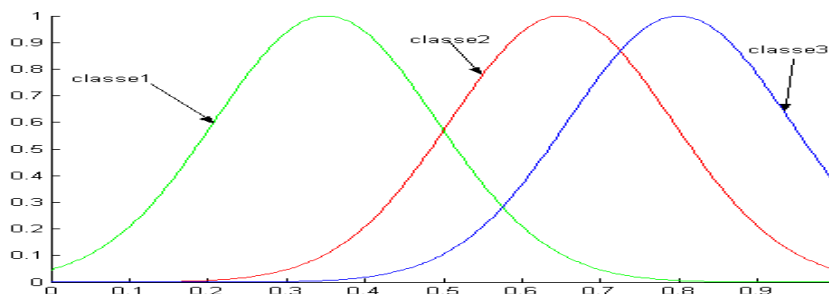
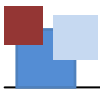


Figure 4.5 Fonctions d'appartenance de l'attribut *classe*.



Il est important de noter que :

- ✓ L'ensemble de références [0,1], sur lequel nos variables sont définies, est spécifique à MATLAB (le langage de programmation que nous utilisons pour la réalisation de notre système). C'est à dire qu'il peut être différent dans d'autres cas.
- ✓ Le nombre de termes linguistiques d'un attribut n'est pas forcément le même pour tous les attributs (par exemple : le nombre de termes linguistiques de l'attribut hue peut être égal ou non au nombre de termes linguistiques des attributs : saturation et value).

B. Composition des règles

La construction des règles initiales de la base requiert deux indications qui doivent être fixées par l'utilisateur : le nombre de clusters qui déterminera le nombre des règles et les centres des clusters ou prototypes initiaux qui seront utilisés pour écrire les règles.

Par exemple si l'utilisateur fixe le nombre de classes à trois. Il sélectionne les centres de classes initiaux recueillis illustrés sur le tableau suivant :

Centre initial	<i>Hue</i> (teinte)	<i>saturation</i>	<i>Value</i> (intensité)
y_1	0.1647	0.4980	0.0000
y_2	0.0492	0.5214	0.4000
y_3	0.8333	0.2000	1.0000

Tableau 4.1 : Initialisation des centres des clusters ou prototypes.

Alors la base de règles initiale comportera trois règles qui seront formées de la manière suivante:

- 1) Déterminer les termes linguistiques (construits auparavant) auxquels appartient chaque attribut par un degré maximal de y_i . Afin d'écrire la partie gauche (l'antécédent) de la règle i définissant le i ème cluster, prenons par exemple $y_1(0.1647, 0.4980, 0)$:

0.1647 ∈ Bash

0.4980 ∈ Moyens

0.0000 ∈ Basv

L'antécédent sera de la forme :

IF *hue* is *Bash* and *saturation* is *Moyens* and *value* is *Basv*



- 2) Déterminer la variable linguistique (l'attribut) de la partie droite de la règle i définissant le $i^{\text{ème}}$ cluster, pour le même exemple y_1 :

Le conséquent sera de la forme : THEN y_1 is *classe1*.

La règle complète sera de la forme :

IF *hue* is *Bash* and *saturation* is *Moyens* and *value* is *Bas* THEN y_1 is *classe1*

L'ensemble des règles créées de la manière exposée plus haut constitue notre modèle initial. Dans notre exemple il a la forme suivante :

IF *hue* is *Bash* and *saturation* is *Moyens* and *value* is *Bas* THEN y is *classe1*.

IF *hue* is *Moyen* and *saturation* is *Elevés* and *value* is *Moyen* THEN y is *classe2*.

IF *hue* is *Elevé* and *saturation* is *Bas* and *value* is *Elevés* THEN y is *classe3*.

3.2.2. Conception du réseau neuroflou

Dans les sections suivantes, nous implémentons notre système d'inférence flou de type Mamdani-Assilian sur un réseau neuronal constitué de cinq couches, chaque couche correspond à un des modules du système flou.

A . Fuzzification des entrées

Cette étape consiste à déterminer, par une fonction d'appartenance, de quel degré chaque entrée appartient aux termes linguistiques de la variable linguistique appropriée. L'entrée de cette phase est toujours une valeur numérique de l'univers de discours de la variable de cette entrée et la sortie est un degré flou d'appartenance de l'entrée aux termes linguistiques de la variable linguistique.

Dans notre univers de discours de la variable *hue*, si nous prenons par exemple la valeur numérique 0.4 comme une entrée, la sortie selon la **figure 4.6** est un degré flou calculé via la fonction d'appartenance exponentielle : $\mu(0.4) = 0.2$

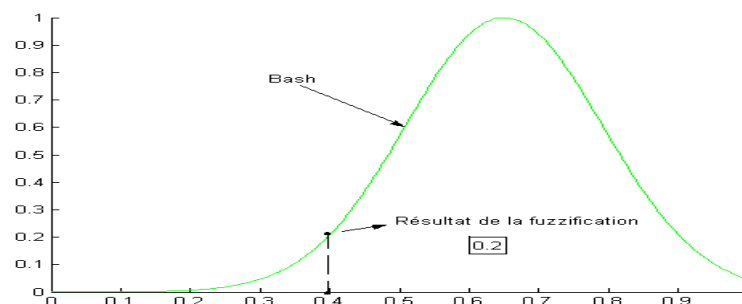


Figure 4.6 Fuzzification des entrées

De la même façon est fuzzifiée chaque entrée relative aux termes linguistiques d'une variable d'entrée requise par les règles.

L'implémentation de ce module sur le réseau de neurones est faite comme suit :

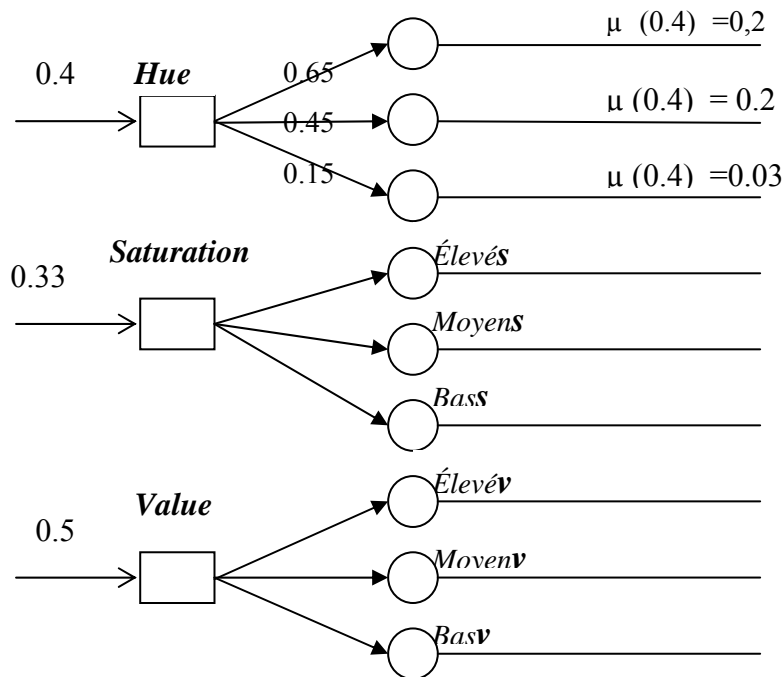


Figure 4.7 Couche de fuzzification

Dans la Figure 4.7, la première couche n'effectue aucun traitement, ce n'est qu'un buffer nécessaire pour l'alimentation du réseau. La deuxième couche correspondant à l'étape de fuzzification, est constituée d'un nombre de neurones égal au nombre de tous les termes linguistiques.

- L'entrée de chaque neurone de la couche est une fonction d'entrée totale calculée à partir de l'entrée (valeur numérique) et du poids de connexion (le centre du terme linguistique).
- La fonction de transfert est la fonction d'appartenance exponentielle.
- La sortie de la couche est un degré d'appartenance.

B. Application des opérateurs flous (AND)

Dans cette étape, l'opérateur *AND* est appliqué sur les propositions élémentaires de l'antécédent pour obtenir un nombre représentant le résultat de l'antécédent d'une règle. En d'autre terme cette évaluation correspond à l'estimation de l'équation de mise à jour de la matrice de partitions (l'appartenance d'un pixel à une classe).

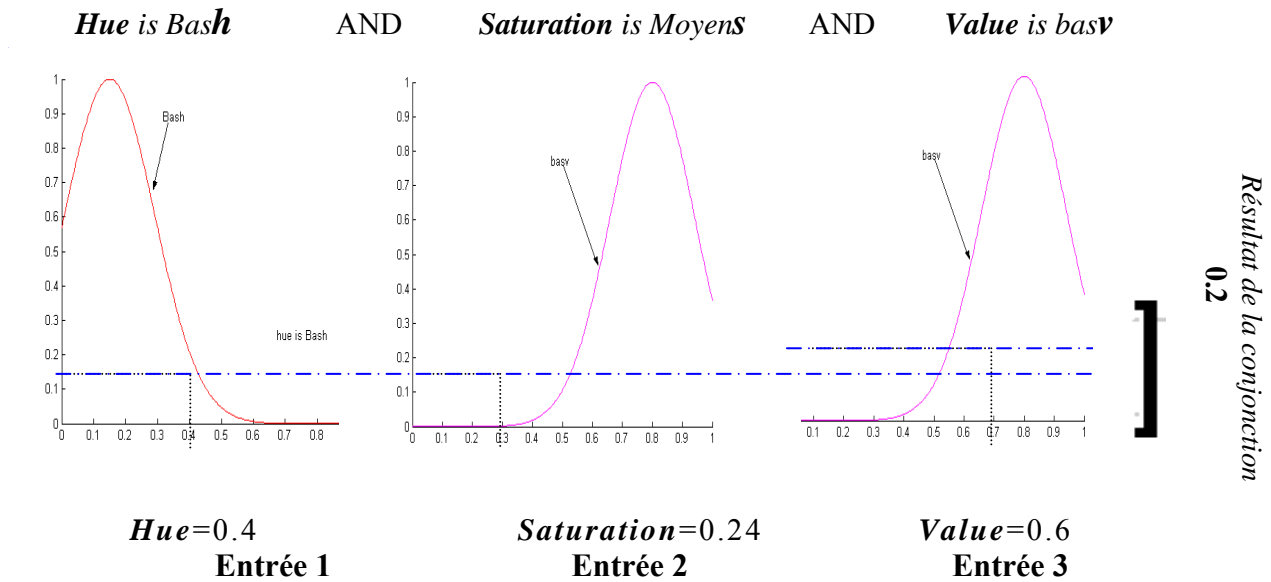


Figure 4.8 Conjunction des propositions de l'antécédent

L'implémentation du module sur le réseau de neurones est faite comme suit :

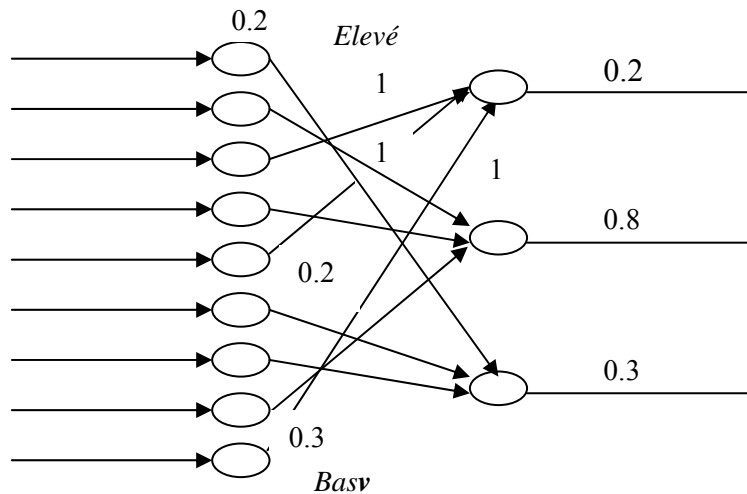


Figure 4.9 Couche de conijonction

La troisième couche correspondant à l'étape de l'application de l'opérateur flou AND dans l'antécédent est constituée d'un nombre de neurones égal au nombre de règles (conçues précédemment).

- L'entrée de chaque neurone de la couche est la fonction d'entrée totale calculée à partir des sorties de la couche précédente (degrés d'appartenance) et des poids de connections (initialisés à 1).
- La fonction de transfert est *T-norm*.
- La sortie de la couche est le degré de l'antécédent.

C. L'implication

Dans cette phase, l'entrée du module est le degré de l'antécédent et la sortie est un ensemble flou. Cet ensemble flou est le résultat de la reconstitution du conséquent de la règle floue, caractérisée par la fonction d'appartenance exponentielle, par l'utilisation du degré de l'antécédent calculé précédemment. La méthode de l'implication appliquée est la fonction du minimum qui tronque l'ensemble flou de la sortie.

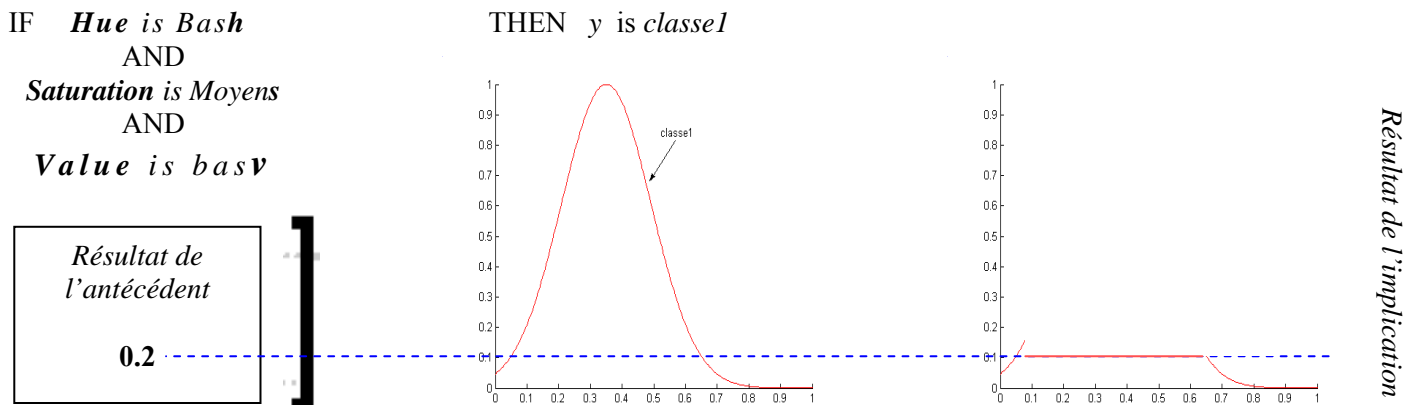


Figure 4.10 Implication de l'antécédent vers le conséquent

L'implémentations de ce module, sur le réseau de neurones est faite comme suit :

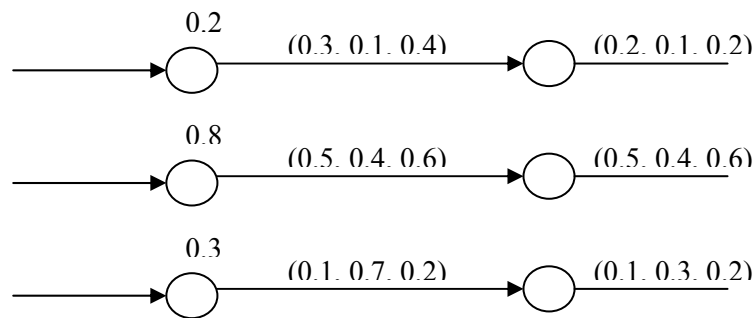


Figure 4.11 Couche de l'implication

La quatrième couche correspondant à l'étape de l'implication est constituée d'un nombre de neurones égal au nombre de règles.

