

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



**Université de Batna
Faculté de Technologie
Département d'Electronique**

**THESE
DOCTORAT EN SCIENCES**

Thème

Classification floue des images

Présentée et soutenu publiquement le 14/12/2013

Par

Ouarda ASSAS

Devant le jury composé de :

N. BENOUIT	Professeur	Univ Batna	Président
K. BENMAHAMMED	Professeur	Univ Sétif	Rapporteur
M.N. HARMAS	Professeur	Univ Sétif	Examineur
A. MOUSSAOUI	Professeur	Univ Sétif	Examineur
M. BENYOUCEF	Professeur	Univ Batna	Examineur

TABLE DES MATIERES

Liste des figures	V
Liste des tableaux	VII
Introduction générale	1
Chapitre I Généralités sur le traitement d'images	
I.1. Introduction.....	4
I.2. Définition de l'image.....	4
I.3. Définition de l'image numérique.....	4
I.4. Types d'images.....	5
I.4.1. Image matricielle (bitmap)	5
I.4.2. Image vectorielle	5
I.5. Les images bitmap.....	5
I.5.1. Concepts et définitions de base.....	5
I.6. Le codage des couleurs	8
I.7. Formats de fichiers d'images	9
I.8. Quelques traitements d'images	10
I.8.1. Binarisation	10
I.8.2. Squelettisation	11
I.8.3. Segmentation	11
I.8.4. Convolution	12
I.8.5. Filtrage	12
I.8.6. Morphologie mathématique	15
I.8.7. La détection de contours	15
I.9. Conclusion	16
Chapitre II Classification d'images	
II.1 Introduction	17
II.3 Méthodes de Classification.....	17
II.4 Méthodes supervisées	18
II.4.1 Algorithmes de Bayes	18

II.4.2 Algorithmes K plus proche voisins (K-PPV)	19
II.4.3 Méthodes basées sur les réseaux de neurones artificiels	19
II.5 Méthodes non Supervisées	20
II.5.1 Algorithme des C-moyennes.....	20
II.5.2 Algorithmes C-moyennes floues ("Fuzzy C-Means" ou FCM) :.....	21
II.5.3 Algorithmes les C-moyennes possibilistes ("Possibilistic C-means" ou PCM).....	23
II.6 Segmentation d'images	24
II.7 Objectifs de la segmentation	25
II.8 Approches de segmentation	25
II.9 Approches basées Contour	25
II.10 Approches basées region	25
II.10.1 Méthode par division (Top-Down).....	26
II.10.2 Méthode par fusion (Bottom-Up).....	26
II.10.3 Méthode par division/fusion (Split and Merge).....	26
II.11 Approches basées classification	26
II.11.1 Algorithme K-means.....	26
II.11.2 Algorithme Estimation-Maximisation (EM).....	27
II.11.3 Classification par Seuillage.....	28
II.12 Exemples d'applications	29
II.13 Conclusion.....	29

Chapitre III Classification d'empreintes digitales

III.1. Introduction	30
III.2. Historique	30
III.3. L'empreinte digitale	31
III.4. Caractéristiques des empreintes digitales	31
III.4.1. Les points singuliers globaux	31
III.4.2. Les points singuliers locaux (minuties)	32
III.5. Classes des empreintes digitales.....	33
III.6. Intérêts de l'empreinte digitale.....	34
III.7. Description d'un système de classification des empreintes digitales.....	34
III.7.1. L'acquisition de l'empreinte	35
III.7.2. Le prétraitement	36
III.7.3. L'extraction des caractéristiques	36
III.7.3.1. Normalisation	36

III.7.3.2. Estimation de champ d'orientation d'image d'empreinte digitale	37
III.7.3.3. Segmentation.....	38
III.7.3.4. Détection des points singuliers globaux.....	39
III.7.4. Classification	41
III.8. Approches de classification des empreintes digitales.....	41
III.8.1. Approches statistiques.....	42
III.8.2. Approches structurales	42
III.8.3. Approches syntaxiques.....	43
III.8.4. Approches des réseaux de neurones.....	44
III.8.5. Approches basées sur les règles	44
III.9. Etat de l'art de la classification des empreintes digitales.....	45
III.10. Conclusion.....	48

Chapitre IV Les Réseaux de Neurones flous

IV.1. Introduction	49
IV.2. Notions fondamentales sur les réseaux de neurones	49
IV.2.1. Introduction	49
IV.2.2. Neurone biologique	49
IV.2.3. Neurone artificiel.....	50
IV.2.4. L'apprentissage.....	51
IV.2.5. La rétro-propagation du gradient de l'erreur [54].....	55
IV.2.6. Avantages et Inconvénients des réseaux de neurones	56
IV.3. Les Réseaux de Neurones Flous.....	58
IV.3.1. La logique floue.....	59
IV.3.2. Systèmes Neuro-Flous	62
IV.3.3. Les combinaisons des réseaux de neurones avec la logique floue	64
IV.3.4. Les types des réseaux de neurones flous utilisés pour la classification	64
IV.3.4.1. La structure de Pedrycz	64
IV.3.4.2. La structure fusionnée compacte	65
IV.3.4.3. Les systèmes neuro-flous de classifications NEFClass	66
IV.4. Conclusion.....	66

Chapitre V Présentation et analyse des résultats

V.1. Introduction	67
V.2. Présentation du système implémenté	68
V.2.1. La base de données des images utilisées	68
V.2.2. Extraction d'image d'orientation de l'empreinte digitale.....	70
V.2.3. Extraction des points singuliers globaux.....	70
V.2.4. Construction du vecteur Caractéristique	72
V.2.5. Classification.....	73
V.3. L'architecture du Réseaux de Neurones Artificiels Adopté	74
V.4. Les Architectures des Systèmes Neuro-Flous Adoptés	74
V.4.1. Le réseau fusionné compact	76
V.4.2. Le réseau NEFClass.....	77
V.4.3. Le réseau de Pedrycz.....	78
V.5. Tests et résultats.....	79
V.6. Discussion.....	82
V.7. Conclusion.....	84
Conclusion générale et perspectives.....	85
Références Bibliographiques	87
Liste des contributions	93

Liste des figures

Fig.I.1. <i>Image matricielle</i>	4
Fig.I.2. <i>La résolution d'une image</i>	5
Fig.I.3. <i>Exemple d'histogramme d'une image</i>	6
Fig.I.4. <i>La transformation ponctuelle d'une image numérique</i>	7
Fig.I.5. <i>Binarisation d'une image</i>	11
Fig.I.6. <i>Squelettisation d'une image</i>	12
Fig.I.7. <i>Principe du filtre médian</i>	15
Fig.III.1. <i>L'empreinte digitale</i>	31
Fig.III.2. <i>Les points singuliers globaux</i>	32
Fig.III.3. <i>Les différents types de minuties</i>	32
Fig.III.4. <i>Les cinq classes des empreintes digitales</i>	33
Fig.III.5. <i>Schémas générale d'un système de classification des empreintes digitales</i> ...	35
Fig.III.6. <i>L'orientation de la crête locale du pixel(x, y)</i>	37
Fig.III.7. <i>L'image d'orientation d'une empreinte digitale</i>	38
Fig.III.8. <i>Effet de la segmentation sur l'image</i>	39
Fig.III.9. <i>L'index de Poincaré calculé sur une courbe C</i>	39
Fig.III.10. <i>L'index de Poincaré calculé avec un voisinage de $N= 8$</i>	40
Fig.III.11. <i>Détection des points singuliers par l'algorithme de l'index de Poincaré</i>	41
Fig.III.12. <i>Le graphe relationnel d'une image d'empreinte digitale</i>	43
Fig.III.13. <i>Un schéma de l'approche chaîne de la construction dans Rao et Balck 1980</i> .	43
Fig.IV.1. <i>Neurone biologique</i>	50
Fig.IV.2. <i>Neurone artificiel</i>	50
Fig.IV.3. <i>fonctions d'activations les plus utilisées</i>	51
Fig.IV.4. <i>Schéma général de perceptron simple</i>	54
Fig.IV.5. <i>Exemple de réseau de type perceptron Multicouche</i>	54
Fig.IV.6. <i>Réseaux de neurones bouclés</i>	55
Fig.IV.7. <i>Variable linguistique</i>	60
Fig.IV.8. <i>Formes des fonctions d'appartenance usuelles</i>	61
Fig.IV.9. <i>Structure d'un système d'inférence floue</i>	62
Fig.IV.10. <i>principe du système neuro-flou</i>	63
Fig.IV.11. <i>Architecteur de réseau de Pedrycz</i>	65
Fig.IV.12. <i>Architecteur de réseau fusionnée compacte</i>	65

Fig.IV.13. <i>Architecteur d'un système NEFClass</i>	66
Fig.V.1. <i>Structure de système de classification des empreintes digitales</i>	67
Fig.V.2. <i>Echantillons de la base de données CASIA-Fingerprint Image Version 5.0</i>	69
Fig.V.3. <i>L'extraction de l'image d'orientation de l'empreinte digitale</i>	70
Fig.V.4. <i>Détection des points singuliers avant la segmentation</i>	71
Fig.V.5. <i>Détection des points singuliers après la segmentation</i>	72
Fig.V.6. <i>Structure de réseaux de neurones artificiels utilisés</i>	74
Fig.V.7. <i>Les fonctions d'appartenance des variables d'entrée du système</i>	75
Fig.V.8. <i>Réseau fusionné compact</i>	76
Fig.V.9. <i>Réseau NEFClass</i>	77
Fig.V.10. <i>Réseau de Pedrycz</i>	78
Fig.V.11. <i>Les six exemples de test des quatre réseaux</i>	80
Fig.V.12. <i>Taux de classification des différents réseaux de neurones</i>	82

Liste des tables

Tab.I.1. <i>Masque de convolution du filtre passe-bas</i>	13
Tab.I.2. <i>Masque de convolution du filtre passe-haut</i>	13
Tab.I.3. <i>Masque de convolution di filtre Gaussien</i>	14
Tab.I.4. <i>Les différents masques de convolution utilisée pour la détection du contour.....</i>	16
Tab.III.1. <i>Les classes d'empreintes et leurs nombre de points singuliers.....</i>	44
Tab III.2. <i>Classification des algorithmes de la classification des empreintes digitales.....</i>	48
Tab.V.1. <i>Le codage de variables d'entrées.....</i>	73
Tab.V.2. <i>Le codage binaire de variable de sortie.....</i>	73
Tab.V.3. <i>Le nombre de couche et le nombre de neurones de chaque système.....</i>	78
Tab.V.4. <i>Les sorties du réseau de neurones artificiels.....</i>	80
Tab.V.5. <i>Les sorties du réseau fusionné Compact.....</i>	80
Tab.V.6. <i>Les sorties du système NEFClass.....</i>	81
Tab.V.7. <i>Les sorties de réseau de Pedrycz.....</i>	81
Tab.V.8. <i>Les taux de classification des différents réseaux de neurones utilisés.....</i>	82

REMERCIEMENTS

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

Ce travail n'aurait pas pu aboutir à des résultats sans l'aide et les encouragements de plusieurs personnes que je remercie.

Mes vifs remerciements accompagnés de toute ma gratitude vont ensuite à mon encadreur BENMAHAMMED Khier, professeur à l'université de Sétif, pour ses conseils judicieux, sa grande disponibilité et pour m'avoir suivie et orientée.

J'exprime ma profonde gratitude à Monsieur BENOUIT Nabil, Professeur à l'université de Batna, qui m'a fait l'honneur de présider le jury de thèse.

J'adresse mes sincères remerciements à Monsieur BENYOUCEF Moussa , professeur à l'université de Batna, Monsieur HARMAS Mohamed Najib, professeur à l'université de Sétif, et Monsieur MOUSSAOUI Abdelouahab, professeur à l'université de Sétif , pour l'intérêt qu'ils ont bien voulu porter à ce travail en acceptant d'en être rapporteurs.