- L'entrée de chaque neurone de la couche est la fonction d'entrée totale calculée à partir de la sortie de la couche précédente (degré de l'antécédent) et le poids de connexion (initialisés par le vecteur des degrés d'appartenance du centre initial).
- La fonction de transfert est *T-norm*.
- La sortie de la couche est un vecteur de degrés d'appartenance.

D. L'agrégation et la defuzzification

A cette dernière étape, tous les ensembles flous représentant les sorties des règles sont combinés afin d'obtenir un seul ensemble flou qui sera defuzzifié

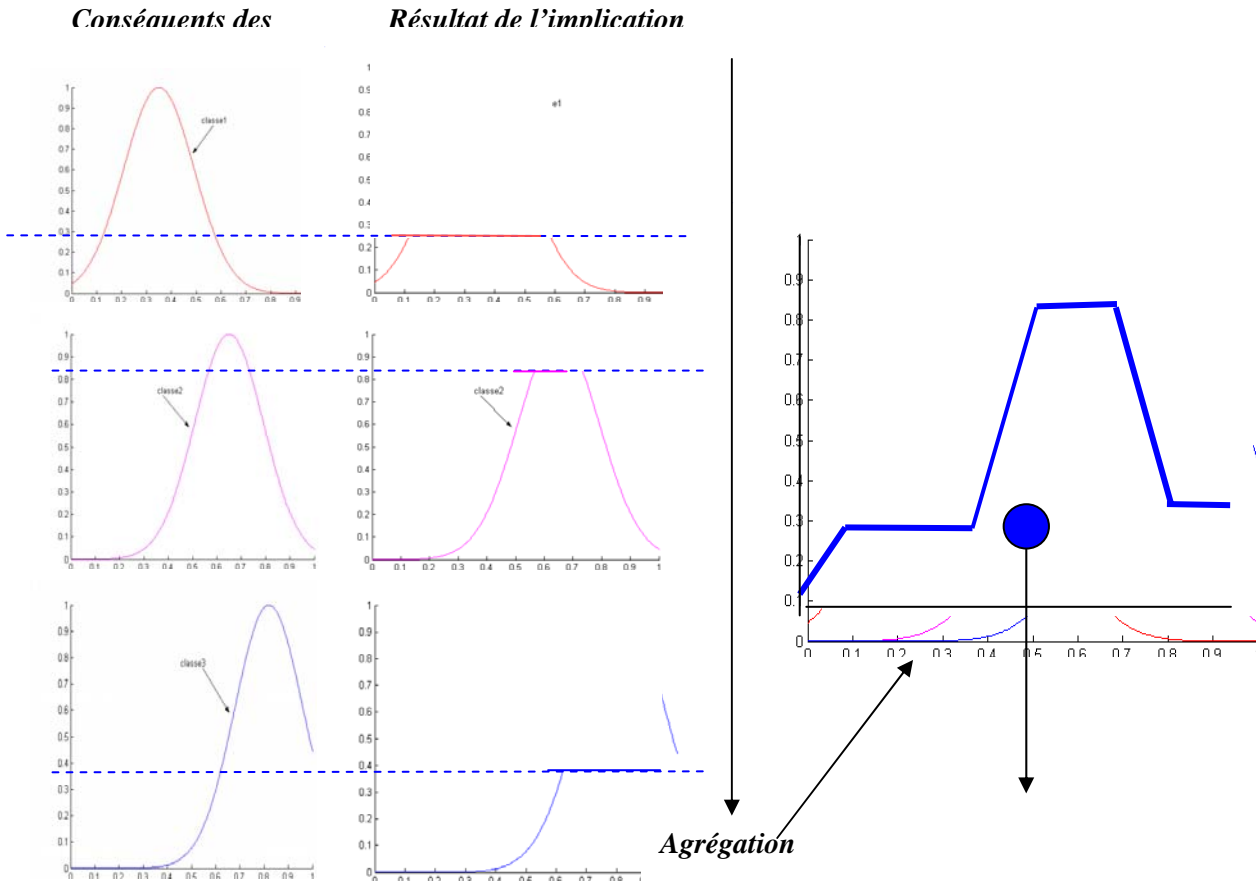


Figure 4.12 Agrégation et Defuzzification.

Ces deux opérations sont implémentées sur la même couche du réseau neuronal :

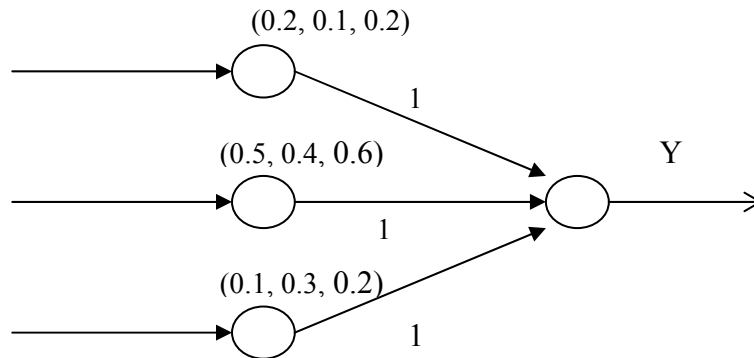


Figure 4.13 Couche d'agrégation et de defuzzification

La cinquième couche correspondant à l'étape d'agrégation et de defuzzification, est constituée d'un seul neurone.

- L'entrée du neurone est la fonction d'entrée totale calculée à partir de la sortie de la couche précédente (vecteur des degrés d'appartenance) et les poids de connections initialisés à 1.
- La fonction de transfert est la fonction de prototype *COD*.
- La sortie de la couche est une valeur numérique (le centre de la classe).

Le réseau *neuroflou* conçu, au complet, est un réseau multicouche de type feed-forward composé de cinq couches : son architecture et ses paramètres sont montrés sur la figure suivante :

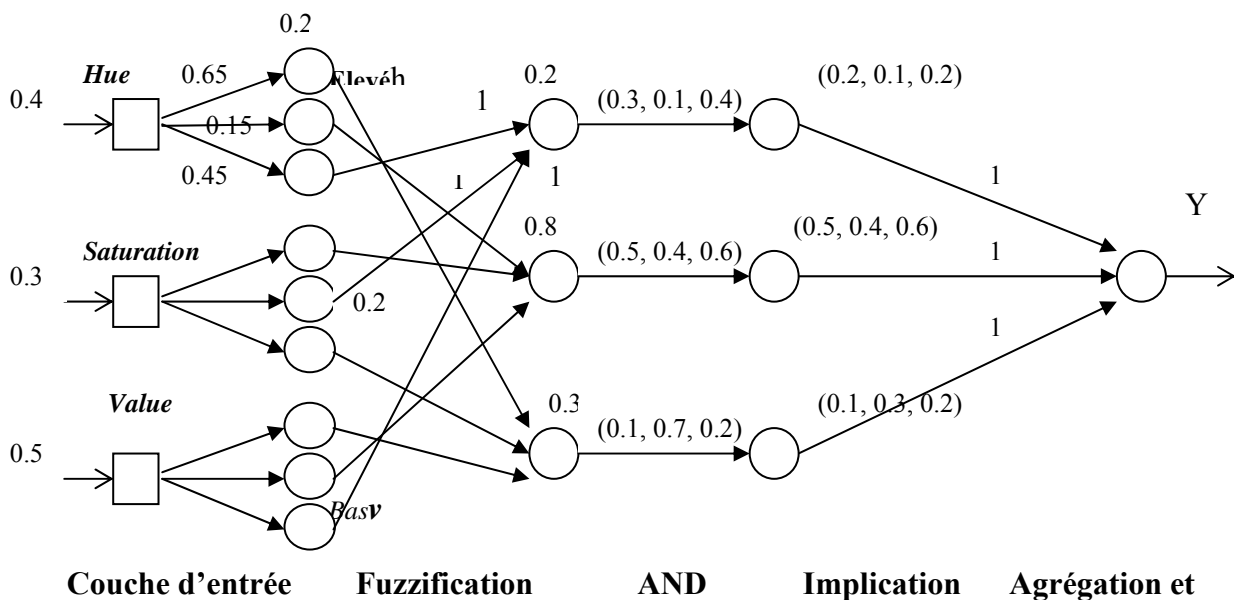


Figure 4.14 Structure du réseau neuro-flou conçu

Afin d'entraîner le réseau neuro-flou conçu, à classifier les pixels d'une image, nous sélectionnons un échantillon de pixels comme un exemple d'apprentissage et nous construisons un algorithme d'apprentissage génétique.

3.2.3 Conception d'un algorithme d'apprentissage génétique

Dans cette phase nous utilisons les algorithmes génétiques pour la construction d'un algorithme d'apprentissage supervisé spécifique au réseau neuroflou développé durant les étapes précédentes.

Un algorithme d'apprentissage est sensé modifier *automatiquement* les données du réseau c'est à dire d'*adapter* ses *poids synaptiques*, jusqu'à arriver à une solution *optimale*. Donc notre algorithme d'apprentissage génétique doit pouvoir *adapter* les poids synaptiques du réseau conçu auparavant et doit trouver la meilleure solution possible.

Pour mettre en œuvre ce concept, nous spécifions d'abord les paramètres qui doivent être adaptés. Ensuite nous décrivons la manière dont l'algorithme d'apprentissage génétique procède pour arriver à une solution optimale.

Les paramètres sujets d'adaptation sont les *poids synaptiques* de la première couche (couche de fuzzification) du réseau neuroflou. Ceux des couches suivantes n'ont pas besoin d'être optimisés directement par l'algorithme, du moment que, notre réseau neuroflou est un *réseau multicouche* où l'information se propage toujours en avant.

Afin d'arriver à une *solution optimale* l'algorithme explore un espace de solutions potentielles, itérativement, et sélectionne à chaque fois les meilleurs candidats (solutions). En ce qui nous concerne, l'espace de recherche est constitué d'un ensemble de solutions où chaque solution est une initialisation possible des centres des termes linguistiques de la couche de fuzzification du réseau.

Pour permettre le fonctionnement de notre algorithme d'apprentissage génétique, des éléments de base nécessaires doivent être définis. Ces derniers sont exposés par la suite.

- **Le gène** : représente un poids de connexion de la première couche du réseau neuroflou (couche fuzzification), c'est à dire un centre de terme linguistique. Le nombre de l'ensemble des **gènes** est au nombre des termes linguistique des variables d'entrée.
- **Les chromosomes** ou **individus** : il est constitué d'un ensemble de gène c'est à dire de l'ensemble des centres des termes linguistiques de la couche de fuzzification.
- **La Population** : elle est constituée de l'ensemble de chromosomes défini ultérieurement. chaque chromosome représente une solution potentielle d'initialisation des centres des termes linguistique de la première couche.
- **La fonction d'adaptation** ou **fitness f** : c'est la fonction qui évalue la différence entre la sortie calculée par le réseau neuroflou et la sortie désirée. Elle est de la forme suivante : $f_i = Y^{(t)} - Y^{(t-1)}$, tel que i est le $i^{\text{ème}}$ chromosome dans la population courante.

Afin que *l'algorithme d'apprentissage génétique*, puisse optimiser les poids synaptiques de la couche de fuzzification, nous effectuons d'abord l'opération de codage ensuite, nous fixons les paramètres de l'algorithme, et finalement nous détaillons chaque opérateur à part.

A. Codage des solutions potentielles

Parmi les principaux mécanismes de codage que l'algorithme génétique utilise on distingue : le codage binaire et le codage réel. Dans notre conception de l'algorithme nous optons pour le codage réel pour les deux raisons principales : le gain de temps et de l'espace.

Si nous prenons par exemple les poids de connexions montrés dans la figure 4.15 comme une solution éventuelle, le codage s'effectue comme suit :

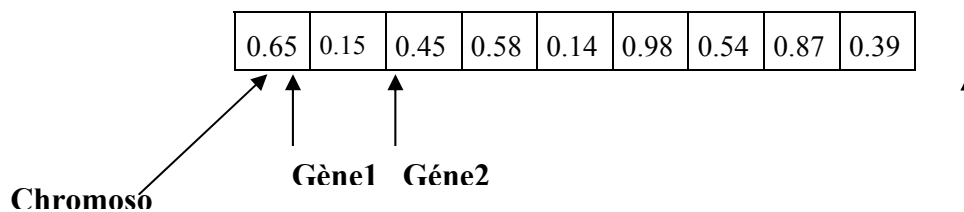


Figure 4.15 Codage réel

Comme montré dans la figure 4.15 le chromosome est une chaîne de nombres réels tels que chaque nombre (centre du terme linguistique) est un gène.

B. Paramètres de l'algorithme d'apprentissage génétique

La mise en œuvre de notre algorithme génétique nécessite le réglage de certains paramètres qui influence la réussite ou non de notre algorithme, qui sont :

Le critère d'arrêt : Comme dans tout algorithme itératif, l'algorithme d'apprentissage génétique doit avoir un critère d'arrêt. Nous utilisons le critère le plus répandu dans les algorithmes génétiques qui concerne le nombre de générations. Nous le fixons à 100.

- 1) **La taille de la population** : des valeurs pratiques intéressantes de la taille de la population sont généralement entre 20 et 100 individus. Mais dans notre cas nous fixons la taille de notre population à 15 car l'espace de solutions potentielles est réduit.
- 2) **La probabilité de mutation et de croisement** : des études ont prouvé que pour avoir de bonnes performances il faut choisir une probabilité de croisement élevée et une probabilité de mutation faible. Les valeurs de croisement (p_c) généralement admises sont comprises

entre 0.6 et 0.9 ; les valeurs de mutation (p_m) les plus utilisées sont entre 0.001 et 0.01. nous fixons (p_c) à 0.6 et (p_m) à 0.01.

C. Opérateurs génétiques

C1. Sélection

La *sélection* consiste à choisir un chromosome dans l'ensemble des chromosomes permis, après avoir évalué tous les chromosomes de la population par la fonction d'évaluation. Nous utilisons l'une des méthodes conçues pour la modélisation de l'opérateur de sélection : la roue de la fortune.

C2. Croisement

De nombreux types d'opérateurs de croisement existent, on distingue principalement : l'opérateur de croisement en un point et l'opérateur de croisement multipoint. Dans notre conception de l'algorithme nous nous intéressons au deuxième type. Le principe du croisement multipoint consiste à choisir plusieurs points de croisement et intervertir les gènes des deux parents. Nous illustrons cette opération dans la figure ci-dessous.

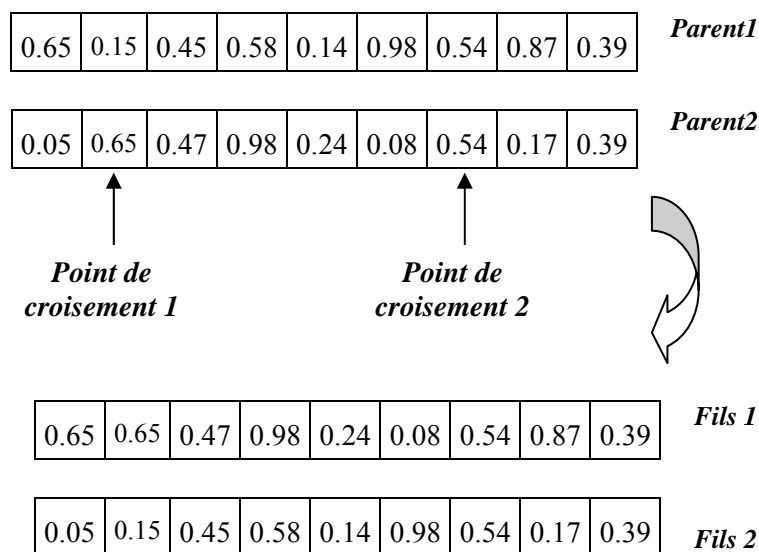


Figure 4.16 Croisement multipoint

C3. Mutation

Le principe de l'opérateur de mutation consiste à modifier aléatoirement et avec une certaine probabilité (p_m), la valeur d'un ou de plusieurs gènes d'un chromosome. Dans le cas du codage réel le gène subit une mutation comme suit : $g = g + \Psi ()$, tel que g est la valeur du

gène représenté par une valeur réelle et $\Psi ()$ représente la fonction de génération des nombres réels aléatoires dans un intervalle particulier. Nous présentons un exemple :

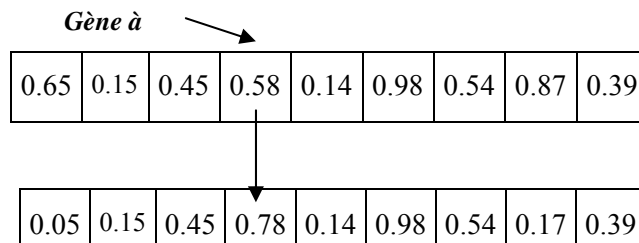


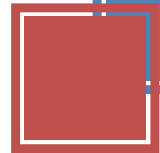
Figure 4.17 Mutation

La valeur retournée par la fonction $\Psi ()$ est de 0.20 donc le quatrième gène du chromosome subit une mutation : $g=0.58 + 0.20$.

4. Discussion

Dans cette partie, nous avons présenté l'approche proposée avec toutes les étapes de sa conception. Nous avons commencé par poser la problématique puis décrire le modèle ACE afin de l'adapter pour l'approche proposée ensuite, nous avons donné les différents schémas d'estimation du réseau neuro-flou ainsi que la structure et le mode de fonctionnement de l'algorithme d'apprentissage génétique.

PARTIE 2 :
PRÉSENTATION ET ANALYSE DES RÉSULTATS



1. Introduction

La segmentation d'images médicales est une opération cruciale dans les problèmes suivants [DOM 09] :

1. Imagerie fonctionnelle : quantification des volumes des tissus, des organes.
2. Aide au diagnostic d'une maladie.
3. Localisation de la pathologie.
4. Etude d'une structure anatomique.
5. Planification de traitement.
6. Chirurgie assistée par ordinateur.

Pour valider la méthode développée, nous nous sommes basés sur deux sources : la première est la base de données Brainweb5. Elle a été choisie dans la mesure où elle est très fréquemment utilisée dans la littérature. Elle permet donc de pouvoir fournir un point de comparaison plus aisé avec des validations proposées dans d'autres documents. Le site Web de Brainweb permet de simuler des IRM cérébrales avec différents niveaux de bruit et d'inhomogénéités [KWA 06]. La deuxième source est l'ensemble d'images médicales fournies par le centre d'imagerie médical privé CIM et le centre hospitalier de Constantine.

Notre but principal dans cette section est de dérouler l'application implémentée sur différentes images médicales en changeant les paramètres, afin de détecter et fixer les meilleurs pour l'obtention d'une bonne segmentation.

Nous remarquons que le nombre des clusters, la fonction d'appartenance (membership) et la fonction de prototype influencent largement sur la qualité de la segmentation.

2. Fonction d'évaluations

Plusieurs méthodes de segmentation ont été développées ces dernières décennies et les chercheurs ont ressenti le besoin d'avoir une mesure de qualité de la segmentation, permettant de l'évaluer et de comparer entre elles la variété de procédures disponibles [BAR 07].

Les méthodes de mesure de la qualité de la segmentation qui existent actuellement peuvent être regroupées en trois grandes catégories :

- Les techniques d'évaluation par un expert humain
- Les techniques d'évaluation avec référence où l'on fait appel à des images « vérité terrain », construites par un expert du domaine de l'application.
- Les techniques d'évaluation sans référence, dans le cas où l'on ne dispose pas de la vérité terrain, et qu'il n'ya pas suffisamment d'à priori sur l'image [CHA 04].

Certaines techniques d'évaluation mesurent un indice de « bonne qualité » de la segmentation. Cet indice est d'autant plus élevé que la segmentation est bonne. D'autres techniques mesurent la « différence » entre l'image segmentée et l'image de référence. Dans ce

cas cette différence est d'autant plus faible que la segmentation est bonne (proche de la référence). Mais en ce qui concerne la référence, elle ne peut être définie de façon unique que si l'image est simple [CHE 08].

Pour comparer de façon objective les résultats ou méthodes de la segmentation, il est préférable d'utiliser des images de synthèse dont les contours et les régions sont parfaitement localisés. Cependant de telles images ont l'inconvénient de ne pas reproduire toutes les situations possibles et de ne pas tenir suffisamment compte de la réalité [MOO 02].

Pour notre approche, l'évaluation a été utilisée comme index par la formule de la ressemblance de Jaccard [JCF 10], cette formule représente le quotient de l'intersection des deux ensembles par leur union. Formellement, elle est définie comme suit :

$$J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|} \quad 4.2$$

S_1 : représente une classe dans la segmentation à évaluer I_{seg} .

S_2 : représente une classe dans la segmentation de référence I_{ref} .

La similarité entre deux segmentations représente l'ensemble des indices de Jaccard exprimant les taux de correspondance de chaque classe de la segmentation de référence S_2 dans S_1 . [SLI 08].

3. Résultats expérimentaux et discussions

Nous avons utilisé pour la réalisation du logiciel, MATLAB[®], de version R2009b. Nous avons été orientés dans nos analyses par un radiologue dans une clinique privée (Centre d'imagerie médicale -Constantine) et d'autres analyses d'experts à l'hôpital (CHU-Constantine)

3.1 Images utilisées

3.1.1 Images réelles

Les images réelles sur lesquelles nous avons travaillé ont été acquises dans le centre d'imagerie médicale privé CIM et l'hôpital CHU de Constantine. Ces images sont sous format DICOM 6(Digital Imaging and Communications in Medicine).

3.1.2 Fantômes (Images de synthèse)

Nous avons utilisé un fantôme qui est ici une base de données synthétique. Le modèle anatomique du fantôme consiste en un ensemble de volumes flous représentant des degrés d'appartenance aux différents tissus constituant l'image IRM (matière blanche, matière grise,

LCR, peau, crâne, graisse, etc.). La construction du fantôme est fondée sur un ensemble de 27 volumes IRM de taille $181 \times 127 \times 181$ voxels par volume et de haute résolution (1mm / voxel). Ces images ont été recalées et un volume IRM a été créé par moyennage de ces 27 volumes recalés. Les voxels du volume moyen ont été étiquetés par un neuroradiologue en MG, MB, LCR, graisse, etc. Les méthodes de classification floue ont été effectuées sur ce volume. Après corrections manuelles des résultats obtenus, les cartes floues des tissus sont finalement construites. En illustration, la figure 4.21 présente une coupe des cartes floues de MG, MB, LCR et du crâne. [ZOU 08]

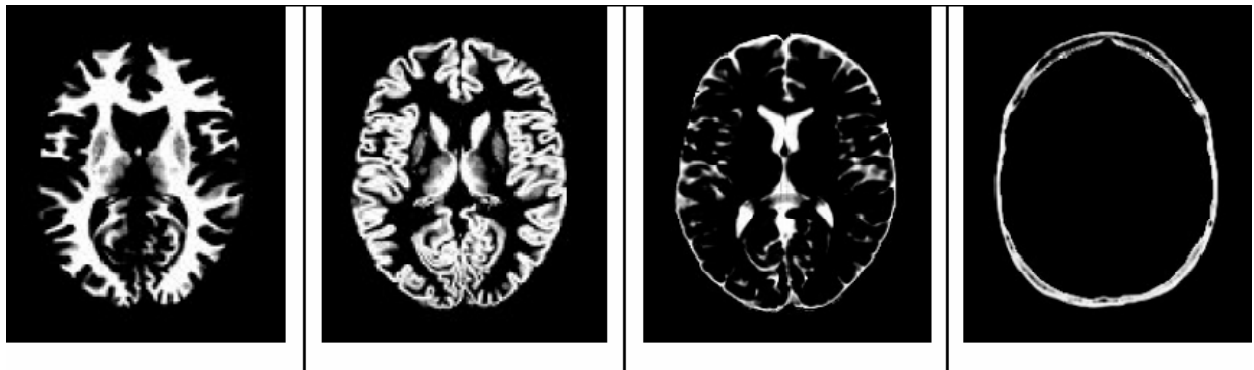


Figure 4. 18 : Exemple de cartes floues du fantôme [KWA 96].

3.2 Expériences en niveau de gris

3.2.1 IRM Cerveau humain (sain)

“Fig. 4.19.a” présente la coupe horizontale de l'image du cerveau humain.

“Fig. 4.19.b”, “fig. 4.19.c”, fig. 4.19.d”, “fig. 4.19.e” et “fig. 4.19.f” présentent les différents résultats de la segmentation de l'image du cerveau. Le premier résultat est obtenu en appliquant la fonction triangulaire avec les paramètres égaux à 2,5 pour Alpha ,1 pour sigma et 3 pour le nombre de clusters. Le second résultat est obtenu en appliquant la fonction d'appartenance exponentielle et les mêmes paramètres que précédemment

Nous remarquons :

Fig 4.19.b : Absence de visualisation de la substance grise.

Fig 4.19.c : augmentation de la détection de la substance grise,

L'espace sous arachnoïdien normalement invisible paraît exagéré.

Le sinus longitudinal supérieur est libre de toute attache.

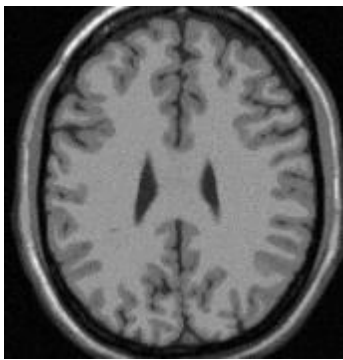
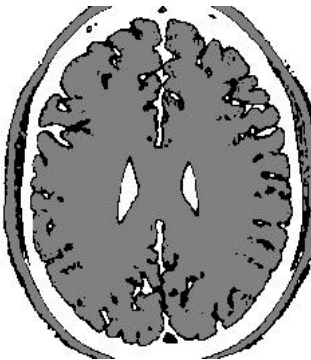
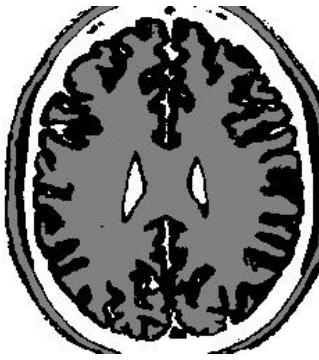
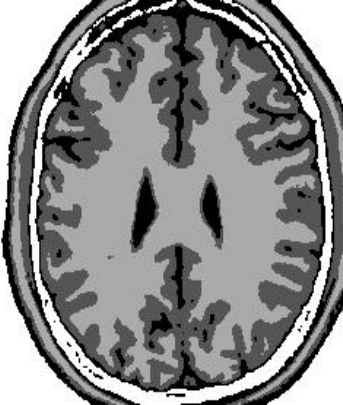
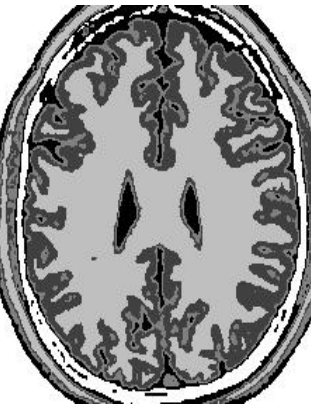
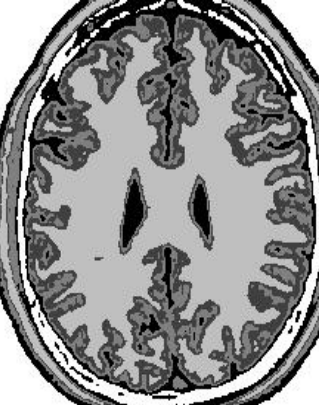
Fig 4.19.d : bonne détection de la substance grise.