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.

Je tiens aussi à exprimer ma reconnaissance à toutes mes amies pour leur soutien inconditionnel.

Introduction générale

Le traitement d'images numériques est actuellement en plein progrès, il est d'utilité incontestable dans de nombreuses disciplines scientifiques et industrielles particulièrement à l'aide d'un formalisme. L'analyse d'images a pour but la description des objets et scènes d'images, et donc l'extraction automatique des informations qualitatives et la discrimination des divers composants de l'image. Ces techniques employées dépendent essentiellement de la nature des images à analyser et du but fixé par l'utilisateur.

La classification consiste à attribuer une forme représentée par un vecteur à une classe parmi d'autres, elle est une étape importante dans plusieurs autres tâches accomplies dans le traitement de signal. La classification est un problème central de la reconnaissance de formes. De nombreuses méthodes dites classiques ont été consacrées telles que : arbres de décision, approche Bayésienne, K plus proche voisin, réseaux de neurones artificiels, etc. Elles ont donné lieu au développement de plusieurs applications. Toutefois, certaines difficultés n'ont toujours pas reçues de solutions satisfaisantes.

Toutes ces méthodes sont utilisées pour faire la classification dure basée sur le principe : qu'une forme appartient à une classe en d'autres termes, une forme est un membre complet ou n'est pas d'une classe donnée d'informations. Cette logique de traitement ne représente pas la meilleure manière de traiter des données qui sont en général mélangées et imprécises en nature. Ainsi l'introduction de la logique floue a apporté la classification floue, cette dernière estime la contribution de chaque classe dans la forme. Elle suppose qu'une forme est une unité décomposable et par conséquent travaille sur un nouveau principe : une forme appartient à plusieurs classes pour fournir plus d'informations sur la forme contrairement à la classification dure qui est pauvre en extraction d'informations, les fonctions d'appartenance des ensembles ou classes floues sont calculées par la procédure du maximum de vraisemblance définie sur des signatures des classes floues.

L'identification automatique des personnes basée sur les empreintes digitales exige la comparaison de l'empreinte digitale d'entrée avec un grand nombre d'empreintes digitales stockées dans une base de données. Cette base de données peut être très grande

(par exemple, la base de données de FBI contient plus de 200 millions des empreintes digitales) dans de nombreuses applications judiciaires et civiles. Dans de tels cas, l'identification a généralement un temps de réponse trop grand. Le processus d'identification peut être accéléré en réduisant le nombre de comparaisons qui doivent être effectuées. Une stratégie intuitive pour atteindre cet objectif est de diviser la base de données d'empreintes digitales en un certain nombre de sous-ensembles. Une empreinte digitale à identifier est ensuite comparée uniquement aux empreintes digitales d'un seul sous-ensemble qui est la base de sa catégorie. Les premières études scientifiques sur la classification des empreintes digitales ont été faites par Francis Galton (1822-1916), qui a classé les empreintes digitales en trois grandes catégories : arc, boucle et spire. Ensuite Edward Henry (1850-1931) a affiné cette dernière par l'augmentation de nombre des classes en cinq classes : arc, arc tendu, boucle à gauche, boucle à droite, et spire, qui est actuellement utilisée. La classification des empreintes digitales se réfère au problème de l'affectation d'une empreinte digitale à une classe d'une manière cohérente et fiable.

La classification des empreintes digitales est un problème très difficile qui a attiré l'attention de plusieurs chercheurs dans les dernières années. Pour cette raison plusieurs approches de classification et différentes méthodes d'extraction des caractéristiques ont été proposées, mais aucune parmi ces approches n'est considérée comme complète. Parmi les méthodes proposées nous nous intéressons particulièrement à la classification par les réseaux de neurones flous.

Les réseaux de neurones flous sont nés de l'association des réseaux de neurones avec la logique floue, de manière à tirer profit des avantages de chacune de ces deux techniques. La principale propriété des réseaux de neurones flous est leur capacité à traiter dans un même outil des connaissances numériques et symboliques d'un système. Ils permettent donc d'exploiter les capacités d'apprentissage des réseaux de neurones d'une part et les capacités de raisonnement de la logique floue d'autre part.

L'objectif de ce travail est justement de construire une méthode de classification des empreintes digitales basée sur les réseaux de neurones flous, dont le but, d'affecter une empreinte digitale à sa classe d'une manière efficace en utilisant les caractéristiques globales de celles-ci. Pour ce faire, Nous utilisons deux outils de l'intelligence artificielle : les réseaux neuronaux, pour un besoin de parallélisme et de puissance de

calcul qui s'avèrent essentiels dans les traitements des empreintes digitales et la logique floue pour une meilleure représentation des informations . On pense que de cette manière on bénéficiera des apports de la théorie des ensembles flous et des réseaux neuronaux, pour rendre la méthode utilisée plus performante.

La thèse est organisée en cinq chapitres qui nous permettront de présenter les différents aspects de notre travail.

Dans **le premier chapitre** nous introduisons des notions générales sur le traitement d'images, en donnant la définition et les caractéristiques de l'image numérique et les filtres les plus utilisées pour l'amélioration de la qualité des images.

Le deuxième chapitre donne un aperçu général sur la classification d'images et les différentes approches utilisées pour sa réalisation.

Le troisième chapitre sera dédié à la présentation de l'empreinte digitale, ces caractéristiques, ces classes et les étapes par lesquelles passe un système de classification des empreintes digitales, ainsi que les approches de classification utilisées.

Le quatrième chapitre est constitué de deux parties :

La première partie introduit les notions fondamentales des réseaux de neurones artificielles, ensuite décrit les topologies, les types de réseaux existants et les modes d'apprentissage.

La deuxième partie contient la théorie de la logique floue, suivi par les plus importants de ses concepts de base, ensuite, on met l'accent sur les réseaux de neurones flous et leurs principaux types et modèles.

Et **le dernier chapitre** contient le fruit de notre travail, les résultats obtenus de la classification des empreintes digitales (modèles de classification) enrichie par évaluation et une analyse de ces derniers.

Résumé

ملخص:

تصنيف بصمة أصبع هو تعيين صنفها بين الاصناف المعترف بها و ذلك من خلال البحث في جميع بصمات الأصابع في قاعدة البيانات وهذا قد يستغرق وقتا طويلا. ومن أجل تقليل وقت البحث والتعقيد الحسابي يجري اختيار مجموعة جزئية من قاعدة البيانات مناسبة للبحث فيها. تصنيف البصمات كان وسيظل مشكلة صعبة في التعرف على نمط بسبب التغير الكبير بين الفئات و التغير الطفيف داخل الصنف الواحد. ويعرض هذا العمل نماذج لتصنيف بصمات الأصابع مبنية على أساس الشبكات العصبية الغامضة واستخدام النقاط المميزة (مركز و دلتا)، وكذلك وضعيتها النسبية. يتم تصنيف البصمات إلى خمس فئات وفقا لتصنيف هنري : ألقوس، القوس المشدودة، عروة، حلقة الي اليسار وحلقة الي اليمين. وقد تم تصميم ثلاثة نماذج من الشبكات العصبية الغامضة واستخدمت قاعدة البيانات CASIA-FingerprintV5 لصور البصمات لبناء واختبار هذه الشبكات. استخدام الشبكات العصبية الغامضة للتصنيف يزيد من مصداقية مخارج نظام التصنيف. استخدام الشبكات العصبية الاصطناعية المنطق الغامض، الشبكات العصبية الغامضة.

Résumé:

La classification des empreintes digitales consiste à attribuer une empreinte donnée à l'une des classes existantes déjà reconnues dans la littérature. Une recherche sur l'ensemble des empreintes digitales dans la base de données prend beaucoup de temps, donc le but est de réduire le temps de recherche et de la complexité des calculs en choisissant un sous-ensemble approprié de base de données pour la recherche. La classification des empreintes digitales est et restera un problème difficile dans la reconnaissance des formes, en raison de la variabilité inter-classe minimale et de la variabilité intra-classe maximale. Ce travail présente des modèles de la classification des empreintes digitales basée sur les réseaux de neurones flous et l'utilisation des points singuliers (centre et Delta) ainsi que leurs positions relatives comme vecteur caractéristique de celles-ci. Le vecteur de sortie est défini en fonction des valeurs d'appartenance aux cinq classes, arc, arc tendu, spire, boucle à gauche et boucle à droite. Trois modèles de réseaux neuronaux flous ont été développés et les images d'empreintes digitales de la base CASIA-FingerprintV5 ont été utilisées pour la construction et le test de ces réseaux. L'utilisation de l'approche neuro-floue pour la classification augmente la crédibilité de la sortie du système de classification.

Mots clés : Biométrie, Empreinte digitale, Classification, Extraction des caractéristiques, Points singuliers globaux, les réseaux de neurones artificiels, la logique floue, les réseaux de neurones flous.

ABSTRACT

Classification means to assign a given fingerprint to one of the existing classes already recognized in the literature. A search over all the fingerprints in the database takes a long time, so the goal is to reduce the search time and computational complexity by choosing an appropriate subset of database for search. Classifying a fingerprint images is and will remain a challenging problem in pattern recognition, due to the minimal interclass variability and maximal intra-class variability. This work presents models of fingerprint classification based on fuzzy neural networks and the use of the singular points (center and Delta) and their relative as characteristic vector. The output vector is defined in terms of membership values to the five classes, arch tented arch, whorl, left Loop and right Loop. Three models of fuzzy neural networks were implemented and fingerprint images from CASIA-FingerprintV5 database were used for training and testing these networks. The classifiers were tested on 200 images of the CASIA-FingerprintV5 database. The use of neuro-fuzzy approach for the classification increases the credibility of the output of the classification system.

Keywords: Biometry, Fingerprint, Classification Approaches, Feature extraction, Singular points, Fuzzy Neural Network.

I.1. Introduction

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

Le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dans le but d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes qu'on va utiliser dans différentes applications par exemple la reconnaissance, la classification, ...etc.

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

I.2. Définition de l'image

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

Elle peut être décrite sous la forme d'une fonction $I(x,y)$ de brillance analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et I est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation [1].

I.3. Définition de l'image numérique

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

La numérisation est le processus qui permet de passer de l'état d'image physique (image optique par exemple) qui est caractérisée par l'aspect continu du signal qu'elle représente (une infinité de valeur de l'intensité lumineuse par exemple), à l'état d'image numérique qui est caractérisée par l'aspect discret (l'intensité lumineuse ne peut prendre que des valeurs quantifiées en un nombre fini de points distincts). C'est cette forme numérique qui permet une exploitation ultérieure par des outils logiciels sur ordinateur [2].

Pour une mesure de simplicité, on utilise le terme image au lieu d'image numérique dans la suite du mémoire.

I.4. Types d'images

I.4.1. Image matricielle (bitmap)

Une image matricielle (voir Fig.I.1) est une image en mode point. Le système de codage le plus universel consiste en effet à décomposer la représentation graphique, l'image, en un certain nombre de points élémentaires caractérisés par leurs coordonnées spatiales et leur couleur.



Fig.I.1. Image matricielles.

I.4.2. Image vectorielle

Dans une image vectorielle les données sont représentées par des formes géométriques simples qui sont décrites d'un point de vue mathématique. Il s'agit de représenter les données de l'image par des formules géométriques qui vont pouvoir être décrites d'une façon mathématique.

Autrement dit, on stocke la succession d'opérations conduisant au tracé dans le cas d'une image vectorielle, alors qu'on mémorise une mosaïque de points élémentaires dans le cas d'image matricielle.

Ces images présentent deux avantages : elles occupent peu de place en mémoire et peuvent être redimensionnées sans perte d'information.