Proche de l'image originale.

Fig 4.19 e et 4.19.f : la paroi ventriculaire est épaisse

Les limites sont plus nettes

L'image est sans artéfacts.

		
<p>(a) Coupe Axiale du cerveau</p>	<p>(b) Triangular membership function Parameters : alpha=2,5, sigma=1 Nbr_clust=3; $J(S_1, S_2)$Evaluation = 0,567</p>	<p>(c) Exponential membership Function Parameters : alpha=2,5, sigma=1 Nbr_clust=3 ; $J(S_1, S_2)$Evaluation = 0,542</p>
		
<p>(d) La charge de l'adhésion exponentielle Parameters : alpha=2, sigma=0.7 Nbr_clust=4; $J(S_1, S_2)$Evaluation = 0,456</p>	<p>(e) Exponential membership function- Parameters : alpha=2,5, sigma=1 Nbr_clust=5; $J(S_1, S_2)$Evaluation = 0,245</p>	<p>(f) Exponential membership function Parameters : alpha=2, sigma=0.7 Nbr_clust=5; $J(S_1, S_2)$ Evaluation = 0,257</p>
<p>Figure. 4.19- Segmentation d'une coupe axiale du cerveau humain avec des différents paramètres.</p>		

Discussion :

On constate que le choix du nombre de clusters influence sur les résultats de la segmentation, la fonction exponentielle donne de meilleurs résultats que la fonction triangulaire. Les valeurs 2 et 0,7 pour Alpha et sigma donnent une respectivement une meilleure segmentation.

3.2.2 IRM Cerveau humain (avec une Tumeur)

“Fig. 4.20.a” présente une coupe coronale du cerveau avec une tumeur intracrânienne et plus précisément une tumeur du noyau du postérieur d'un malade âgé de 51 ans.

“Fig. 4.20.b”, “fig. 4.20.c”, fig. 4.20.d”, “fig. 4.20.e” et “4.20.f” présentent les différents résultats pour la segmentation de la même image.

Fig. 4.20.b : Absence de visualisation des sillons corticaux sous arachnoïdien ainsi que la tente du cervelet.

Ne reflète pas l'état réel de pression intra crânienne

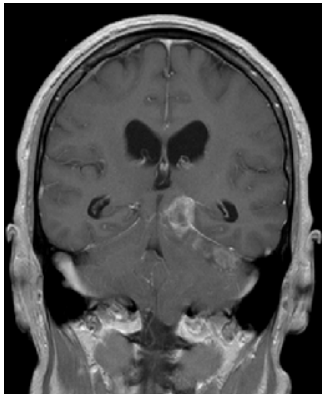


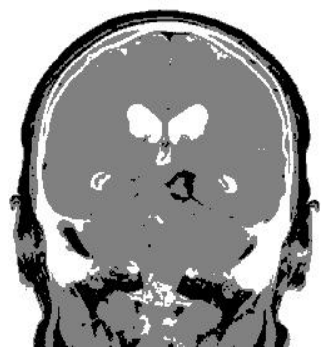


Fig. 4.20.c : Structure anatomique et tumeur sont les mieux discernées.

Fig. 4.20.d : Manque de détails

Fig. 4.20.e : Image parasitée, pigmentation punctiforme intra ventricule.

N'exprime pas la réalité.

Fig. 4.20.f : Bonne détection de la tumeur ainsi que d' autres composants anatomiques.

		
<p>(a) IRM du cerveau avec une tumeur intracrânienne : Coupe coronale</p>	<p>(b) Fonction : Triangulaire - Paramètres : $\alpha=2, \sigma=0.7$ Nbr_clsse=3, $J(S_1, S_2)$Evaluation = 0,499</p>	<p>(c) Fonction de partition : Exponentielle Paramètres : $\alpha=2, \sigma=0.7$ Nbr_classe= 4 ; $J(S_1, S_2)$Evaluation = 0,675</p>
		
<p>(d) Fonction de partition : Triangulaire Parameters : $\alpha=2, \sigma=0.7$ Nbr_clsse=4; $J(S_1, S_2)$Evaluation = 0,5</p>	<p>(e) Fonction de partition :Triangulaire Nbr_clsse=5; $J(S_1, S_2)$Evaluation = 0,521</p>	<p>(f) Fonction de partition : Exponentielle Nbr_clsse=5; $J(S_1, S_2)$Evaluation = 0,576</p>
<p>Figure. 4.20- Segmentation d'une coupe coronale du cerveau humain avec différents paramètres.</p>		

Discussion:

Quand le nombre des clusters est trois, nous notons un encombrement de la matière blanche alors que la matière grise est presque absente. Le Liquide céphalo-rachidien (LCR) qui est représenté par la couleur blanche est parfaitement distingué du reste de la structure. Nous remarquons aussi une tâche de couleur sombre, presque située au milieu de la matière blanche.

L'interprétation des composants est complètement améliorée quand le nombre de clusters est cinq. Une bonne détection de la matière grise, la matière blanche et le LCR, la tumeur est presque distinguable par la couleur grise claire au centre du cerveau.

Vus les résultats obtenus dans cette expérience aussi, il est clair que la fonction exponentielle fournit un bon résultat. Donc dans ce que suit, on se contente de travailler seulement avec cette fonction.

3.2.3 Coupe axiale du cerveau en T2**Image originale :**

- Une tumeur type : méningiome Sphénoïdale droit en signal intermédiaire avec des zones en hyper signal refoulant les structures latérales et les structures médianes vers la gauche.
- Système ventriculaire et espaces sous arachnoïdiens en hyper signal
- Substance grise.
- Substance blanche.
- Le Cuir chevelu.

IRM en 4 classes :

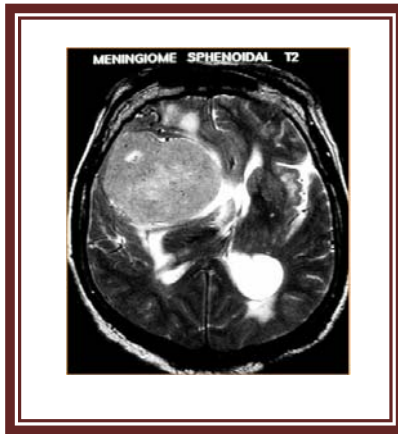
- Masse est plus nette(tumeur)
- Système ventriculaire et les espaces sous arachnoïdiens sont bien délimités.
- Substance blanche et substance grise sont bien systématisées(Artéfactées)

IRM en 5 classes :

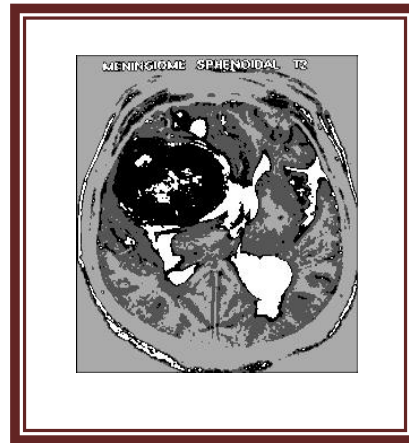
- L'image est parasitée.
- Les limites des structures anatomiques sont moins précises.
- La tumeur bien délimitée.

IRM en 6 classes :

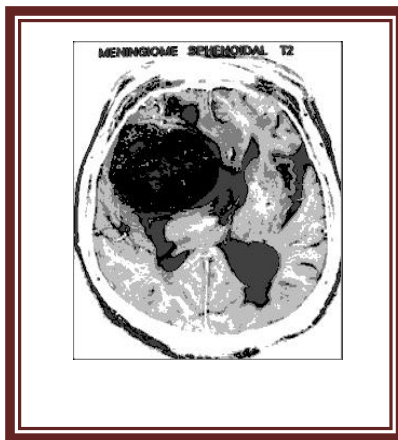
- Image sur-segmentée.



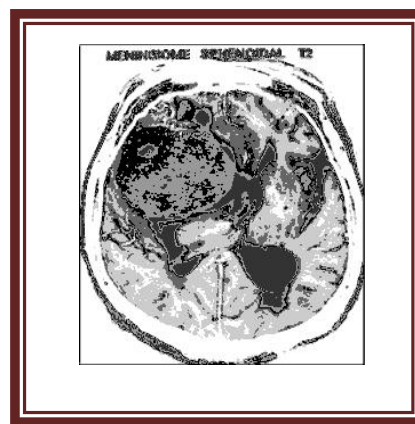
A : Image Originale



B : Image segmentée en 4 classes ; Evaluation = 0,148



C : Image segmentée en 5 classes, $J(S_1, S_2) = 0,168$



D : Image segmentée en 6 classes, $J(S_1, S_2) = 0,232$

Figure. 4.21- Segmentation de la coupe axiale du cerveau en T2 avec différentes classes

Discussion

La meilleure segmentation ici est le résultat obtenu en 4 classes.

3.2.4 Coupe axiale du cerveau en T1

Image originale(A) :

- Tumeur en hyper signal
- Système ventriculaire
- Cellules éthmoïdales
- Orbites
- Os

IRM segmentée en 4 classes(B) :

- Les différentes structures anatomiques sont nettes.
- Les détails sont bien visibles.

IRM segmentée en 5 et 6 classes(C et D) :

- Les détails ont noyés.
- Image très parasitée.

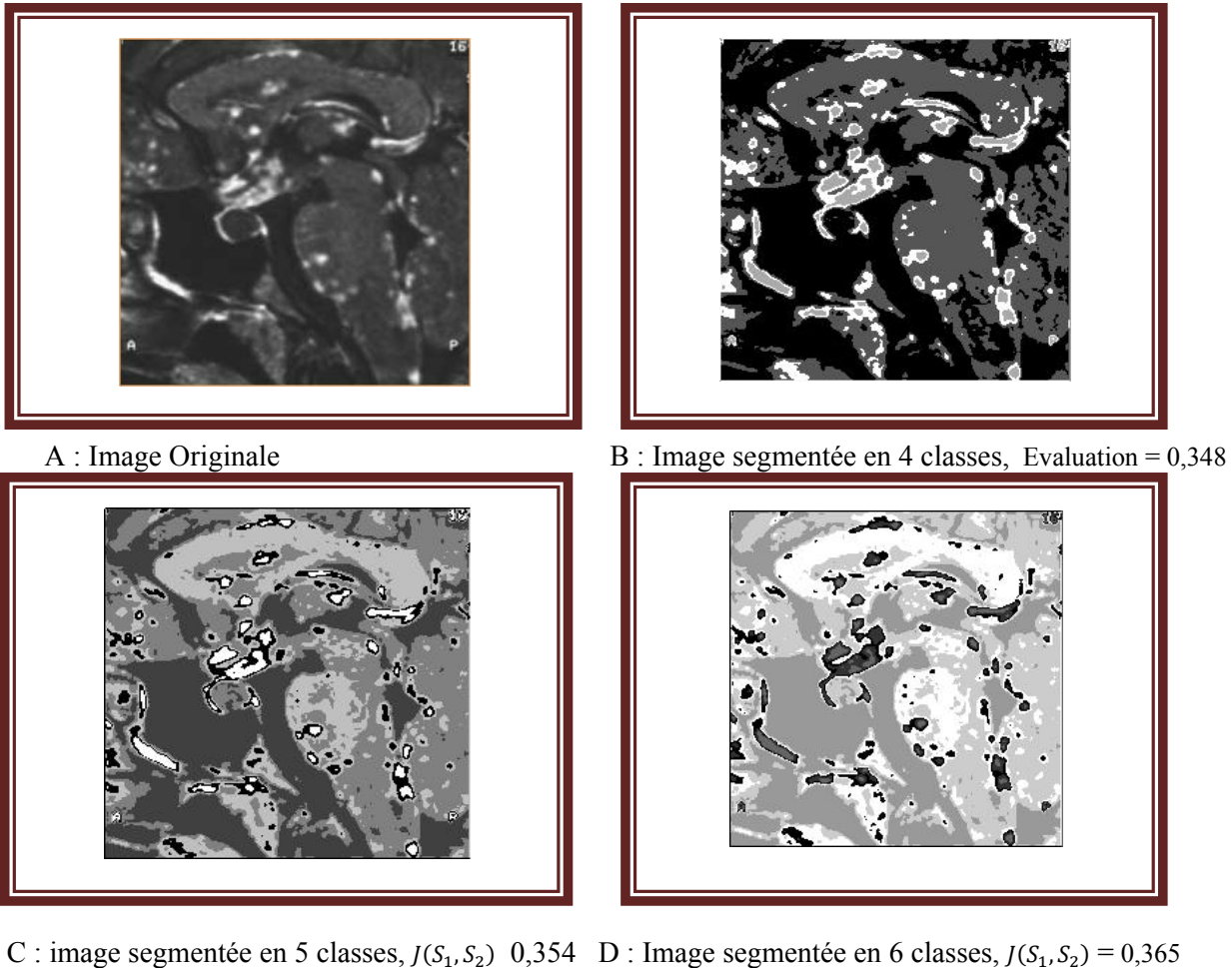


Figure. 4.22- Segmentation de la coupe axiale du cerveau en T1 avec différentes classes.

3.2.5 Coupe Sagittale du cerveau

Image originale (A) :

- Multiple modules en hyper signal diffusé en niveau du cerveau, du cervelet et du rachidien correspondant à des localisations cérébrales d'une sarcoïdose (neuro sarcoïdose qui est une maladie du système nerveux)
- LCR (les ventricules, les citernes, les espaces sous arachnoïdiennes)
- Matière blanche, matière grise.

Image segmentée en 4 classes(B) :

- Les structures anatomiques sont bien visibles.
- Les lésions de la neuro sarcoïse sont mieux délimitées.

Image segmentée en 5 classes(C) :

- Les substances grise et blanche sont mieux visualisées.
- L'image est plus détaillée.