I.5. Les images bitmap

I.5.1. Concepts et définitions de base

I.5.1.1. Pixel

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

I.5.1.2. Le codage de pixel

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

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

I.5.1.3. La taille d'une image

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

I.5.1.4. La résolution d'une image

Dans le domaine de l'imagerie numérique, la résolution est une mesure de la finesse de l'affichage ou de la capture d'une image, exprimée en nombre de pixels par unité de surface, c'est-à-dire la « densité » en pixels.

La résolution d'une image numérique s'exprime en PPI (Pixels Per Inch) ou PPP (Pixels Par Pouce).

Plus la résolution d'une image est grande plus sa qualité est meilleure (voir Fig.I.2)

[3].

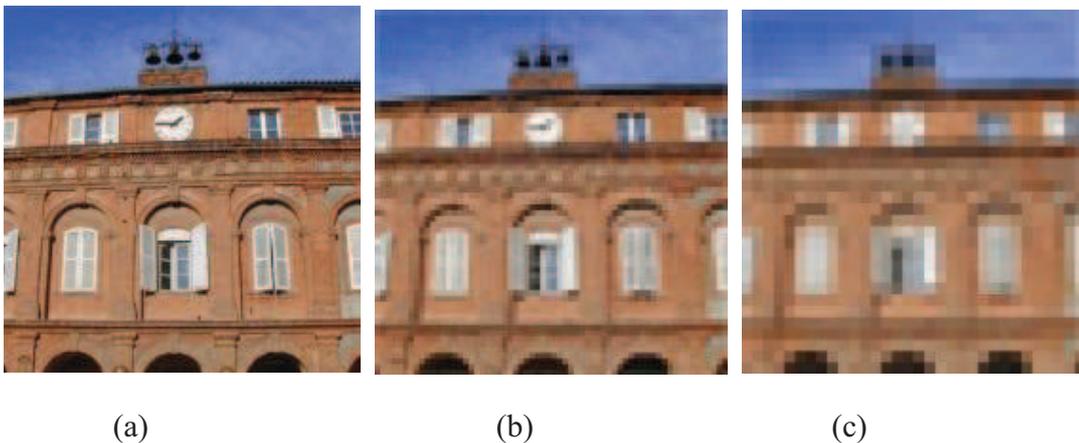


Fig.I.2. La résolution d'une image (a) image acquise à 256 dpi, (b) image acquise à 64 dpi, (c) image acquise à 32 dpi.

I.5.1.5. La luminance

Le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. La luminance est le degré de luminosité des pixels de l'image. Elle est définie aussi comme l'intensité d'une source étendue dans une direction donnée, divisée par l'aire apparente de cette source dans cette même direction [3].

I.5.1.6. Le contraste

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux régions d'une image.

Si L_1 et L_2 sont les degrés de luminosité respectivement de deux régions voisines A_1 et A_2 d'une image, le contraste C est défini par le rapport :

$$C = \frac{L_1 - L_2}{L_1 + L_2} \quad (I.1)$$

I.5.1.7. Le bruit

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

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

I.5.1.8. L'histogramme

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui associe à chaque valeur d'intensité le nombre de pixels de l'image ayant cette valeur.

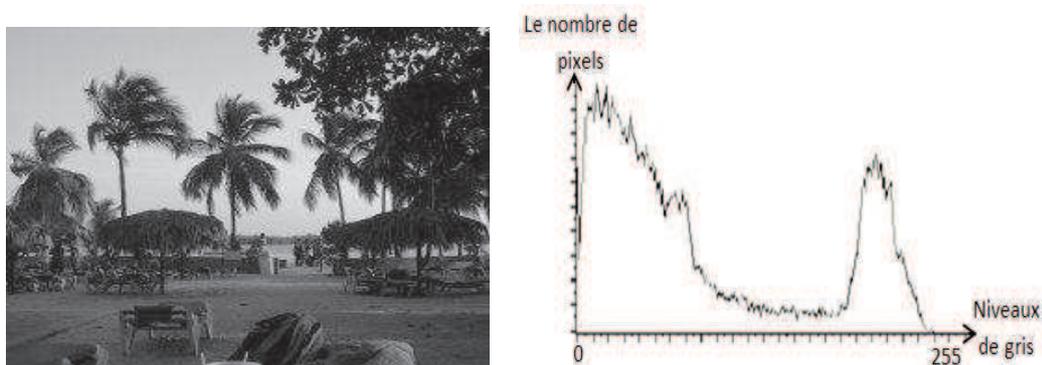


Fig.I.3. Exemple d'histogramme d'une image.

I.5.1.9. La palette des couleurs

On appelle la palette des couleurs, la liste des couleurs que peut contenir une image. La valeur de chaque pixel représente le rang d'une couleur dans cette liste.

Il est fréquent de voir des images qui n'utilisent jamais certaines couleurs, il devient dès lors intéressant de limiter la palette de couleur en ne sélectionnant que la ou les couleurs utilisées réellement par l'image (voir Fig.I.4).

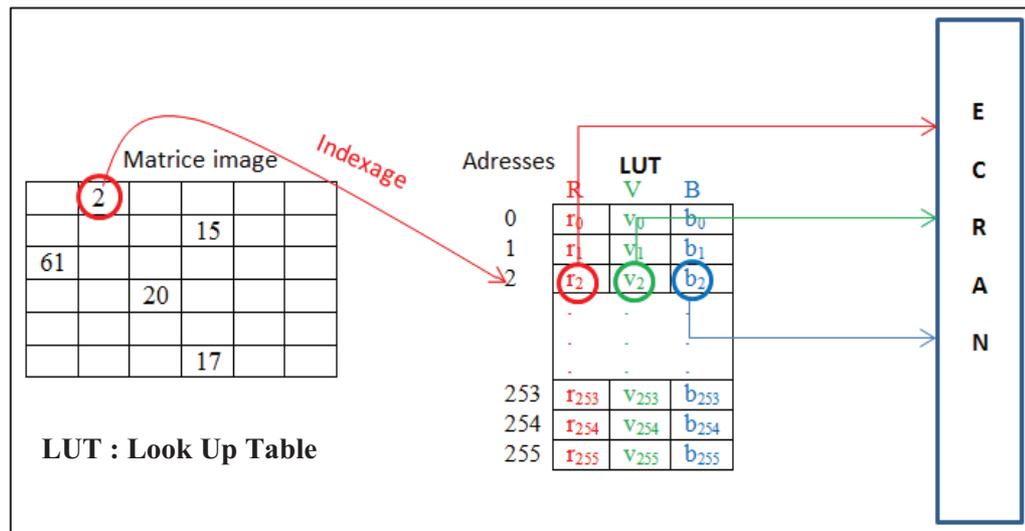


Fig.I.4. La transformation ponctuelle d'une image numérique.

I.6. Le codage des couleurs

1. Les images binaires (noir ou blanc)

Les images binaires sont des images de profondeur $K=1$ bit, donc un pixel peut prendre l'une des valeurs : noir ou blanc (0 ou 1).

C'est typiquement le type d'images que l'on utilise pour scanner du texte quand celui-ci est composé d'une seule couleur.

2. Les images en niveaux de gris

En général, les images en niveaux de gris sont des images de profondeur $k=8$ bits, donc chaque pixel peut prendre l'une des valeurs de l'intervalle $[0 \dots 255]$, où la valeur 0 représente la brillance minimale (le noir) et 255 la brillance maximale (le blanc). Ce type d'image est fréquemment utilisé pour reproduire des photos en noir et blanc ou du texte.

Dans plusieurs applications professionnelles de photographie et d'impression ainsi qu'en médecine et astronomie, 8 bits par pixel n'est pas suffisant, pour cela il existe d'autres types d'images en niveaux de gris de profondeur $K=12$, $K=14$ ou $K=16$ bits.

3. Les images couleurs

L'espace couleur est basé sur la synthèse additive des couleurs, c'est à dire que le mélange entre différentes couleurs (trois, quatre...) donne une couleur.

La plupart des images couleurs sont basées sur trois couleurs primaires : Rouge, Vert et Bleu (RVB) (RGB en anglais), et utilisent typiquement 8 bits pour chaque composante de couleur, donc chaque pixel nécessite $3 \times 8 = 24$ bits pour coder les trois composantes, et chaque composante de couleur peut prendre l'une des valeurs de l'intervalle $[0 \dots 255]$.

I.7. Formats de fichiers d'images

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

La plupart des formats sont composés d'un en-tête contenant des attributs (dimension de l'image, type de codage, LUT, etc.), suivi des données (l'image proprement dite). La structuration des attributs et des données diffère d'un format à un autre. Il existe plusieurs formats d'images, nous allons mentionner quelques-uns :

I.7.1. windows BitMaP (BMP)

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

I.7.2. Tagged Image File Format (TIFF)

Ce format est orienté vers les professionnels (imprimeurs, publicitaires...) car il a l'avantage d'être reconnu sur tous types de système d'exploitation : Windows, Mac, Linux, Unix, ... etc.

Il permet d'obtenir une image de très bonne qualité, mais sa taille reste volumineuse, même si elle est inférieure à celle du fichier BMP.

I.7.3. Joint Photographic Expert Group (JPEG)

C'est le format le plus courant, on le rencontre sur Internet. Il occupe peu d'espace disque. C'est le format développé par les photographes pour transmettre des images de qualité photographique professionnelle. Il gère des millions de couleurs mais il ne possède pas de palette de couleurs associée et donc les couleurs peuvent être différentes sur des machines et des systèmes différents.

I.7.4. Graphics Interchange Format (GIF)

Les fichiers au format GIF sont fortement compressés tout en gardant une qualité très correcte. Ils possèdent une palette de couleurs associée (limité à 256 couleurs) et occupent peu d'espace disque.

I.7.5. Portable Network Graphic (PNG)

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

Nous avons utilisés dans notre système des images de format BMP, Pour cela, nous allons décrire la structure d'un fichier BMP.

I.8. Quelques traitements d'images

Il existe une variété de traitements d'images, nous allons présenter quelques-uns :

I.8.1. Binarisation

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

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

$$g(x, y) = \begin{cases} 0 & \text{si } f(x, y) < S \\ 255 & \text{si } f(x, y) \geq S \end{cases}$$

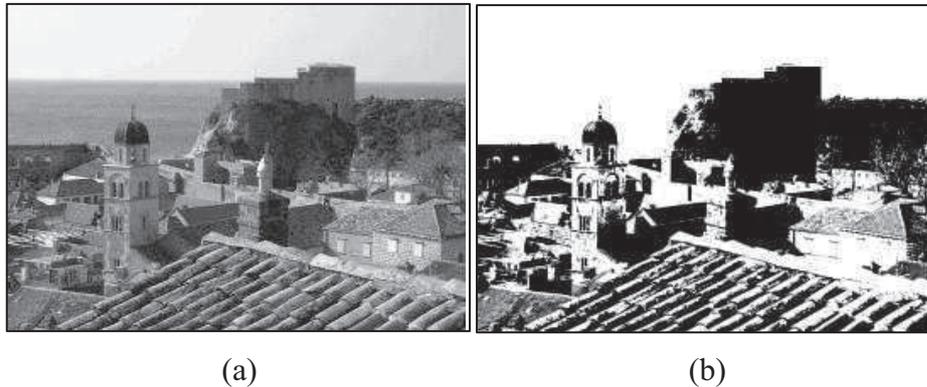


Fig.I.5. Binarisation d'une image (a) image originale, (b) image binarisée.

I.8.2. Squelettisation

La procédure de squelettisation s'effectue sur une image binaire, et a pour but de réduire l'épaisseur du tracé à un pixel seulement, tout en conservant la continuité de celui-ci (voir Fig.I.6) [7].