Image segmentée en 6 classes(D) :

- Pas une grande différence avec 05 classes.

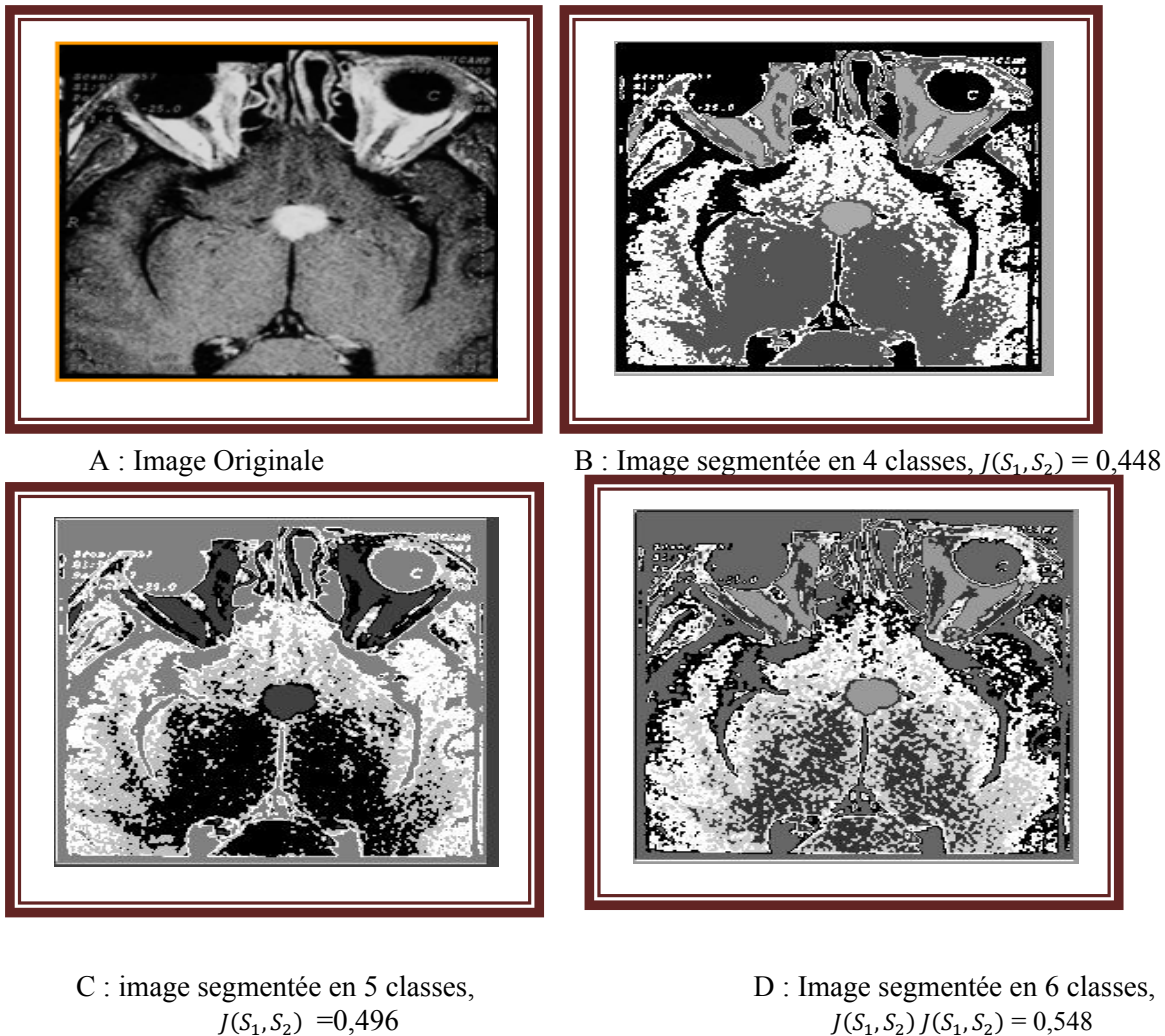


Figure. 4.23- Segmentation de la coupe sagittale du cerveau avec différentes classes.

4. Expériences en couleur

4.1 Image cérébrale SPECT

“Fig. 4.24.a” est l’image cérébrale SPECT qui représente une coupe axiale du cerveau avec une pathologie au niveau de la région temporelle gauche qui déplace le troisième ventricule vers la droite légèrement (peut évoquer une tumeur ou un accident vasculaire).

La segmentation de cette image SPECT nous a aidés à détecter les milieux liquidiens, LCR. On constate en plus une zone sombre dans la région temporelle gauche, de forme irrégulière en occipitale droite entourée par un afflux liquidien péri lésionnel hyper actif.

“Fig. 4.24.b” : obtenu par la combinaison de fonctions de prototype : MOM (Mean value of Maximum) pour l'Intensité, LOM pour la teinte et le SOM (Smallest value of maximum) pour la Saturation.

Tumeur mieux cernée (délimitée)

Edème mieux visualisé

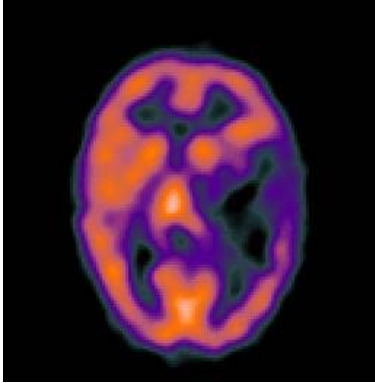
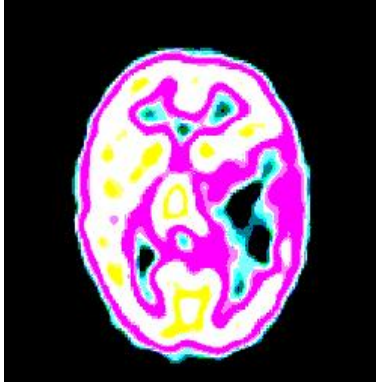
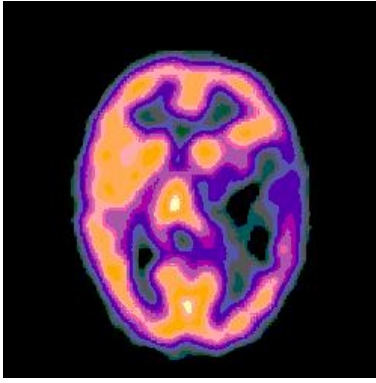
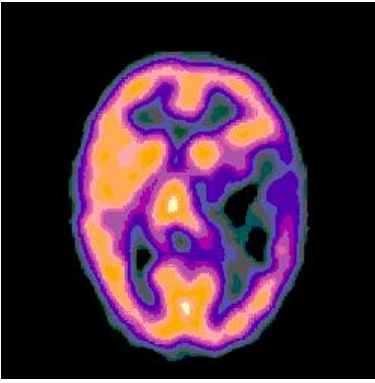
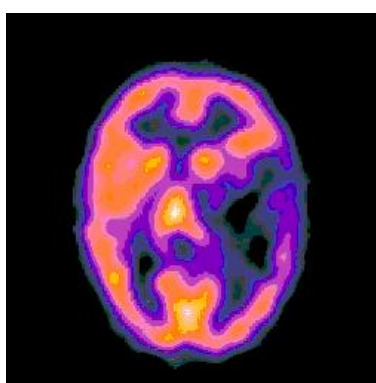

“Fig. 4.24.b”, “Fig. 4.24.c”, Fig. 4.24.d.” et “Fig. 4.24.e” : obtenus par la variation de la combinaison de la fonction du prototype concernant les trois composante de la couleur,

Tumeur bien cernée.

Œdème mal délimité.

Discussion:

D'après les résultats obtenus et par l'application de la formule de ressemblance de Jaccard, l'évaluation de la meilleure segmentation est obtenue par les paramètres: MOM, MOM, SOM.

		
<p>Figure. 4.24. a : Image SPECT Coupe axiale du cerveau</p>	<p>Figure.4.24. b Fonction Exponentiel Paramètres : alpha=2, sigma=0,7 fonction Prototype : Lom, Mom, Som Nbr_classe=5; $J(S_1, S_2) = 0,543$</p>	<p>Figure. 4.24.c Fonction Exponentiel Paramètres : alpha=2, sigma=0,7 Fonction Prototype ; Lom, Lom, Som Nbr_classe=5; $J(S_1, S_2) = 0,337$</p>
		
<p>Figure. 4.24.d- fonction ;Exponentiel Paramètres : alpha=2, sigma=0.7 fonctions de Prototype Mom, Lom, Som Nbr_clust=5; $J(S_1, S_2) = 0,248$.2.1.a.1.1</p>	<p>Figure. 4.24.e- Fonction :Exponentiel Paramètres : alpha=2, sigma=0.7 Fonctions de Prototype; om, Som, Som Nbr_clust=5 ; $J(S_1, S_2) = 0,289$</p>	<p>Figure.4.24.f- Fonction : Exponentiel Paramètres : alpha=2, sigma=0.7 Fonctions de prototype: Mom, Mom, Som Nbr_clust=5 ; $J(S_1, S_2) = 0,256$.2.1.a.1.2</p>
<p>Figure. 4.24 : Segmentation d'image spect (coupe axiale du cerveau humain)</p>		

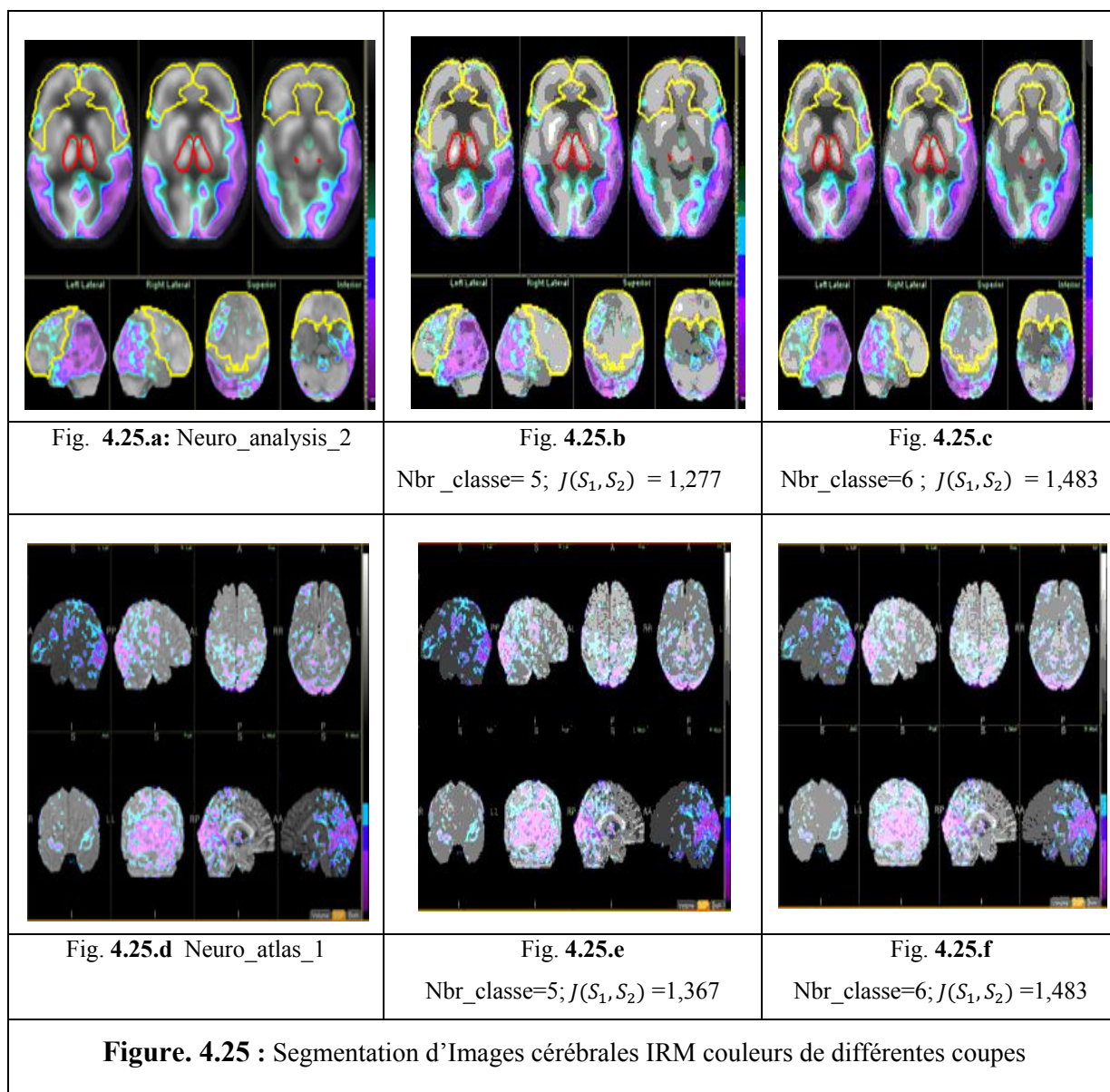
4.2 Images cérébrales IRM couleurs de différentes coupes

Pour cela, à partir de la figure **Fig. 4.25.a** jusqu'à la figure **Fig. 4.25.f**, nous allons présenter la segmentation des différentes images médicales avec deux nombres différents de clusters, en calculant à chaque fois leur évaluation et en appliquant les meilleurs paramètres détectés et fixés comme suit:

- La fonction d'appartenance exponentielle
- Les paramètres: $\alpha=2,5$, $\sigma=1$,
- La fonction de prototype : Mom, Mom, Som

Discussion:

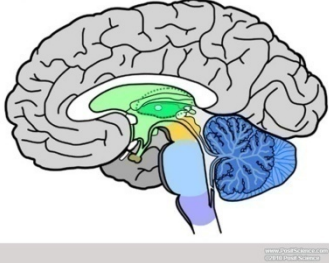
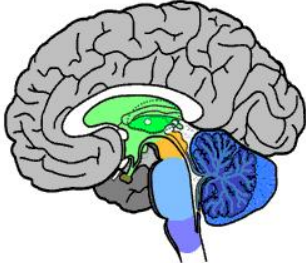

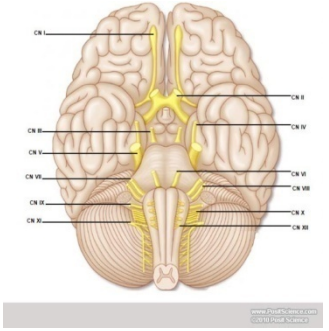
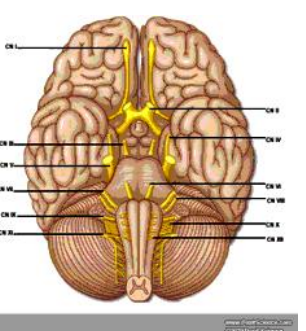
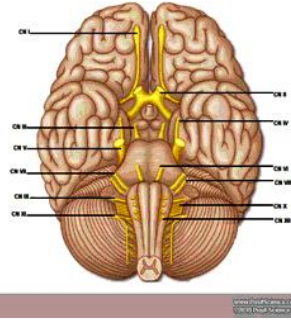
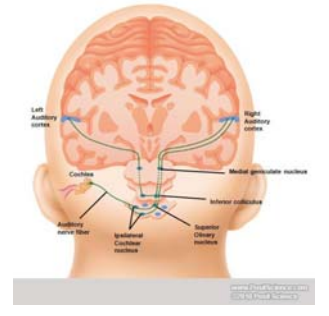
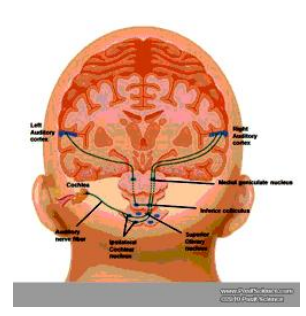
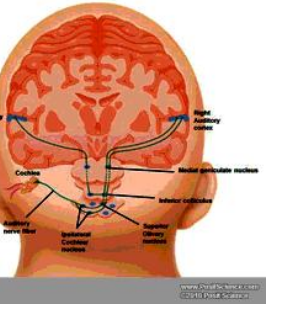
Nous avons une bonne détection de la matière blanche et grise avec une netteté du contour de chaque composant du cerveau. La représentation est presque similaire à l'originale et les résultats obtenus sont satisfaisants.



4.3 Images Couleurs anatomiques

Pour le reste, nous allons appliquer seulement la fonction exponentielle avec les paramètres égaux à 2 pour alpha et 0, 7 pour sigma.

Les résultats sont présentés par les Fig 4.26 à Fig 4.28.

		
<p>Fig. 4.26 Brain X section</p>	<p>Fig. 4.26.a Nbr_clust=5; $J(S_1, S_2) = 1,483$</p>	<p>Fig. 4.26.b Nbr_clust=6; $J(S_1, S_2) = 1,367$</p>
<p>Figure. 4.26 : Segmentation de l'image anatomique (Brain X section)</p>		
		
<p>Fig. 4.27. cranial_nerve</p>	<p>Fig. 4.27.a Nbr_clust=6; $J(S_1, S_2) = 1,483$</p>	<p>Fig. 4.27.b Nbr_clust=8; $J(S_1, S_2) = 1,687$</p>
<p>Figure. 4.27 : Segmentation de l'image : Cranial_nerve</p>		
		
<p>Fig. 4.28 Auditory-pathway</p>	<p>Fig. 4.28.a Nbr_clust=6; $J(S_1, S_2) = 1,483$</p>	<p>Fig. 4.28.b Nbr_clust=8; $J(S_1, S_2) = 1,687$</p>
<p>Figure. 4.28 : Segmentation de l'image : Auditory-pathway</p>		

Discussion:

- Les différentes structures anatomiques sont mieux visualisées sur les images segmentées.
- Le contraste est meilleur.

5. Différentes images médicales

Fig 4.29.a : Représente une coupe axiale abdominale.

Fig. 4.29.b et Fig. 4.29 : Exposent la segmentation en 8 et 6 classes respectivement.

- Les organes dans ces images sont mieux individualisés (Reins, foie, Aorte ...etc.).
- L'évaluation automatique montre que la meilleure segmentation est en 6 classes.

Fig. 4.30. A : Présente une coupe sagittale couleur du cerveau humain.

Fig. 4.30.b et 4.30.c sont les résultats de la segmentation en 5 et 6 classes :

- Les structures anatomiques sont mieux visualisées dans les images segmentées (hémisphère cérébrale, cervelet ...etc.).
- Pas grande différence entre les deux résultats obtenus mais d'après l'évaluation, on juge que la bonne segmentation est en 5 classes.

Fig. 4.31. a présente une coupe axiale couleur du pelvis.

Fig. 4.31.b et Fig. 4.31.c : Les images segmentées en 5 et 6 classes sont presque identiques à l'image originale. La détection des différents organes est bonne.

Fig. 4.32 : Présente une coupe sagittale d'IRM en T1 avec méningiome frontale.

On distingue :

- Méningiome frontal en hyper signal avec œdème perilésionnel.
- Substance grise, substance blanche.
- Ventricules, citernes(LCR)
- Os, Sinus.

Fig. 4.32. A :

- Tumeur bien délimitée.
- L'œdème perilésionnel mal individualisé.
- Substance grise et la substance blanche mal différenciées.

Fig. 4.32. B :

- Tumeur bien visible ainsi que l'œdème perilésionnel.
- L'image est meilleure.

Fig. 4.33 : Coupe sagittale d'IRM en T2.

On distingue

- Système ventriculaire en hyper signal.
- Cervelet.
- Substance grise et blanche
- Face et Os.

Fig. 4.33.a :

- Substances blanche et grise sont bien différenciées.
- Les autres structures anatomiques sont bien visibles.

Fig. 4.33.b :

- Substance blanche et grise sont moins différenciées.

Fig. 4.34 : Coupe axiale d'IRM au niveau de la base du crâne.

- Sinus maxillaires mentionnés par (1 et 2).
- Sinus Sphénoïdale(3).
- Lobe temporale(4)
- Cervelet (5) et Citernes et Fosse nasales(6).

Fig. 4.34. a :

- Bonne détection de la Substance grise et blanche.
- Complément de la fosse nasale droite.

Fig. 4.34. b :

- Détection de la substance blanche et grise est médiocre.
- Sillions du cervelet ne sont pas visibles.
- La segmentation en 6 classes présentée par la Fig. a est meilleure.

Fig. 4.35 : Coupe axiale anatomique de la base du crane sur cadavre.

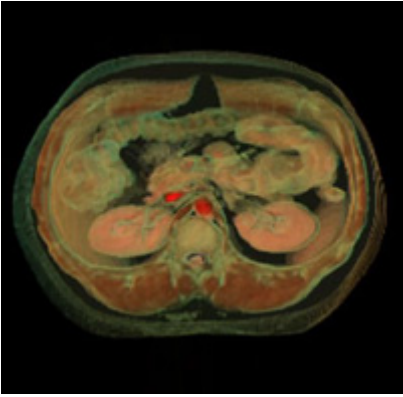
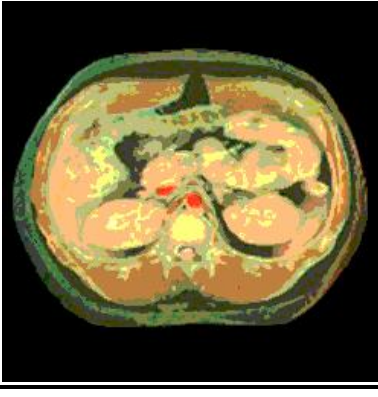
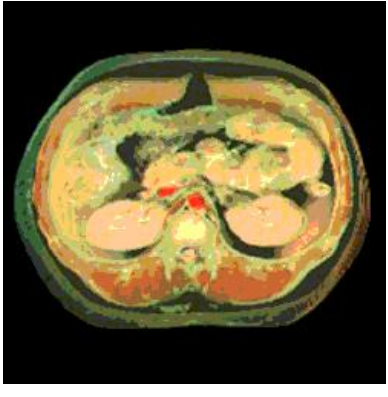
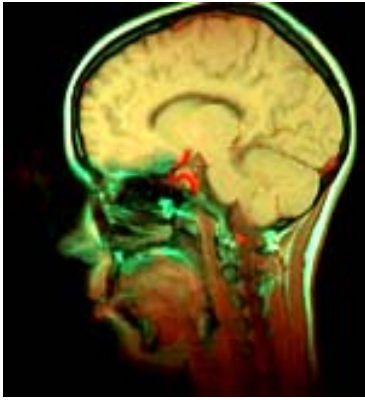
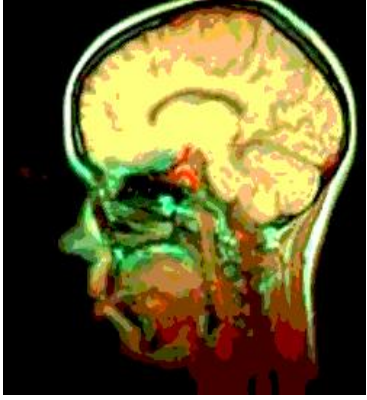
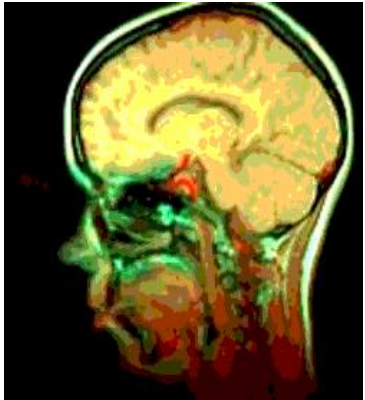
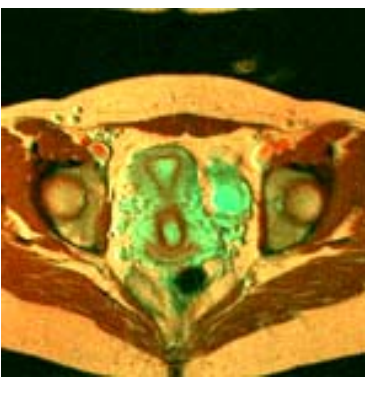
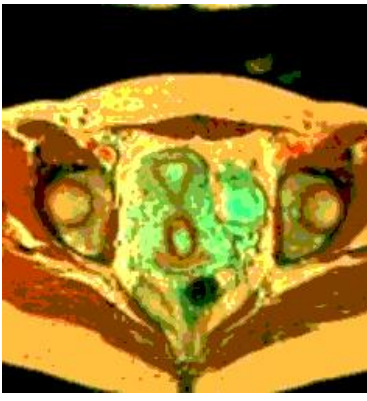
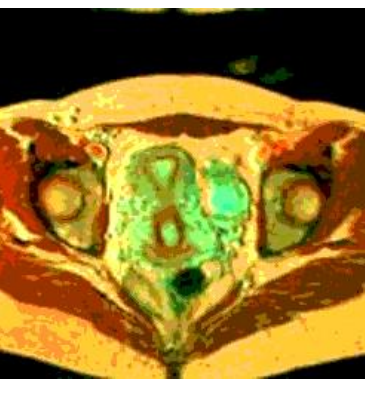
- Tumeur brunâtre de la selle turcique de volume important.
- Structures osseuses de la base du crâne
- Parenchyme cérébral, artères, veines.

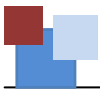
Fig. 4.35. a :


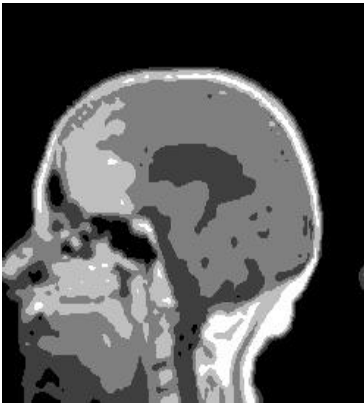


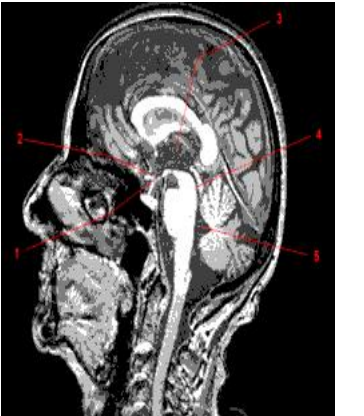

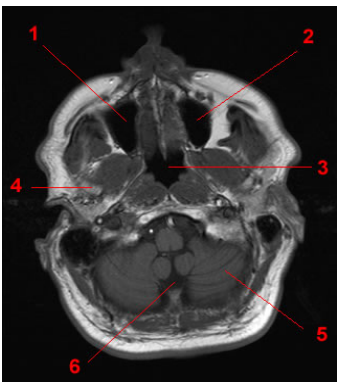
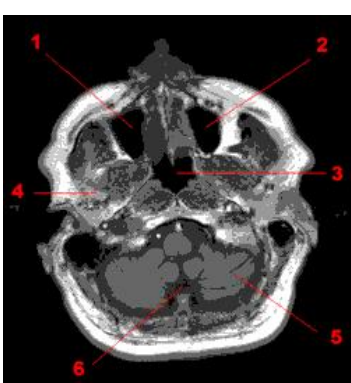
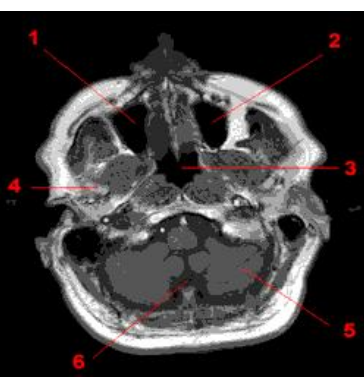
- Tumeur présente plusieurs composantes.
- Parenchyme cérébral (Substance grise et blanche).
- Bonne détection des artères, veines, os et cuir chevelu.

Fig. 4.35. b :

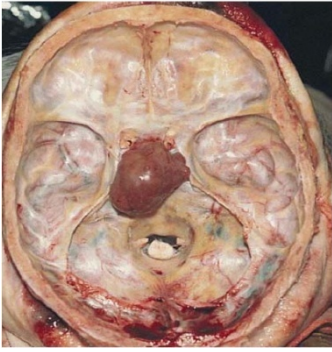


- L'image est artefactée.
- L'image segmentée en 5 classes et meilleure.