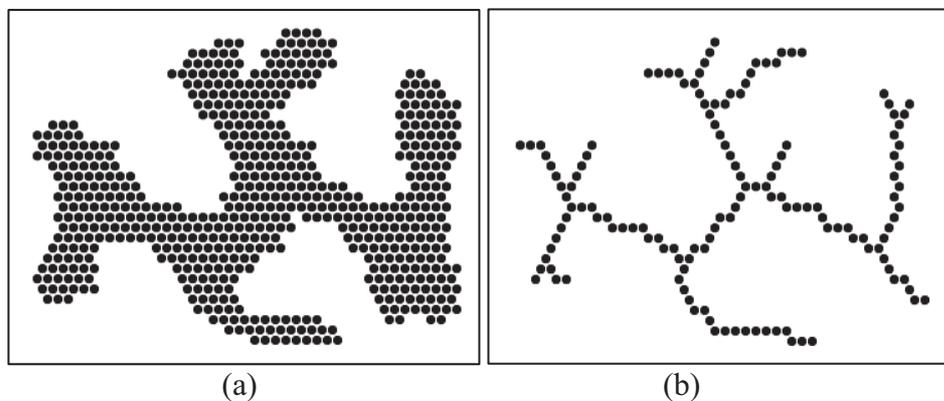


Fig.I.6. Squelettisation d'une image (a) image originale, (b) image squelettisée.

I.8.3. Segmentation

La segmentation d'images est une opération qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis, et peut-être accomplie selon plusieurs méthodes. Les pixels ainsi regroupés en régions constituent un pavage ou une partition de l'image. La segmentation est une étape primordiale en traitement d'images. [3]

I.8.4. Convolution

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

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

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

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

$$S(i, j) = \sum_{k=-(N-1)/2}^{(N-1)/2} \sum_{l=-(N-1)/2}^{(N-1)/2} E_{i+k, j+l} \cdot h_{k, l} \quad (I.2)$$

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

I.8.5. Filtrage

La notion de filtre est empruntée à la physique et aux techniques de traitement de signal. Si un signal (électrique, radio, image,...etc.) présente des composantes de fréquences très différentes, il peut être avantageux d'en éliminer certaines, en parle dans ce cas de filtrage.

Le filtrage des images est un traitement local utilisé principalement pour réaliser une analyse spatiale d'une image. Son objectif est d'accentuer les variances d'intensité de l'image, ou de détecter les contours et de réduire les bruits existants.

Il existe un grand nombre de filtres, on peut les classer en deux grandes catégories : les filtres linéaires et les filtres non linéaires.

I.8.5.1. Filtres linéaires

Un filtre linéaire transforme un ensemble de données d'entrée en un ensemble de données de sortie selon une opération mathématique appelée convolution. Il permet pour chaque pixel de la zone à laquelle il s'applique, de modifier sa valeur en fonction des valeurs des pixels avoisinants.

Il existe plusieurs filtres linéaires, nous allons présenter quelques-uns :

a. Les filtres passe-bas (lissage)

Consistent à atténuer les pixels de l'image ayant une fréquence haute (pixels foncés). Ce type de filtrage est généralement utilisé pour atténuer le bruit de l'image, c'est la raison pour laquelle on parle habituellement de lissage.

Les filtres moyens sont des types de filtres passe-bas dont le principe est de faire la moyenne des valeurs des pixels avoisinants. Le résultat de ce filtre est une image plus floue.

1	1	1
1	4	1
1	1	1

Tab. I.1. Masque de convolution du filtre passe-bas.

b. Les filtres passe-haut (accentuation)

À l'inverse des filtres passe-bas, les filtres passe-haut atténuent les pixels de basse fréquence de l'image et permettent notamment d'accentuer les détails et le contraste, c'est la raison pour laquelle le terme de "filtre d'accentuation" est parfois utilisé.

0	-1	0
-1	5	-1
0	-1	0

Tab. I.2. Masque de convolution du filtre passe-haut.

c. Filtre Gaussien

Le filtre Gaussien est un filtre isotrope spécial avec des propriétés mathématiques bien précises. La fonction gaussienne $G(x)$ est souvent utilisée dans les distributions statistiques.

Dans le traitement d'images, on traite des données à deux dimensions (x, y) , on introduit alors une fonction gaussienne à deux dimensions :

$$G(x, y) = \frac{1}{2\pi\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (I.3)$$

Etant données que les images numériques sont composées de valeurs discrètes, le noyau gaussien sera composé aussi de valeurs discrètes :

$$\frac{1}{6} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Tab. I.3. Masque de convolution du filtre Gaussien.

I.8.5.2. Filtres non linéaires

Ils sont conçus pour régler les problèmes des filtre linéaires, surtout ce qui concerne la mauvaise conservation des contours. Leur principe est le même que les filtres linéaires, il s'agit toujours de remplacer la valeur de chaque pixel par la valeur d'une fonction calculée dans son voisinage. La différence est que cette fonction n'est plus linéaire mais une fonction quelconque. Les filtres non linéaires les plus connus sont :

a. Le filtre médian

Le filtre médian appartient à la famille des filtres d'ordre. Les filtres d'ordre procèdent en remplaçant la valeur de chaque pixel par la valeur du pixel qui occupe un certain rang lorsqu'on trie les valeurs des pixels rencontrés dans un certain voisinage de ce pixel.

La valeur du pixel central est remplacée par la valeur médiane de tous les pixels de la fenêtre d'analyse centrée sur ce pixel. Le filtrage médian est très robuste à différents types de bruit.

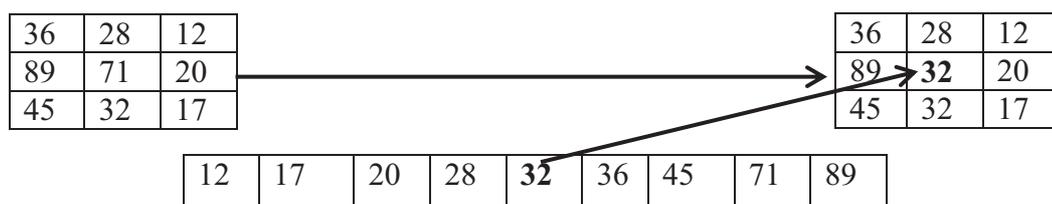


Fig. I.7. Principe du filtre médian.

b. Le filtre maximum

On applique le même traitement que celui du filtre médian mais la valeur du pixel du centre va être remplacée par le maximum.

c. Le filtre minimum

On applique le même traitement que celui du filtre médian mais, cette fois, la valeur du pixel du centre va être remplacée par le minimum [4].

I.8.6. Morphologie mathématique

La morphologie mathématique est une théorie et technique mathématique et informatique d'analyse de structures, elle est liée avec l'algèbre et s'effectue sur une image binaire. Une des idées de base de la morphologie mathématique est d'étudier ou de traiter un ensemble à l'aide d'un autre ensemble, appelé élément structurant (masque binaire constitué de pixels blancs et noirs), qui sert de sonde.

A chaque position de l'élément structurant, on regarde s'il touche ou s'il est inclus dans l'ensemble initial. En fonction de la réponse, on construit un ensemble de sortie. On obtient ainsi des opérateurs de base qui sont relativement intuitifs [3].

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

- **Erosion** : qui consiste à enlever un pixel sur les pourtours de la forme, agrandir les trous.
- **Dilatation** : consiste à ajouter un pixel sur les pourtours de la forme, à boucher les trous. [5].

I.8.7. La détection de contours

Le but de la détection de contours est de repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. La détection des contours d'une image réduit de manière significative la quantité de données et élimine les informations qu'on peut juger moins pertinentes, tout en préservant les propriétés structurelles importantes de l'image.

Le tableau (Tab.I.4) illustre les différents types de masques de convolution utilisables pour l'obtention du contour d'une image [6].

Type de masque	Gradients partiels	Extraction de l'amplitude	Extraction de la direction																		
Masques de Roberts Gradient selon x <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>-1</td><td>0</td></tr> <tr><td>0</td><td>1</td></tr> </table> Gradient selon y <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	-1	0	0	1	0	-1	1	0	$G_1 ; G_2$ Substitution du pixel supérieur gauche	$A = \sqrt{G_1^2 + G_2^2}$	$\theta = \frac{\pi}{4} + \arctan\left(\frac{G_2}{G_1}\right)$										
-1	0																				
0	1																				
0	-1																				
1	0																				
Masques de Sobel Gradient selon x <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>2</td><td>0</td><td>-2</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table> Gradient selon y <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>2</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-2</td><td>-1</td></tr> </table>	1	0	-1	2	0	-2	1	0	-1	1	2	1	0	0	0	-1	-2	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
1	0	-1																			
2	0	-2																			
1	0	-1																			
1	2	1																			
0	0	0																			
-1	-2	-1																			
Masques de Prewitt Gradient selon x <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> <tr><td>1</td><td>0</td><td>-1</td></tr> </table> Gradient selon y <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>-1</td><td>-1</td><td>-1</td></tr> </table>	1	0	-1	1	0	-1	1	0	-1	1	1	1	0	0	0	-1	-1	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
1	0	-1																			
1	0	-1																			
1	0	-1																			
1	1	1																			
0	0	0																			
-1	-1	-1																			
Masque Laplacien <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>-8</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	-8	1	1	1	1	L	$ L $										
1	1	1																			
1	-8	1																			
1	1	1																			

Tab. I.4 : les différents types de masques de convolution utilisée pour la détection du contour d'une image.

I.9. Conclusion

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

Dans ce chapitre, nous avons présenté les différents concepts liés à l'image numérique, on a parlé sur les types d'une image, ces caractéristiques et quelques formats de fichiers d'images, on a expliqué aussi la structure d'un fichier bitmap et finalement nous avons présenté quelques traitement d'images usuels.

II.1 Introduction

La classification est une activité mentale qui intervient fréquemment dans la vie courante. En effet, les objets sont souvent répertoriés par rapport à des classes ou des catégories auxquelles ils sont censés appartenir. Cette appartenance est, la plupart du temps, vague et/ou graduelle. Des modificateurs linguistiques tels que très, trop, assez, insuffisamment, traduisent l'incertitude que l'on peut rencontrer concernant les caractéristiques ou les propriétés de certaines entités.

II.2 Classification

La classification consiste à attribuer à chaque pixel dans l'image une classe (étiquette). Cette affectation peut être effectuée on se basant sur des régions dont on connaît les classes d'appartenance a priori, alors, on parle de classification supervisée, ou non, dans ce cas on parle de classification non-supervisée (segmentation/clustering).

La classification peut être représentée comme une fonction π qui associe à chaque pixel p l'indice k de la classe C_k à laquelle il appartient. De même que pour les régions R_k , le contenu des classes C_k doit être homogène (par exemple, les pixels qui composent une classe doivent avoir des valeurs f similaires).

Notant que les problèmes de segmentation et classification sont très liés et peuvent être utiliser pour signifier la même chose : Un classifieur implicitement segmente une image : l'ensemble des pixels ayant la même classe forme une région de l'image. Une segmentation implique une classification (les régions sont étiquetées selon leurs appartenances)

II.3 Méthodes de Classification

D'un point de vue général, les méthodes de classification ont pour but de regrouper les éléments d'un ensemble $X = \{X_1, \dots, X_n, \dots, X_N\}$ en un nombre C optimal de classes selon leurs ressemblances [8]

De manière générale, les problèmes de classification s'attachent à déterminer des procédures permettant d'associer un objet (individu) à une classe. Ces problèmes se déclinent essentiellement en deux variantes selon Bezdek [9] : la classification dite " supervisée " et la classification dite " non supervisée ".

La classification, supervisée ou non, en tant que discipline scientifique, n'a été automatisée et massivement appliquée que relativement récemment. Comme la plupart des activités scientifiques, l'essor des différentes techniques de classification a largement bénéficié de l'avènement et du perfectionnement des outils informatiques. De nos jours, la

classification est une démarche qui est appliquée dans d'innombrables domaines. Un autre nom possible pour cette branche de la recherche est la typologie, et la science qui lui est associée est la taxinomie. Les méthodes de classification ont pour but de regrouper les éléments d'un ensemble X , de nature quelconque, en un nombre restreint de classes. La qualité de la classification peut être jugée sur la base des deux critères suivants :

- Les classes générées doivent être les plus différentes possibles les unes des autres vis-à-vis de certaines caractéristiques
- Chaque classe doit être la plus homogène possible vis-à-vis de ces caractéristiques[10].

Si l'on se place dans un espace métrique, en termes d'inertie, le premier critère peut être traduit par le fait que l'inertie inter-classe doit être la plus élevée possible, et le deuxième critère traduit la préoccupation d'obtenir une inertie intra-classe la plus petite possible.

II.4 Méthodes supervisées

Ces sont des méthodes dans lesquelles les classes sont connues *a priori* avant d'effectuer l'opération d'identification des éléments de l'image. Elles demandent une **phase d'apprentissage** sur l'échantillon représentatif dans le but d'apprendre les caractéristiques de chaque classe et une autre phase pour décider l'appartenance d'un individu à telle ou telle classe.

Nous pouvons donc citer des algorithmes de classification pixels supervisés comme celui de Bayes [11], Les K plus proches voisins (K-PPV) ou les réseaux de Neurones Multicouches.

II.4.1 Algorithmes de Bayes

Ce sont des méthodes qui reposent sur la théorie de Bayes représentant une référence théorique pour les approches statistiques de résolution des problèmes de classification. Le principe de cette théorie est le suivant : Soit X_j le vecteur forme dont la classe est inconnue et qu'on veut la déterminer, et soit C_i une hypothèse (X_j appartient à la classe C_i par exemple). On cherche à déterminer $P(C_i/X_j)$ la probabilité de vérification de C_i après l'observation de X_j . $P(C_i/X_j)$ est la probabilité postérieure (après la connaissance de X_j) tandis que $P(C_i)$ est la probabilité à priori représentant la probabilité de vérification de C_i pour n'importe quel vecteur forme. Le théorème de Bayes propose une méthode de calcul de $P(C_i/X_j)$ en utilisant les probabilités $P(C_i)$, $P(X_j)$ et $P(X_j/C_i)$:

$$P(C_i / X_j) = \frac{P(C_i)P(X_j / C_i)}{\sum_{k=1}^C P(C_k).P(X_j / C_k)} \quad \forall i \in [1...C] \quad (\text{II. 1})$$

$P(C_i/X_j)$ est donc la probabilité d'appartenance de X_j à la classe C_i , $P(C_i)$ la probabilité d'apparition de la classe C_i (peut être calculée comme le rapport entre le nombre d'échantillons appartenant à la classe C_i et le nombre total d'échantillons). $P(X_j/C_i)$ peut être considérée comme la probabilité d'apparence de chaque valeur des attributs de X_j dans les attributs des échantillons appartenant à la classe C_i . Alors l'algorithme cherche à déterminer pour chaque élément X_j la classe C_i qui maximise la probabilité de contenir cet élément.

II.4.2 Algorithmes K plus proche voisins (K-PPV)

L'algorithme des k plus proches voisins (*KPPV*) (*k-Nearest-Neighbors* (*kNN*) en anglais) est une méthode non paramétrique et supervisée de classification introduite dans [12]. Elle est largement utilisée en classification d'une manière générale et en segmentation d'images en particulier. Elle repose sur un principe simple et intuitif de regroupement d'individus en fonction de leur voisinage.

L'algorithme de *K Plus Proche Voisin* se base essentiellement sur les deux éléments principaux suivants :

1. le nombre de cas les plus proches (K) à utiliser et une métrique pour mesurer le plus proche voisin.
2. La valeur de K est spécifiée à chaque utilisation de l'algorithme puisqu'il détermine le nombre de cas existants qui sont considérés pour prédire un nouveau cas.

Le *K Plus Proche Voisin* est basé sur le concept de distances. Une métrique est nécessaire pour déterminer les distances, cette dernière est à la fois importante car le choix de métrique influe beaucoup sur la qualité des prédictions et arbitraire du fait qu'il n'existe pas de définition préalable sur ce qui constitue une bonne métrique.

La méthode des *k plus proches voisins* repose sur le regroupement des pixels en fonction de leur voisinage : chaque point est affecté à la classe la plus représentée parmi ses k plus proches voisins. Cette méthode requiert l'établissement d'une règle de distance et la détermination du nombre de voisins à prendre en considération, ainsi qu'un ensemble d'apprentissage représentant les différentes classes.

II.4.3 Méthodes basées sur les réseaux de neurones artificiels

Un *réseau de neurones* est constitué d'un certain nombre d'unités élémentaires (neurones) fortement interconnectés (neurones) fonctionnant en parallèle et organisée en couches avec un flux d'informations à anticipation. L'architecture du perceptron multicouches

est organisée de la manière suivante: les signaux flux successivement dans les différentes couches de l'entrée à la couche de sortie. Les couches intermédiaires sont appelées couches cachées. Pour chaque couche, chaque unité élémentaire calcule un produit scalaire entre un vecteur de poids et le vecteur de sortie donné par la couche précédente. Une fonction de transfert est alors réalisée à la suite de faire une entrée pour la couche suivante. Les poids des connexions sont ajustés au cours du processus d'apprentissage pour obtenir la relation d'entrée/sortie désirée du réseau. De nombreux modèles de réseaux existent (réseaux de Hopfield, perceptrons multicouche, etc.),

Les méthodes de segmentation supervisée offrent l'avantage d'être plus rapides et plus reproductibles que les méthodes non supervisées. Toutefois, leur inconvénient est de rester très dépendant de la *base d'apprentissage*. C'est pourquoi il est intéressant de développer des algorithmes entièrement automatiques "*non supervisée*", qui présenteront l'avantage d'offrir un résultat reproductible et indépendant des actions de l'opérateur.

II.5 Méthodes non Supervisées

L'intérêt des méthodes *non supervisées* est qu'elles ne nécessitent **aucun apprentissage** et par là même aucune tâche préalable d'étiquetage manuel. Elles ont pour but de découper l'espace d'individus (pixels) en zones homogènes selon un critère de ressemblance (critère de proximité de leurs vecteurs d'attributs dans l'espace de représentation entre les individus).

Nous pouvons donc citer des algorithmes de classification pixels non-supervisés comme les C-moyennes proposée par Mac Queen [13], C-moyennes floues [15] [11] [14] ou Algorithmes les C-moyennes possibilistes [16]:

II.5.1 Algorithme des C-moyennes

L'algorithme des C-moyennes (C-means) (centres mobiles) est l'un des algorithmes de clustering les plus connus, il a été mis au point par Mac Queen à la fin des années soixante [13]. L'algorithme des C-moyennes (C-means) [13] réalise un partitionnement automatique des pixels de l'image en C classes homogènes. Soit $X = \{X_j : j=1, \dots, N\}$ un sous-ensemble d'un espace N-dimensionnel. Une partition de X en C classes peut être représentée par des sous-groupes mutuellement disjoints $C_1 \dots C_k$, tel que : $C_1 \cup \dots \cup C_k = X$.

Chaque vecteur X_j est assigné à une et seulement une classe qui lui est la plus proche. La fonction objectif correspondante s'exprime sous la forme

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

Ou: b_i désigne le prototype (centre) de cette classe.

d : mesure l'inverse de la similarité entre le prototype et le vecteur de données.

U_{ij} : est un indicateur binaire désignant l'appartenance ou non du vecteur x_j à la classe

C_i tel que U_{ij} ($U_{ij} = 1$ si $x_j \in C_i$ et $U_{ij} = 0$ sinon pour $i=1..C$). L'indicateur

d'appartenance U_{ij} s'exprime comme suit :

$$U_{ij} = \begin{cases} 1 & \text{si } d^2(x_j, b_i) = \min \{ d^2(x_j, b_k) \} \text{ pour } k=1..C \\ 0 & \text{sinon} \end{cases}$$

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

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

II.5.2 Algorithmes C-moyennes floues ("Fuzzy C-Means" ou FCM) :

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

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

$$\forall i \in [1, C], \forall j \in [1, N] \begin{cases} u_{ij} \in [0, 1] \\ 0 < \sum_{j=1}^N u_{ij} < N \end{cases} \quad (\text{II. 3})$$

$$\forall i \in [1, C] \sum_{j=1}^N u_{ij} = 1 \quad (\text{II. 4})$$

La fonctionnelle à minimiser J (5) et les solutions b_i (6), u_{ij} (7), au problème du FCM sont décrites par les formules suivantes :

$$J(B, U, X) = \sum_{i=1}^C \sum_{j=1}^N (U_{ij})^m d^2(x_j, b_i) \quad (\text{II. 5})$$

$$b_i = \frac{\sum_{j=1}^N (u_{ij})^m \cdot X_j}{\sum_{j=1}^N (u_{ij})^m} \quad (\text{II. 6})$$

$$u_{ij} = \left[\sum_{k=1}^C \left(\frac{d^2(X_j, b_i)}{d^2(X_j, b_k)} \right)^{\frac{2}{(m-1)}} \right]^{-1} \quad (\text{II. 7})$$

Avec la variable m est le coefficient de fuzzification qui prend ses valeurs dans l'intervalle $[0, +\infty[$. Lorsque m vaut 1, on retrouve l'algorithme classique et lorsque m tend vers l'infini, on obtient une partition infiniment floue où les coefficients U_{ij} tendent vers $1/C$. L'algorithme de FCM s'arrête lorsque la partition devient stable, c'est-à-dire lorsqu'elle n'évolue plus entre deux itérations successives.

Le FCM souffre de plusieurs inconvénients : [17]

- Les degrés d'appartenance sont des degrés relatifs. Autrement dit, l'appartenance d'un individu à une classe dépend de l'appartenance de cet individu aux autres classes. Les fonctions d'appartenance estimations des centres des classes ne correspondent pas aux centres réels ou typiques.
- Les points aberrants (éloignés) peuvent avoir des valeurs d'appartenance élevées et ils peuvent affecter de façon significative l'estimation des centres des classes.
- Ces algorithmes modélisent dans la phase de classement l'hésitation ou l'ambiguïté entre classes à partir de règles de décisions floues fixées a priori [18]. Dans la phase de classification, les points ambigus ou non représentatifs des classes peuvent alors affecter la position des centres.

II.5.3 Algorithmes les C-moyennes possibilistes

Krishnapuram et Keller ont proposé une approche possibiliste des c-moyennes appelée *Possibilistic C-Means*, ou PCM [16]. Leur approche est censée conduire à une meilleure performance en présence de bruit. Mais leur travail est motivé essentiellement par le désir de remédier au caractère relatif des degrés d'appartenance générés par les FCM. En effet, ces derniers sont interprétés en tant que degrés de vérité relatifs décrivant l'appartenance d'un vecteur quelconque à chacune des classes possibles. Un élément à classer est donc, en quelque sorte, partagé entre ces différentes classes. A cette idée de partage Krishnapuram et Keller préfèrent substituer la notion de typicalité. En effet, le résultat d'un regroupement devrait décrire la parenté absolue entre un objet et chacune des C classes possibles, indépendamment du lien entre cet objet et les (C-1) classes restantes.