		
<p>Figure. 4.29 : Coupe axiale de l'abdominal.</p>	<p>Figure. 4.29. a Nbr_classe=8. $J(S_1, S_2) = 1,687$</p>	<p>Figure.4.29.b Nbr_classe= 6. $J(S_1, S_2) = 1,483$</p>
<p align="center">Figure. 4.29 Segmentation d'une Coupe axiale de l'abdominal</p>		
		
<p>Fig. 4.30. a Coupe sagittale couleur du cerveau</p>	<p>Fig. 4.30.b Nbr_classe=5 ; $J(S_1, S_2) = 1,364$</p>	<p>Fig. 4.30.c Nbr_classe= 6 ; $J(S_1, S_2) = 1,483$</p>
<p align="center">Figure. 4.30 Segmentation d'une Coupe sagittale couleur du cerveau</p>		
		
<p>Fig. 4.31. a Coupe axiale du Pelvis</p>	<p>Fig. 4.31.b Nbr_classe= 5 ; $J(S_1, S_2) = 1,367$</p>	<p>Fig. 4.31.c Nbr_classe= 6 ; $J(S_1, S_2) = 1,483$</p>
<p align="center">Figure. 4.31 Segmentation d'une Coupe axiale du Pelvis.</p>		



		
<p>Fig. 4.32 : Coupe sagittale d'IRM en T1 :((méningiome)</p>	<p>Fig. 4.32.a Nbr_classe=5; $J(S_1, S_2) = 1,367$</p>	<p>Fig. 4.32.b Nbr_classe=6; $J(S_1, S_2) = 1,483$</p>
<p>Figure 4.32 Segmentation d'une coupe sagittale d'IRM en T1 :((méningiome)</p>		
		
<p>Fig. 4.33 : Coupe sagittale d'IRM en T2.</p>	<p>Fig. 4.33.a Nbr_classe=5; $J(S_1, S_2) = 1,367$</p>	<p>Fig. 4.33.b Nbr_classe=6; $J(S_1, S_2) = 1,483$</p>
<p>Figure 4.33 Segmentation d'une coupe sagittale d'IRM en T2 (système-ventriculaire).</p>		
		
<p>Fig. 4.34 : Coupe axiale d'IRM au niveau de la base du crane.</p>	<p>Fig. 4.34. a Nbr_classe=6; $J(S_1, S_2) = 1,483$</p>	<p>Fig. 4.34. b Nbr_classe=7; $J(S_1, S_2) = 1,358$</p>
<p>Figure 4.34 Segmentation d'une coupe axiale d'IRM au niveau de la base du crane</p>		



		
<p>Fig. 4.35 : Coupe axiale anatomique de la base du crane sur cadavre Selle turcique</p>	<p>Fig. 4.35.a: Nbr_clust= 5, $J(S_1, S_2) = 1,323$</p>	<p>Fig. 4.35.b: Nbr_clust= 6, $J(S_1, S_2) = 1,424$</p>
<p>Figure 4.35 Segmentation d'une coupe axiale anatomique de la base du crâne sur cadavre (Selle turcique)</p>		

Discussion:

En appliquant la méthode développée sur différentes images médicales avec les meilleurs paramètres détectés et fixés auparavant, les résultats obtenus sont encourageants et très satisfaisants.

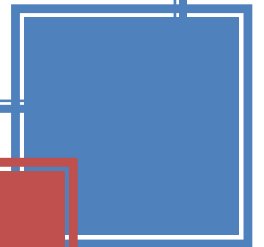
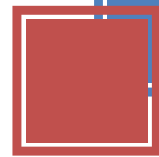
6. Conclusion

Dans ce travail, nous avons défini une nouvelle approche pour la segmentation d'images. La méthode proposée présente deux avantages fondamentaux: la flexibilité, dans le sens où elle permet à l'aide d'opérateurs fournis par l'algorithme génétique, la génération des solutions plus adéquates pour la segmentation d'une image donnée, et la généralisation d'après l'utilisation du modèle d'optimisation ACE. De plus, la méthode détecte les bons clusters, ceci signifie que l'extraction des régions de l'image est bien effectuée.

L'objectif de la segmentation de l'image du cerveau est d'identifier correctement les tissus principaux de la structure. En se basant sur l'analyse visuelle et l'index défini par la formule d'évaluations de la performance de Jaccard, nos objectifs principaux sont atteints par l'obtention de bons résultats de la segmentation de différentes coupes et types des images médicales ainsi que la fixation de meilleurs paramètres.

Nos perspectives visent à l'augmentation du nombre de la population et la modification des opérateurs de l'algorithme génétique d'apprentissage.

CONCLUSION GÉNÉRALE



Dans ce travail, le problème de la segmentation d'images médicales a été abordé. La méthode proposée est une nouvelle méthode hybride. Le but de cette recherche est de construire une nouvelle méthode de segmentation d'images, tirant profit d'un côté tirant profit d'un côté, d'un modèle de clustering flou basé sur l'estimation simultanée des classes (ACE), et d'un autre côté, de trois concepts d'apprentissage symbolique : systèmes d'inférence flous, les réseaux neuronaux, et les algorithmes génétiques.

Conceptuellement, notre méthode s'avère performante, dans le sens où elle bénéficie, des caractéristiques intrinsèques et avantageuses de chacun des outils cités préalablement. Ces caractéristiques peuvent se résumer par la capacité des algorithmes génétiques d'atteindre toujours une solution optimale, le parallélisme de traitement qu'offrent les réseaux neuronaux et la capacité de la logique floue de traiter l'imprécision et l'imperfection de données. En outre, la méthode conçue tire aussi ses avantages de l'adaptation de l'ACE ; qui lui a donné la particularité d'être générale et flexible dans la mesure où l'utilisateur n'est pas limité par l'utilisation d'un modèle particulier qui, peut être, ne pas avec ses besoins.

Afin de bien fixer les meilleurs paramètres de la méthode proposée, une évaluation automatique est effectuée sur les résultats obtenus en faisant varier les paramètres : nombre de classe et fonction d'adaptation.

L'objectif de la segmentation de l'image du cerveau est d'identifier correctement les tissus principaux de la structure. En se basant sur l'analyse visuelle et automatique donnée par la formule d'évaluation de la performance de Jaccard, nos objectifs principaux sont atteints. Les expériences réalisées sur des différents types d'images cérébrales montrent que la méthode hybride proposée, segmente rigoureusement les différents tissus de cerveau.

Nos perspectives visent à :

- L'amélioration de la fonction d'adaptation
- L'augmentation du nombre de population.
- Modification quelques opérateurs de l'algorithme d'apprentissage génétique

Dans ce travail, le problème de la segmentation d'images médicales a été abordé. La méthode proposée est une nouvelle méthode hybride. Le but de cette recherche est de construire une nouvelle méthode de segmentation d'images, tirant profit d'un côté tirant profit d'un côté, d'un modèle de clustering flou basé sur l'estimation simultanée des classes (ACE), et d'un autre côté, de trois concepts d'apprentissage symbolique : systèmes d'inférence flous, les réseaux neuronaux, et les algorithmes génétiques.

Conceptuellement, notre méthode s'avère performante, dans le sens où elle bénéficie, des caractéristiques intrinsèques et avantageuses de chacun des outils cités préalablement. Ces caractéristiques peuvent se résumer par la capacité des algorithmes génétiques d'atteindre toujours une solution optimale, le parallélisme de traitement qu'offrent les réseaux neuronaux et la capacité de la logique floue de traiter l'imprécision et l'imperfection de données. En outre, la méthode conçue tire aussi ses avantages de l'adaptation de l'ACE ; qui lui a donné la particularité d'être générale et flexible dans la mesure où l'utilisateur n'est pas limité par l'utilisation d'un modèle particulier qui, peut-être, ne pas avec ses besoins.

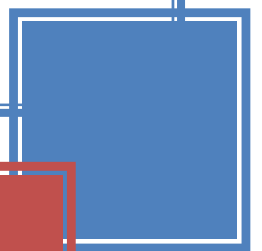
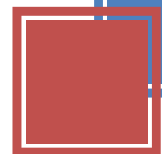
Afin de bien fixer les meilleurs paramètres de la méthode proposée, une évaluation automatique est effectuée sur les résultats obtenus en faisant varier les paramètres : nombre de classe et fonction d'adaptation.

L'objectif de la segmentation de l'image du cerveau est d'identifier correctement les tissus principaux de la structure. En se basant sur l'analyse visuelle et automatique donnée par la formule d'évaluation de la performance de Jaccard, nos objectifs principaux sont atteints. Les expériences réalisées sur des différents types d'images cérébrales montrent que la méthode hybride proposée, segmente rigoureusement les différents tissus de cerveau.

Nos perspectives visent à :

- L'amélioration de la fonction d'adaptation
- L'augmentation du nombre de population.
- Modification de quelques opérateurs de l'algorithme d'apprentissage génétique

RÉFÉRENCES BIBLIOGRAPHIQUES



Référence	Auteur /Titre / Année
[ABD 94]	Hervé.Abdi, " Les Réseaux de Neurones", Presses Universitaire de Grenoble, 1994.
[BAR 07]	V.Barra,J.F.Klauchen,A. "Lundervol, Evaluation of MR-brain- volumery method ", Research report 07_08, « avril 2007
[BEN 07]	Benoit Lavoie , "Logique classique et logique floue", 2007.
[BER 95]	Bernadette, Bouchon-Meunier, " La logique floue et ses applications ", Addison-Wesley, 1995.
[BER 09]	Berrichi Fatima Zohra 1, Benyettou Mohamed 1, "L'Interprétation des Images Médicales Cérébrales : Segmentation par L'approche Split et Merge", Juin 2009.
[BEZ 93]	J.C. Bezdek : "Fuzzy Models -What Are They, and Why ?". IEEE Transactions on Fuzzy Systems, Vol. 1, No. 1, February 1993
[BLO 03]	I. Bloch. " Traitement d'images", chap. 3, p. 95-151. Paris, Lavoisier - Hermès Science Publications, 2003, vol. Traitement de données complexes et commande en logique floue, collection Traite IC2 Information - Commande - Communication, série Informatique et Syst_emes d'Information
[Bou 01]	Abdel Hakim. Bourmel," Apprentissage Neuro-Flou ", Thèse de Magister, Département de'Informatique Université Mentouri Constantine. Septembre 2001
[CHA 04]	S.Chabrier, B.Emile, H.Laurent, C.Rosenberger, and P.Marché, " Unupervised evaluation of image segmentation-application to multi-spectral images", In IEEE conf.Proc.of International Conference on Pattern Recognition ICPR, VOL.1, PP, 576-579, Cambridge, Royaume-Uni,Aout 2004.
[CHE 01]	H.D. Cheng, X.H. Jiang, Y. Sun et Jingli Wang," Color image segmentation: advances and prespects", Pattern Recognition, 34:2259-2281, 2001.
[CHE 06]	F. Chevie F. Guély ; "la logiqque floue "; 2006.
[CHE 08]	CHERFA Yazid, "Segmentation d'IRM Cérébrale par Approche Coopérative Application à la caractérisation d'un AVC ", Mémoire pour l'obtention du grade de docteur d'Etat spécialité : traitement d'images, Ecole Nationale Polytechnique, décembre 2008.
[CLA 98]	M. C. Clark, L. O. Hall, D. B. Goldgof, R. Velthuizen, R. Murtagh, M. S. Silbiger. " Automatic tumor segmentation using knowledge-based techniques. IEEE Transactions on Medical Imaging", vol. 17, n_2 p. 187{201, 1998
[COL 04]	O. Colliot, O. Camara, I. Bloch. {Combinaison de relations spatiales et de modèles d_éformables pour la segmentation de structures cérébrales internes en IRM. 14_eme Congres Francophone AFRIF/AFIA de Reconnaissance des Formes et Intelligence Artificielle (RFIA), p. 563{572, 2004.
[COL 98]	D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes, A. C. Evans. "Design and construction of a realistic digital brain phantom. IEEE Transactions on Medical Imaging", vol. 17, n_3 p. 463{468, 1998
[CYB 05]	Cybele Ciofola , "Segmentation de formes guidée par des modèles en neuro-magerie", thèse présentée pour obtenir le grade de docteur de l'Université de Rennes 1 Mention Informatique , equipe d'accueil : VisAGeS – IRISA, ecole doctorale : Matisse,composante universitaire : IFSIC, le12 décembre 2005.
[DIN 07]	Dinesh K. Sharma, Loveleen Gaur and Daniel Okunbor, "Image Compression and Feature Extraction with Neural Network", <i>Proceedings of the Academy of</i>

- Information and Management Sciences*, Vol.11, No.1, 2007, pp. 33-38.
- [DOM 09] Dominique.Bereziat, "Segmentation des images médicales", Université Pierre et Marie Curie, Laboratoire d'Informatique de Paris 6, 8 octobre 2009.
- [ELA 87] Rich.Elaine, "Intelligence Artificielle", Masson, 1987.
- [ELK 11] Sabour Elkosantini, "Introduction à la logique floue, les concepts fondamentaux et application", 2011.
- [FAB 10] Fabien Moutarde, «Introduction aux réseaux de neurones», Centre de Robotique (CAOR) Mines ParisTech (ENSMP), Janvier 2010.
- [FER 98] S.Fernando, "Un Système Hybride Neuro-Symbolique pour l'Apprentissage Automatique", Thèse de Doctorat, l'Institut National Polytechnique de Grenoble-Labo.Leibnitz-IMAG, soutenue le : 03/02/1998 .
- [GOL 94] David E.Goldberg, «Algorithmes Génétiques», Addison-Wesley, France, S.A, Juin 1994.
- [GRI 02] V.H. Grisales, A. Gauthier, C.V. Isaza et G.A. Villamarín, "Identificación y Modelado Difuso, TS de un Bioreactor Anaerobio", X Congrès Latino-américain d'Automatique, Mexique, 2002 (version Française).
- [HAC 01] Abdel Hakim. Bourmel, "Apprentissage Neuro-Flou", Thèse de Magister, Département de l'Informatique Université Mentouri Constantine. Septembre 2001.
- [HAL 92] L. O. Hall, A. M. Bensaid, L. P. Clarke, R. P. Velthuizen, M. S. Silbiger, J. C. Bezdek. { A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain. *IEEE Transactions on Neural Networks*, vol. 3, n_5 p. 672-682, 1992.
- [HAN 04] X. Han, D. L. Pham, D. Tosun, M. E. Rettmann, C. Xu, J. L. Prince. "CRUISE: Cortical reconstruction using implicit surface evolution". *Neuroimage*, vol. 23, n_3 p. 997-1012, 2004
- [HAT 91] Jean Paul.Haton et co, "Le raisonnement en intelligence artificielle", Inter Édition, 1991.
- [HEN 95] R. Henkel : "Segmentation in scale space". *Computer Analysis of Images and Patterns*, pages 41-48, 1995.
- [HUL 03] R. Hult : "Grey-level morphology combined with an artificial neural networks approach for multimodal segmentation of the hippocampus". Dans *International Conference on Image Analysis and Processing*, pages 277-282, 2003
- [JAN 96] Jander Moreira and Luciano Da Fontoura Costa, "Neural-based color image segmentation and classification using self - organizing maps", *Anais do IX SIBGRAPI*, 1996, pp.47-54
- [JCF 10] J.C. Fu, C.C. Chen, J.W. Chai, S.T.C. Wong, I.C. Li, "Image segmentation by EM-based adaptive pulse coupled neural networks in brain magnetic resonance imaging", *Computerized Medical Imaging and Graphics*, 2010.
- [JEA 05] Jean-Marc Alliot, Nicolas Durand, "Algorithmes génétiques", 14 Mars, 2005.
- [JEA 91] Jean-Mark.Alliot, « Apprentissage Symbolique Automatique », Cépaduès, 1991.
- [JER 08] Jérémy Lecoœur, Christian Barillot "Segmentation d'images cérébrales : État de l'art"; institut national de recherche en informatique et en automatique, inria-00175188, version 3 - 18 feb 2008 ;