Les coefficients U_{ij} ne reflètent plus l'appartenance d'un point X_j à une classe C_i mais son degré de compatibilité. Les hypothèses deviennent :

$$\forall i \in [1, C], \forall j \in [1, N] \left\{ \begin{array}{l} U_{ij} \in [0, 1] \\ 0 < \sum_{j=1}^N U_{ij} < N \end{array} \right. \quad (\text{II. 8})$$

$$\forall i \in [1, C] \max_j(u_{ij}) > 0 \quad (\text{II. 9})$$

Le nouveau critère à minimiser est :

$$J(B, U, X) = \sum_{i=1}^C \sum_{j=1}^N (U_{ij})^m d^2(x_j, b_i) + \sum_{i=1}^C \eta_i \sum_{j=1}^N (1 - U_{ij})^m \quad (\text{II. 10})$$

Dans cette expression, le premier terme correspond au critère, de FCM, de l'équation (5). Le second terme impose des valeurs les plus grandes possibles aux possibilités d'appartenance. La valeur de η_i fixe la distance à partir de laquelle la possibilité d'appartenance est égale à 0,5. Krishnapuram propose de prendre une valeur proportionnelle à la distance moyenne intra-classe. L'algorithme est initialisé avec la matrice de partition obtenue par la méthode FCM. La valeur de η_i est déterminée par :

$$\eta_i = \frac{\sum_{j=1}^N (u_{ij})^m \cdot d(X_j, b_i)}{\sum_{j=1}^N (u_{ij})^m} \quad (\text{II. 11})$$

Les centres et les possibilités d'appartenance sont calculés de manière itérative par les expressions :

$$g_i = \frac{\sum_{j=1}^N (u_j)^m \cdot X_j}{\sum_{j=1}^N (u_j)^m} \quad (\text{II. 12})$$

$$u_j = \left[1 + \left(\frac{(d(X_j, b_i))^2}{\eta_i} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (\text{II. 13})$$

Le calcul se termine lorsque la différence entre chaque possibilité d'appartenance et la possibilité de l'itération précédente est inférieure à un seuil ε fixé.

La méthode PCM permet de résoudre le problème des points éloignés des classes en leur associant une possibilité d'appartenance plus faible que le coefficient d'appartenance de nature probabiliste défini par la méthode FCM, rendant possible l'introduction des notions de rejet. Cependant, PCM ne solutionne ni le problème des classes de forme complexe, ni celui de la recherche du nombre de classes dans un nuage de points. Son utilisation demeure restreinte à la caractérisation de classes de formes hyper-sphériques et hyper-elliptiques ayant une orientation parallèle aux axes [19].

II.6 Segmentation d'images

Lorsqu'un être humain observe une image naturelle, il voit généralement des objets physiques ou leurs parties. Il peut donc diviser l'image en régions, ou segments, les représentants. Nous allons étudier dans ce chapitre comment une machine peut reproduire cette tâche de segmentation.

La segmentation est l'opération qui consiste à regrouper les individus d'une population en un nombre limité de groupes, les segments (ou clusters, ou partition), qui ont deux propriétés : D'une part, ils ne sont pas prédéfinis, mais découverts automatiquement au cours de l'opération, contrairement aux classes de la classification. D'autre part, les segments regroupent les individus ayant des caractéristiques similaires et séparent les individus ayant des caractéristiques différentes (homogénéité interne et hétérogénéité externe). Appliquée à l'image, la segmentation a pour but de partitionner l'espace des pixels E d'une image f en un ensemble de K régions $\{R_k\}_{1 \leq k \leq K}$ homogènes au sens d'un certain critère (par exemple les valeurs v des pixels composants chaque région). Il s'agit donc d'une fonction $\pi : E \rightarrow \mathbf{R}$ qui associe à chaque pixel p l'indice k de la région R_k à laquelle il appartient. Chaque région R_k est construite comme une composante connexe, c'est-à-dire un ensemble de pixels adjacents (ou voisins deux à deux) de valeur k .

Alors, le but de la segmentation est de partitionner une image en des régions connexes et homogènes telles que des pixels voisins doivent appartenir à la même région. C'est un problème de classification très difficile à résoudre car l'espace des configurations possibles est très vaste.

II.7 Objectifs de la segmentation [20]

- L'extraction d'éléments et d'informations qualitatives de l'image.
- Fournir des régions homogènes
- Localiser les contours de région.
- Fournir une description de haut niveau : représentation par un graphe dont les nœuds sont des régions, chaque région est connectée à ses voisines et chaque région porte une étiquette qui donne des informations comme la taille, la couleur, la forme,...
- Réduire le bruit.

II.8 Approches de segmentation

Dans la littérature, plusieurs manières de catégoriser les méthodes de segmentation d'images ont été proposées : [21] [22], [23], [24]...etc. Shankar [21] les a classées en quatre classes : a) Approche par contours b) Approche Pixels c) Approche régions d) Approche hybride. Ameer [22] en deux classes principales: Approche frontières et approche régions. Guo [23], aussi en deux classes : couleur et texture. Pour mieux connaître l'ensemble des différentes classifications consulter [24].

Dans ce travail on va se contenter des trois catégories suivantes : Approches basées contour, Approches basées région, et Approches basées classification

II.9 Approches basées Contour

De façon général, un contour est défini comme étant la frontière entre deux régions. La détection de contours consiste à repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. Les approches contour opèrent à trouver les zones de variations significatives d'intensité lumineuse (niveaux de gris) ou de couleur dans l'image. Nous pouvons citer les approches se basant sur l'opérateur de gradient, l'opérateur Laplacien et les différents filtres à savoir : le filtre de Sobel, Prewitt et Roberts ou bien des approches analytiques comme le filtre de Canny [25]. Mais ce genre de techniques est peu utilisable car il donne souvent des contours non fermés, bruités ou non détectés ; une utilisation des propriétés des régions comprise entre ces contours pourrait nettement améliorer la détection de ce derniers [26].

II.10 Approches basées région

L'approche région cherche à regrouper les pixels en régions homogènes. Elle se caractérise par la mesure d'uniformité des régions construites dans l'image. Ces régions sont construites en évaluant la similarité entre les pixels ou entre un pixel et ceux d'une même région. On distingue les méthodes par division, fusion et par division/fusion.

II.10.1 Méthode par division (Top-Down)

Ce sont des approches qui opèrent en découpant l'image originale en régions plus petites suivant un critère d'hétérogénéité. Le découpage s'arrête quand il n'y a plus de régions inhomogènes [27]. L'approche Quadtree est souvent la plus utilisée pour réaliser ce découpage. Il s'agit d'une décomposition récursive en quatre sous régions carrées des régions hétérogènes en commençant par l'image entière.

II.10.2 Méthode par fusion (Bottom-Up)

Ces méthodes opèrent en regroupant des pixels ou des groupes de pixels selon un critère d'homogénéité pour obtenir un ensemble de régions homogènes [27]. Elles suivent une hiérarchie arborescente. Leurs inconvénients majeurs c'est que le résultat final (image segmentée) dépend fortement du choix des régions de départ.

II.10.3 Méthode par division/fusion (Split and Merge)

Proposées par Horowitz [28], elles réunissent l'ensemble des algorithmes utilisés dans les techniques précédemment citées (méthodes par fusion et méthodes par division). Le processus de segmentation se déroule en deux étapes: la division en blocs (régions) suivant un critère d'hétérogénéité est effectuée en premier lieu ; puis, suit l'étape de fusion où toutes les régions adjacentes qui découlent de l'étape de division et vérifiant une heuristique d'homogénéité sont regroupées.

II.11 Approches basées classification

Ce type de méthode considère une région comme un ensemble de pixels connexes appartenant à une même classe. Elle suppose donc que les pixels qui appartiennent à une même région possèdent des caractéristiques similaires et forment un nuage de points dans l'espace des attributs. La classification consiste à retrouver ces nuages de points qui correspondent aux classes des pixels présentes dans l'image.

II.11.1 Algorithme K-means

C'est l'un des algorithmes les plus connus en classification non supervisée. Il vise à produire un partitionnement des pixels de manière à ce que les pixels d'une même classe

soient semblables et les pixels issus de deux classes différentes soient dissemblables. L'idée principale est de définir C centres, un pour chaque classe. Chaque classe C_i est ainsi caractérisé par son centre noté b_i et le nombre d'éléments N_i . L'algorithme C-Means dans sa formulation originale cherche à minimiser une fonction de coût global définie par :

$$J = \sum_{i=1}^C \sum_{(x,y), k \in C_i} d(f(x,y), b_i)^2 \quad (\text{II. 14})$$

Où $f(x,y)$ représente le niveau de gris du pixel de coordonnées (x,y)

Il se déroule selon les étapes suivantes :

1. Initialisation de chaque centre b_i

Pour chaque pixel (x,y) , calculer la distance $d(f(x,y), b_i)$ aux différents centres des classes b_i , et affecter à la classe la plus proche C_i avec $d(f(x,y), b_i) = |d(f(x,y) - b_i)|$

3. Mise à jour de nombre de pixels et des centres b_i des classes $b_i = \frac{\sum_{(x,y) \in C_i} f(x,y)}{N_i}$
4. Arrêt si $N_i = N_{i+1} \forall (x,y) \in C_i$, sinon retour à l'étape 2.

Le principal inconvénient de cette méthode est que la classification finale dépend du choix de la partition initiale. Le minimum global n'est pas obligatoirement atteint, on est seulement certain d'obtenir la meilleure partition à partir de la partition de départ choisie [29]. De nombreuses variantes peuvent être rencontrées. Par exemple, au lieu de calculer le centre des classes, après avoir affecté tout les pixels, les centres de gravité peuvent être calculés immédiatement après chaque affectation. La méthode des K-means a été généralisée sous l'appellation de la "méthode des nuées dynamiques" [30]. Au lieu de définir une classe par un seul point (son centre de gravité), elle est définie par un groupe de points (noyaux de classe). Un autre algorithme proposé dans la littérature et qui est issu de l'algorithme K-means est l'algorithme ISODATA [31]. L'avantage de ce dernier est qu'il permet de regrouper les pixels sans connaître a priori le nombre exact de classes présentes dans l'image. Ce nombre pourra être modifié au cours des itérations.

II.11.2 Algorithme Estimation-Maximisation (EM)

L'algorithme EM a été développé par Dempster et al. En 1977 [32] et a été utilisé dans de nombreux domaines, notamment en segmentation d'images [33] [34]. Nous allons présenter le principe de base de cet algorithme : pour plus de détails voir [32]. Nous nous plaçons dans le cas d'une approximation d'un histogramme d'une image par une combinaison de gaussiennes. Le problème est d'arriver à estimer les paramètres des différentes gaussiennes qui approchent le mieux l'histogramme. Par exemple, dans le cas de deux classes c'est-à-dire

d'un histogramme approché par deux gaussiennes, le vecteur de paramètres à estimer est θ [P1, u1, σ 1, P2, u2, σ 2] associé aux deux classes (objet et fond). Cette estimation est faite au sens de la maximisation du critère de maximum de vraisemblance $L(X|\theta)$ défini par :

$$L(X|\theta) = \sum_{X=0}^{L-1} h(X) \log [p(X|\theta)] \quad (\text{II. 15})$$

La solution du problème ci-dessus ne peut pas être obtenue analytiquement, car le problème est intrinsèquement mal posé. L'algorithme EM va donc être utilisé afin d'y remédier. Cet algorithme est basé sur l'interprétation de X comme un vecteur de données, incomplet, où la partie qui manque est Y (l'image segmentée). En d'autres termes, la partie manquante peut être évaluée comme un ensemble de L étiquettes $Z = \{z(x) : x \in [0, L-1]\}$. Chaque étiquette $z(x)$ de Z indique la composante (objet ou fond) qui est à l'origine de la réalisation x . Dans notre cas, l'ensemble Z est binaire : $z(x) = \{z_1(x), z_2(x)\}$, avec $z_i(x) = 1$ si x appartient à la composante i et $z_i(x) = 0$ dans le cas contraire. Le critère de segmentation final est :

$$L(X, Z, \theta) = \sum_{i=1}^2 \sum_{X=0}^{L-1} Z_i(x) h(X) \log [P_i(X|\mu_i, \sigma_i)] \quad (\text{II. 16})$$

L'algorithme EM consiste à itérer les deux étapes : Espérance et Maximisation, jusqu'à la convergence totale de l'algorithme. L'étape d'espérance va permettre de calculer les étiquettes $z_i(x)$, en utilisant les paramètres estimés en cours. L'étape de maximisation va permettre de mettre à jour l'estimation du vecteur θ . La méthode est détaillée dans [33][34].

II.11.3 Classification par Seuillage

Dans beaucoup d'applications de traitement d'images, on considère que les niveaux de gris des pixels appartenant à un objet sont relativement différents de ceux du fond [35] [36] [37] [38] [39]. Le seuillage est une technique simple et efficace pour séparer les objets du fond. Toutefois, la difficulté apparaît dans le choix des seuils optimaux pour une séparation optimale. Soit 'I' l'image originale et 'Is' l'image segmentée. Segmenter I en N classes revient à chercher N-1 seuils. La classification de chaque pixel est définie comme suit :

$$I_s(x,y) = k \text{ si } k \leq I(x,y) < S_{k+1} \text{ pour } k=0, \dots, n$$

Où x, y sont les coordonnées du pixel ; S_0, \dots, S_N sont les différents seuils de segmentation.

A titre d'exemple, nous allons présenter un seuillage manuel qui comporte quatre étapes :

1. Observation de l'histogramme de l'image
2. Choix des seuils dans les vallées
3. Définition des classes de régions par intervalle de couleurs
4. Classification des pixels

Dans la littérature, on y trouve plusieurs classifications des méthodes de segmentation par seuillage [35] [36] [37] [38] [39] pour plus de détails voir le travail. Sezgin [37]. Les auteurs les ont classées en deux catégories :

- Les méthodes non paramétriques : qui ont la réputation d'être robustes ; elles sont basées sur l'optimisation d'un ou plusieurs critères comme, par exemple, maximiser la variance interclasse. La plus connue d'entre elles est la méthode d'Otsu [38].
- Les méthodes paramétriques : qui sont basées sur l'hypothèse que les densités de probabilité des niveaux de gris des différentes classes sont gaussiennes, une approximation de l'histogramme de l'image à traiter par une combinaison linéaire de gaussiennes est toujours effectuée avant l'opération de seuillage, ou les seuils optimaux sont à l'intersection de ces dernières [36]

II.12 Exemples d'applications

La classification et la segmentation ont été utilisées dans de nombreuses applications dont certaines concernent: l'étude du sommeil ,le diagnostic médical ,le traitement d'images aériennes, la segmentation d'images RMN (Résonance Magnétique Nucléaire), le traitement des eaux usées ,l'étude du climat, les domaines de l'agriculture, l'astronomie, la chimie, la géologie, le traitement d'image, la reconnaissance de la maturité de squelettes à partir de radiographies aux rayons X, la reconnaissance de la parole, la reconnaissance des caractères écrits, le traitement automatique des documents,...etc.

II.13 Conclusion

Dans ce chapitre, nous avons donné un aperçu sur les différentes approches utilisées pour la classification : supervisées et non supervisées et pour la segmentation d'images : les approches basées contour, basées région et les approches basées classification. Nous avons vu que la segmentation (clustering) permet de regrouper des objets (individus ou variables) en un nombre limité de groupes ou de classes (segmentes, ou cluster). La classification consiste à grouper les pixels de l'image ayant des caractéristiques assez similaires souvent au niveau de gris, en des sous ensembles de classes.

III.1. Introduction

L'empreinte digitale est l'une des caractéristiques les plus fréquemment employées dans le domaine d'identification d'auteurs de crimes ou délits ainsi que de personnes. L'empreinte, qu'elle soit digitale, palmaire ou plantaire est sans nul doute, le meilleur élément individualisant. Cette individualité est directement liée au phénomène de morphogenèse ainsi qu'à la nature de la trace qui se définit comme un mécanisme d'apparition des crêtes papillaires.

La classification est le processus qu'associe un objet inconnu à un ensemble, de sorte que les éléments de cet ensemble ont les mêmes caractéristiques que cet objet. Dans le domaine d'identification des individus par les empreintes digitales, il est souhaitable de classer les empreintes digitales en classes de sorte que l'empreinte digitale en entrée doit être assortie seulement avec un sous ensemble de la base des empreintes et donc réduire le temps de recherche.

III.2. Historique

Les premières traces d'utilisation des empreintes digitales ont été découvertes en Egypte et datent de l'époque des pyramides il y a plus de 4000 ans. Les Chinois ont aussi utilisé très tôt ce moyen pour signer les documents officiels, mais ils ne savaient sûrement pas que les empreintes étaient uniques pour chaque personne et permettaient ainsi une identification fiable. En 1892 l'anthropologue anglais Francis Galton étudie les empreintes digitales pendant dix ans et publie un ouvrage *Finger-Prints*, il établit une classification expérimentale de plus de 2500 séries d'empreintes. Il calcule qu'il y a seulement une chance sur 64 milliards que deux individus aient la même empreinte. En 1898 Edward Richard Henry chef de la police londonienne a mis en place un système de classification des empreintes, ce système a été largement adopté et a remplacé le Bertillonage.

De nos jours les empreintes sont largement utilisées et reconnues comme méthode d'identification fiable [40].

III.3. L'empreinte digitale

Une empreinte digitale est une marque laissée par les stries des doigts lorsqu'elles touchent un objet. Chaque personne dans le monde a ses propres empreintes digitales avec l'unicité permanente. Ainsi, les empreintes digitales sont utilisées depuis longtemps pour l'identification et l'investigation juridique.

Une empreinte digitale se compose de beaucoup des stries (crêtes) et vallées (voir Fig.III.1). Ces stries et vallées présentent de bonnes similitudes dans chaque petite fenêtre locale, comme le parallélisme et la largeur moyenne, il en existe deux types : l'empreinte directe (qui laisse une marque visible) et l'empreinte latente (saleté, sueur ou autre résidu déposé sur un objet) [41].

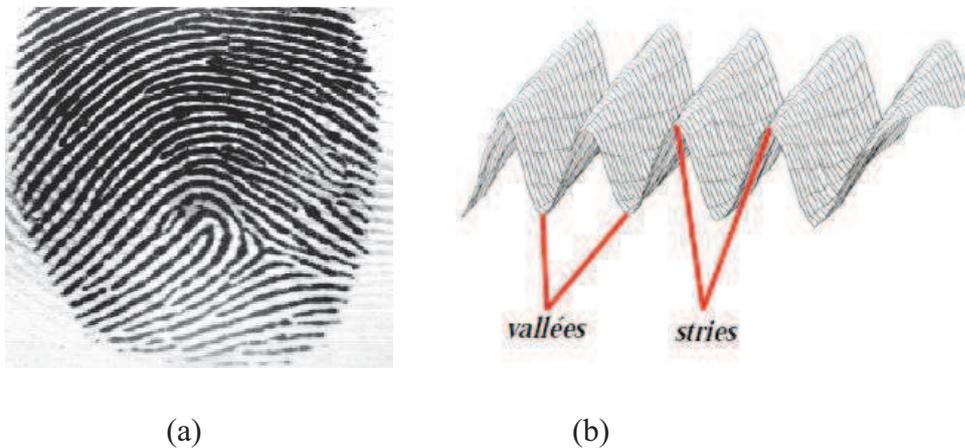


Fig.III.1. *L'empreinte digitale (a) Exemple d'une empreinte digitale (b) vallées et stries.*

III.4. Caractéristiques des empreintes digitales

Les caractéristiques qui permettent de différencier deux empreintes digitales ayant le même motif sont : les points singuliers globaux qui sont utilisées généralement pour la classification des empreintes digitales, et les points singuliers locaux qui sont utilisées pour l'identification des individus.

III.4.1. Les points singuliers globaux

Les points singuliers globaux sont de deux types :

- **Le core:** qui est le lieu de courbure maximale des lignes de l'empreinte les plus internes. Il est aussi appelé **centre** ou également **noyau**.

- **Le delta** : qui est le lieu de divergence des lignes les plus internes, en d'autres termes un delta est proche du lieu où se séparent deux lignes (voir Fig. III.2).

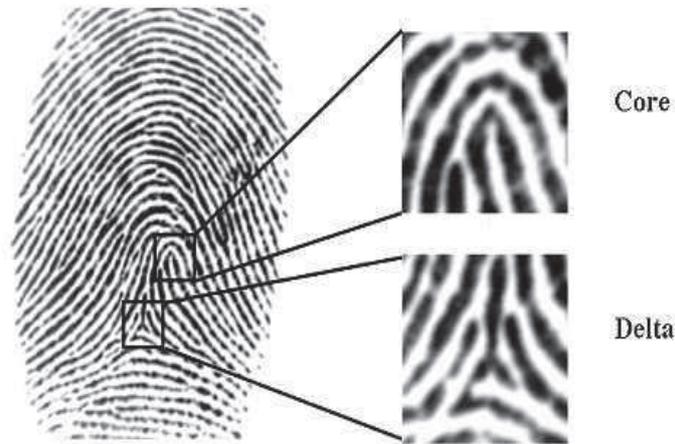


Fig. III.2. Les points singuliers globaux, core et le delta.

III.4.2. Les points singuliers locaux (minuties)

Dans la biométrie des empreintes digitales, le terme minuties désigne les différentes discontinuités de lignes d'une empreinte. On peut relever jusqu'à seize types de minuties mais dans les algorithmes on n'en retient généralement que quatre types (voir Fig.III.3) :

- Terminaison à droite ou à gauche : minutie située en fin de strie (voir figure (a))
- Bifurcation à droite ou à gauche : intersection de deux stries (voir figure (b))
- Île: assimilée à deux terminaisons : (voir figures (d, e))
- Lac: assimilée à deux bifurcations : (voir figure (c))

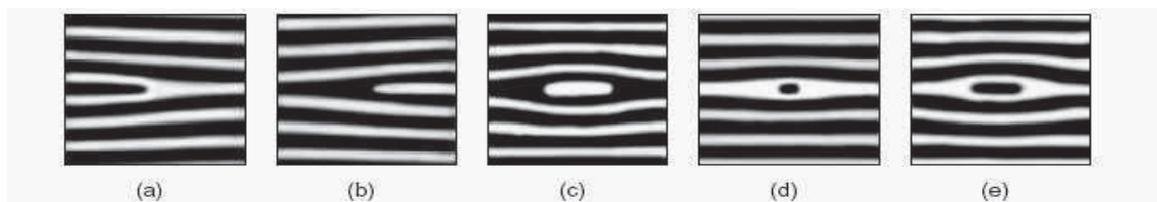


Fig. III.3. Les différents types de minuties.

III.5. Classes des empreintes digitales

Les premières études scientifiques sur la classification des empreintes digitales ont été faites par Francis Galton (1822-1916), qui a classé les empreintes digitales en trois grandes catégories : arc, boucle et spire, ensuite Edward Henry (1850-1931) a affiné la classification de Galton par l'augmentation de nombre des classes en cinq classes : arc, arc tendu, boucle à gauche, boucle à droite, et spire, qui est actuellement utilisé, la figure (Fig.II.4) illustre des exemples des cinq classes des empreintes digitales.

1. **Classe arc** : une empreinte d'arc a les crêtes qui entrent d'un côté, se lèvent à une petite bosse, et sortent du côté opposé dont elles sont entrées. Les empreinte d'arcs n'ont pas des cores ou des deltas;
2. **Classe arc tendu** : une empreinte d'arc tendu est semblable à une empreinte digitale d'arc, sauf qu'au moins une crête montre une courbure élevée et elle contient un core et un delta.
3. **Classe boucle**: une empreinte de boucle a une ou plusieurs crêtes qui entrent d'un côté, se courbent en arrière, et sortent du même côté qu'elles sont entrées. Un core et un delta sont présents. Les boucles peuvent être encore subdivisées en :
 - **Empreinte de boucle à gauche** : qui ont des crêtes qui entrent et sortent du côté gauche.
 - **Empreinte de boucle à droite** : qui ont des crêtes qui entrent et sortent du côté droit.
4. **Classe spire** : une empreinte de spire contient au moins une crête qui fait un chemin 360° autour du core de l'empreinte digitale. Deux centres et deux deltas peuvent être trouvés dans des empreintes digitales de spire. [41]



a. Arc

b. Arc tendu

c. Boucle à gauche



d. Boucle à droite

e. Spire

Fig. III.4. Les cinq classes des empreintes digitales [46].