- [KHA 05] Kouadio M.R, Khanfouf .H, "une approche adaptative pour la segmentation d'images:implementation sur la plate-forme multi-agents", Net Logo, 2005
- [KWA 06] R Kwan, A Evans, G Pike, "An Extensible MRI Simulator for Post Processing Evaluation", SPIE, Visualization in Biomedical Computing, 1131:135-140, 1996.
- [KWA 96] R .Kwan, A. Evans, G Pike, "An Extensible MRI Simulator for Post Processing Evaluation", SPIE, Visualization in Biomedical Computing, 1131:135-140, 1996.
- [LAC 98] Jérôme Lacaille : "Présentation de la Logique Floue", 1998
- [Lam 00]. Lambalgen : "Fuzzy logic", ILLC University of Amesterdam, 2000
- [LEI 01] T. Lei, J. K. Udupa, P. K. Saha, D. Odhner. "Artery-vein separation via MRA - An image processing approach". IEEE Transactions on Medical Imaging, vol. 20, n_8 p. 689{703, 2001.
- [LIL 05] Lilia Lazli et Mohamed Tayeb Laskri , "Nouvelle méthode de fusion de données pour l'apprentissage des systèmes hybrides MMC/RNA, Application pour la reconnaissance automatique de la parole", Novembre 2005, Numéro spécial CARI'04, Revue ARIMA.
- [MAE 08] J. Maeda, A. Kawano, S. Yamauchi, Y. Suzuki A. R. S. Marcal and T. Mendonc, "Perceptual Image Segmentation Using Fuzzy - Based Hierarchical Algorithm and Its Application to Dermoscopy Images", IEEE Conference on Soft Computing in Industrial Applications (SMCia/08),June 25-27, 2008, Muroran, JAPAN, pp.66-71.
- [MAN 95] K.F.Man,K.S.Tang, "Genetic Algorithm for Control and Signal Processing", Departement of Electrnic Engeneering,City University of Hong kong,0-7803-3932-0. Master 2005-20006, "INTRODUCTION AUX RESEAUX DE NEURONES", cours_RN_2006 .pdf,
- [MAS 06] Timothy.Masters," Advanced Algorithms for Neural Network", Wiley, 1995.
- [MAT 01] Matlab 7.1, " Fuzzy Logic Toolbox", 2011
- [MAU 07] Mausumi Acharyya and Malay K. Kundu, "Image Segmentation Using Wavelet Packet Frames and Neurofuzzy Tools", International Journal of Computational Cognition, Vol.5, No.4, December 2007, pp.27-43.
- [MEZ 11] N. Mezhoud, F.Hachouf "A New Hybrid Method for Medical Image Segmentation", Journal of Theoretical and Applied Information TECHNOLOGY (JATIT), Paper-id: 14080, Avril 2011.
- [MIC 93]. R.S.Michalski, J.G.Carbonnel, T.M.Mitchell et Y.Kodratoff, " Apprentissage symbolique", Tome1, Cépaduès, 1993.
- [MOH 05] Mohamed Ali BEN HASSINE : " contribution a l'implémentation d'une base de données floue sous un système de gestion de base de données relationnel ", 2005
- [MOO 02] N.Moon, E.Bullitt, K.Van leemput, G.Gerig, "Model-based brain and tumor segmentation", Proc. 16tn International Conference on Pattern Recognition, Vol.1, PP 528-531, 2002
- [NAD 04] LASSOUAOUI Nadia, HAMAMI Latifa, NOUALI Nadia, " LES ALGORITHMES GENETIQUES, APPLICATION A LA SEGMENTATION DES IMAGES " RIST Vol.14 N°02 Année 2004, p40 à 49.
- [NAK 98] Nakache, "Data Warhouse et Data Mining", Document issu du CNAM de l'Ille, France 15 Juin 1998.
- [PHA 02] D. L. Pham, X. Han, M. E. Rettmann, C. Xu, D. Tosun, S. M. Resnick, J. L. Prince. "New approaches for measuring changes in the cortical surface using an automatic reconstruction algorithm". The International Society of Optical Engineering. SPIE Medical Imaging: Image Processing, vol. 4684, p. 191-200, 2002.
- [PHA 99] D. L. Pham, J. L Prince. "Adaptive fuzzy segmentation of magnetic resonance imaging". IEEE Transactions on Medical Imaging, vol. 18, n°9 p. 737-752, 1999.

- [RAC 11] Rachid Ladjaj, "Les Réseaux de Neurones", 2011
- [RUA 02] S. Ruan, B. Moretti, J. Fadili, D. Bloyet. "Fuzzy Markovian segmentation in application of Magnetic Resonance Images". Computer Vision and Image Understanding, vol. 85, n_1 p. 54-69, 2002.
- [RUN 99] Thomas A.Runkler, James C.Bezdek, "Alternating Cluster Estimation : A new Tool for Clustering and Function Approximation", IEEE Transaction on Fuzzy Systems, vol 7, N° 4, August 1999.
- [SAT 10] S.Sathish Kumar¹, M.Moorthi², M.Madhu³, Dr.R.Amutha, "AN IMPROVED METHOD OF SEGMENTATION USING FUZZY-NEURO LOGIC", Second International Conference on Computer Research and Development, 978-0-7695-4043-6/10 \$26.00 © 2010 IEEE DOI 10.1109/ICCRD.2010.155, pp -671-675.
- [SEN 09] N. Senthilkumaran¹ and R. Rajesh², "Edge Detection Techniques for Image Segmentation – A Survey of Soft Computing Approaches", International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009, pp 25-254
- [SLI 08] SLIME Samir, "Environnement de segmentation d'image à base d'une approche biomimétique », 2008.
- [SOU 04] Souquet Amédée, Radet Francois-Gérar, "ALGORITHMES GENETIQUES", TE de fin d'année Tutorat de Mr Philippe Audebaud, 21/06/2004.
- [STR 05] Olivier Strauss, Serge Guillaume, "Sous-ensembles flous, logique floue et systèmes d'inférence floue", 2005
- [SUN 05] Wei Sun and Yaonan Wang, "Segmentation Method of MRI Using Fuzzy Gaussian Basis Neural Network Neural Information Processing", Letters and Reviews, Vol.8, No.2, August 2005, pp.19-24
- [TAK 85] T. Takagi, M. Sugeno, "Fuzzy Identification of Systems and its Application to Modeling and Control", IEEE Transactions on Systems, Man and Cybernetics, 15, 1985.
- [TAL 04] M. Talibi-Alaoui, R. Touahni et A. Sbihi ; "Classification des Images Couleurs par association des Transformations Morphologiques aux Cartes de Kohonen" ; 2004
- [TOU 92] Claude Touzet, "Les réseaux de neurones artificiels, introduction au connexionismes, cours, exercices et travaux pratiques", Juillet 1992.
- [UDU 02] J. K. Udupa, P. K. Saha, R. A. Lotufo. "Relative fuzzy connectedness and object definition theory, algorithms and applications in image segmentation". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, n_11 p. 1485-1500, 2002.
- [UDU 96] J. K. Udupa, S. Samarasekera. "Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation". Graphical Models and Image Processing, vol. 58, n_3 p. 246-261, 1996
- [UDU 97] J. K. Udupa, L. Wei, S. Samarasekera, Y. Miki, M. A. van Buchem, R. I. Grossman. "Multiple sclerosis lesion quantification using fuzzy-connectedness principles". IEEE Transactions on Medical Imaging, vol. 16, n_5 p. 598-609, 1997.
- [WHI 93] D.Whitley, "Genetic Algorithm Tutorial", Technical report CS-93-103, Departement of Computer Science, Colorado state University, 10 Mars 1993.
- [XUE 01] J.-H. Xue, S. Ruan, B. Moretti, M. Revenu, D. Bloyet. "Knowledge-based segmentation and labeling of brain structures from MRI images". Pattern Recognition Letters, vol. 22, n_3-4 p. 395-495, 2001.
- [XUE 01] J.-H. Xue, S. Ruan, B. Moretti, M. Revenu, D. Bloyet. "Knowledge-based segmentation and labeling of brain structures from MRI images". Pattern Recognition Letters, vol. 22, n_3-4 p.395-495, 2001.
- [YAO 04]. J. Yao, M. Miller, M Franaszek, R. M. Summers. "Colonic polyp segmentation in CT



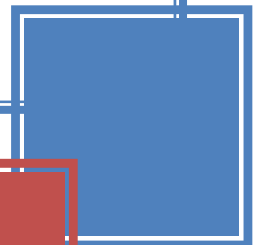
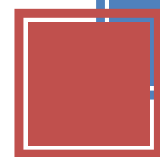
- colography based on fuzzy clustering and deformable models". IEEE Transactions on Medical Imaging, vol. 23, n_11 p. 1344{1352, 2004.
- [ZOU 08] Zouaoui Hakima, "Clustering par fusion floue de données appliqué à la segmentation d'images IRM", mémoire de magister, Université M'hamed BOUGARA de BOUMERDES, 2008.

Références Web

- [1] Scanner Rayon X et compagnie, élèves de 1^{ère} S au Lycée Saint Pierre de Cusset.
<http://WWW.pmsi.fr/Skanda2.htm>.
- [2] <http://recherche.enac.fr/opti/papers/thesis/HABIT/main002.html>
- [4] <http://www.pmsi.fr/gainit.htm>
- [5] http://fr.wikipedia.org/wiki/Logique_floue, 2011.
- [6] http://labo.algo.free.fr/defi_des_250_villes.html



LISTE DES CONTRIBUTIONS



Communications:

- « *Un système à bases de règles floues pour la détection du réseau routier* », H.Fellla, N.Mezhoud, Fourth International Symposium on Programming and Systems, 5-6-7 Mai 2003, Alger, ALGERIE.
- « *An improved fuzzy rule-based segmentation system* », Seventh International Symposium on Signal processing and its application, 1-4 Juillet 2003, Paris, France.
- « *Segmentation d'Images Couleurs par une Approche Neuro-floue* », N.Mezhoud, H.hachouf, MC.Batouche, pp 170-176, ISBN N° 9973 TAIMA'2003 « Ateliers Traitement & Analyse D'information Methods Et Applications » ;Taima03 hammamat, Tunisie, 29-30-1-2-3 septembre 2003.
- «*Segmentation Neuro-floue Application aux Images à Résonance Magnétique* », N.Mezhoud, H.hachouf, Conférence Internationale sur les systèmes de télécommunications, d'Electronique medicale et d'Automatique CISTEMA'2003 le 27-28-29 Septembre 2003.
- “*Color Image Segmentation Using A Fuzzy Clustering Method*”. H.Fellla, N.Mezhoud, IEEE International Conference on Industrial Technology,IEEE, ICIT'04, December, 08-10-,2004, Tunisia
- “*Satellite Color images Segmentation Using a fuzzy Clustering Method*”, N.Mezhoud, H.Fellla, International Symposium On Electromagnetism Satellites and Cryptography, ISEC05, June,19-20-21,2005, Jijel-Algeria.
- « *A Clustering Approach for Color Image Segmentation* », H.Fellla, N.Mezhoud, Advanced Concepts for Intelligent Vision Systems, ACIVS 2005, Antwerp, Belgium , LNCS 3708, pp. 515 – 522, 2005.© Springer-Verlag Berlin Heidelberg 2005
- « *Fusion d'Images Médicales par la transformée en Ondelettes* », N.Mezhoud, H.Fellla Ecole Internationale d'Eté sur le Traitement de Signal et ses Applications, I3SPA'06, le 08-09-10-11-12 juillet 2006.
- « *Fusion d'Images du cerveau humain, Application d'une méthode hybride floue* », N.Mezhoud, H.Fellla, 2ièmes Journées Internationales sur l'Informatique Graphique JIG 2006 (6-7 novembre 2006).
- «*Fusion Floue d'Images Multimodales du cerveau humain*», N.Mezhoud, H.Fellla, E-Medisys conference, "first international conference of medical systems" 24-27 octobre 2007.

- « *An Image Segmentation Algorithm Based on Fuzzy Clustering and Genetic Algorithms with a New Distance*», H.Fellla, N.Mezhoud The 6th Jordanian International Electrical & Electronics Engineering Conference , JIEEEEC 2007, 15-17 Nov, Amman, Jordan
- « *Fusion d'Images Multimodales du cerveau humain par une méthode hybride basée sur la logique floue et le calcul des modules maximums de la transformation des ondelettes* » N.Mezhoud, H.Fellla, mcseai08, Avril 28 - 30, 2008 Oran, Algérie.

Publication:

“*A New Hybrid Method for Medical Image Segmentation*”, N.Mezhoud, H.hachouf Journal of Theoretical and Applied Information TECHNOLOGY (JATIT), Paper-id: 14080, Avril 2011.