Cependant, les empreintes digitales ne sont pas uniformément distribuées parmi ces cinq classes : les proportions sont 3,7%, 2,9%, 33,8%, 31,7% et 27,9% pour les arcs, les arcs tendus, les boucles gauches, les boucles droites et les spires, respectivement. Ainsi, trois de ces cinq classes représentent plus de 93% de la population totale [42].

III.6. Intérêts de l'empreinte digitale

Les empreintes digitales ont des caractéristiques qui les rendent intéressantes à utiliser pour l'identification des individus. En voici un résumé :

- *Universelles* : Plus de 96% de la population mondiale en possède, ce qui permet d'avoir un moyen de caractérisation important pour tous les humains, indépendamment de leur origine.
- *Uniques* : Chaque personne a une empreinte qui lui est propre.
- *Persistantes* : Les coupures et autres blessures au niveau du doigt n'altèrent pas l'empreinte (il faudrait brûler le doigt en profondeur pour cela).
- *Performantes* : Les traitements applicables pour comparer deux empreintes ne sont pas trop lourds et peuvent être effectués dans des temps raisonnables.
- *Acquisitions transparentes* : L'acquisition n'est ni invasive (pas de piqûre), ni contraignante pour l'utilisateur (juste le doigt à appliquer sur un capteur).

III.7. Description d'un système de classification des empreintes digitales

Un système de classification des empreintes digitales est une chaîne de processus qui à partir d'une image d'empreinte digitale en entrée renvoie un résultat en sortie qui est la classe auxquels appartient cette empreinte digitale (voir Fig.III.5). La première phase de ce système permet d'obtenir une image d'empreinte, qui ensuite va subir un prétraitement pour extraire l'information utile, ensuite ces informations sont stockées dans la base de données. Ainsi, le système de classification va classer l'empreinte dans une des cinq classes d'empreinte.

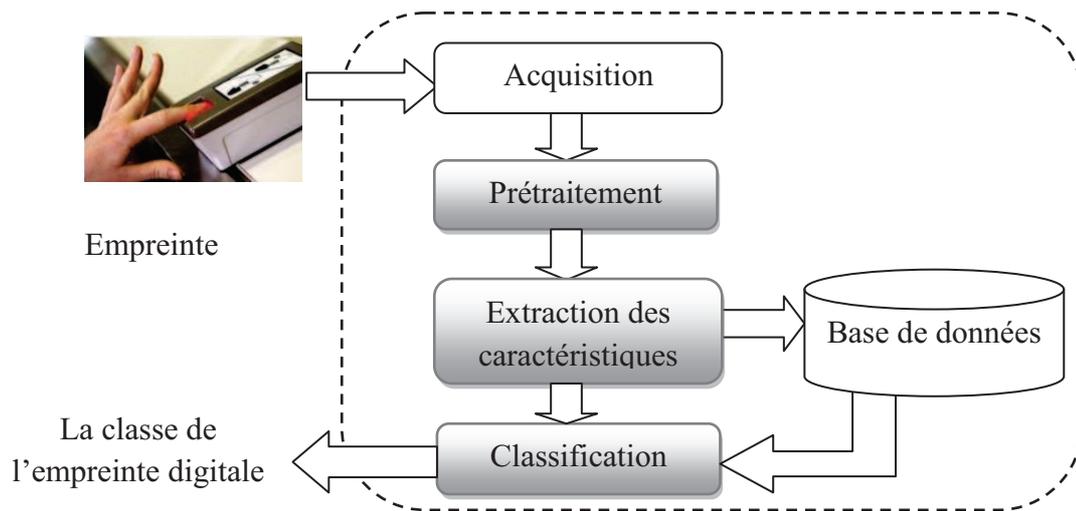


Fig.III.5. Schéma générale d'un système de classification des empreintes digitales.

III.7.1. L'acquisition de l'empreinte

La première étape d'un système de classification des empreintes digitales consiste à obtenir une image de l'empreinte du doigt. C'est le rôle du capteur qui consiste à trouver les lignes tracées par les crêtes de l'empreinte qui se trouvent en contact avec le capteur et les vallées qui ne le touchent pas. Il existe trois principaux types de capteurs :

- **Capteur Optique** : Il s'agit plus ou moins d'une caméra digitale. On applique le doigt sur une platine en plastique dur. Ce type de capteur résiste bien aux fluctuations de températures mais il est gêné par la luminosité ambiante et reste assez volumineux. Il est privilégié pour son coût assez faible et permet d'avoir des images assez nettes et précises :
- **Capteur Silicium** : Il utilise l'un des quatre effets observables sur les semi-conducteurs : l'effet piézo-électrique, l'effet capacitif, l'effet thermoélectrique ou l'effet photo-électrique. Sa taille et son coût sont assez réduits et il possède une durée de vie importante. Par contre, il est fragile aux décharges électrostatiques. Exemple de capteur silicium : les capteurs thermiques
- **Capteur Ultra Sonique** : La capture est possible grâce à l'envoi d'ultra - son dirigé très précisément vers des zones de l'empreinte. Le capteur calcul alors le temps que met l'onde pour faire un aller - retour. L'image est générée point par point. Ce capteur est très précis. De plus, il possède des avantages propres aux ultra-sons, c'est-à-dire que les ondes peuvent traverser certains matériaux comme le latex ou la saleté qui peut gêner la bonne capture de l'empreinte. Ce type de capteur est par contre très onéreux et volumineux. Il conviendra plutôt à une population hétérogène.

N'oublions pas l'encre qui a été pendant très longtemps la seule solution pour relever les empreintes. Cette méthode est aujourd'hui assez obsolète de par l'obligation d'avoir un matériel adéquat pour les relever et du fait qu'il faille ensuite acquérir l'empreinte relevée avec un scanner, d'où une redondance dans les manipulations [43].

III.7.2. Le prétraitement

Les algorithmes de classification des empreintes digitales sont sensibles à la qualité des images des empreintes digitales. Alors l'étape de prétraitement est nécessaire avant d'effectuer l'étape d'extraction des caractéristiques.

La qualité des images des empreintes digitales dépend de plusieurs facteurs comme : le contact avec le sonde, la qualité de sonde, la profondeur de stries et vallée, ... Typiquement le prétraitement peut se composer du lissage, l'amélioration de contraste, le filtrage.

III.7.3. L'extraction des caractéristiques

La très grande majorité des techniques de classification des empreintes digitales sont basés sur la détection de singularités globales des empreintes (cores et deltas) et l'extraction de leurs caractéristiques (nombre et position), afin de détecter les points core et delta en passe par les étapes suivantes :

III.7.3.1. Normalisation

Soit l'image entrée I , est une image en niveau de gris $N \times M$ pixels où $I(i, j)$, $i=1 \dots N$, $j=1 \dots M$ représente la couleur en niveau de gris à la $i^{\text{ème}}$ ligne et la $j^{\text{ème}}$ colonne. La valeur moyenne de l'empreinte digitale est défini par :

$$\text{Moy}(I) = \frac{1}{N * M} \sum_{i=1}^N \sum_{j=1}^M I(i, j) \quad (III. 1)$$

Et la variance de l'empreinte digitale est défini par :

$$\text{VAR}(I) = \frac{1}{N * M} \sum_{i=1}^N \sum_{j=1}^M (I(i, j) - \text{Moy}(I))^2 \quad (III. 2)$$

L'image originale est normalisée en utilisant sa valeur moyenne $\text{Moy}(I)$ et sa variance $\text{VAR}(I)$.

La matrice $G(I)$ indique l'image normalisée en niveau de gris et $G(i, j)$ représente sa couleur à (i, j) .

$$G(i, j) = \begin{cases} M_0 + \sqrt{\frac{VAR_0(I(i, j) - Moy(I))^2}{VAR}}, & \text{Si } I(i, j) > M \\ M_0 - \sqrt{\frac{VAR_0(I(i, j) - Moy(I))^2}{VAR}}, & \text{autrement} \end{cases} \quad (III.3)$$

M_0 et VAR_0 sont les valeurs de moyenne et de variance désirées. [44]

III.7.3.2. Estimation de champ d'orientation d'image d'empreinte digitale

Le champ d'orientation d'une image d'empreinte digitale définit l'orientation locale des crêtes contenues dans l'empreinte (voir Fig.III.6) [45].

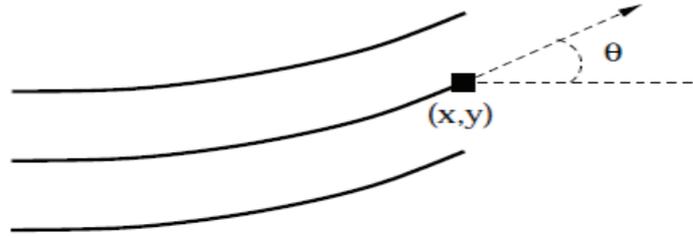


Fig.III.6. L'orientation de la crête locale du pixel (x, y) .

Les étapes pour calculer l'orientation au pixel (i, j) sont comme suit:

- Étape 1 : Diviser l'image G par bloc de $w \times w$ (16 x 16).
- Étape 2 : Pour chaque pixel (i, j) , calculer les gradients $\partial_x(i, j)$ et $\partial_y(i, j)$, qui sont les grandeurs de gradient dans les directions x et y respectivement, pour ce faire on utilise l'opérateur de Sobel.
- Étape 3 : L'orientation locale au pixel (i, j) peut alors être estimée en utilisant les équations suivantes:

$$V_x(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2\partial_x(u, v)\partial_y(u, v) \quad (III.4)$$

$$V_y(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (\partial_x^2(u, v) - \partial_y^2(u, v)) \quad (III.5)$$

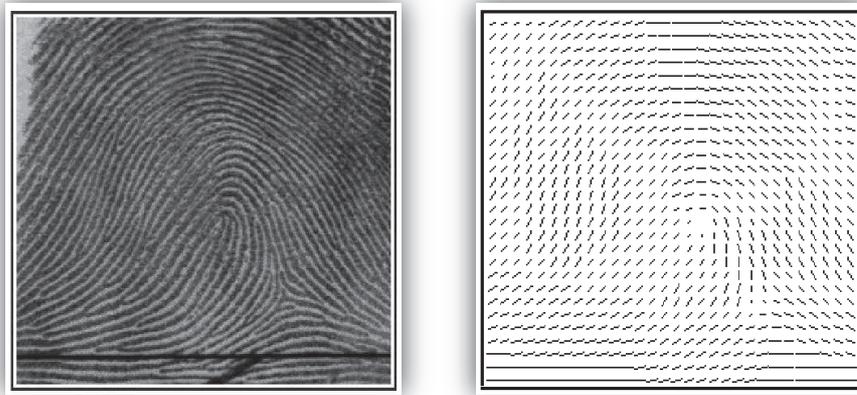
$$O(i, j) = \frac{V_x(i, j)}{V_y(i, j)} \quad (III.6)$$

$$\theta_1(i, j) = \frac{1}{2} \tan^{-1} O(i, j) \quad (III.7)$$

$$\theta(i, j) = \theta_1(i, j) + \frac{\pi}{2} \quad (III.8)$$

Où $O(i, j)$ est l'estimation de moindre carré de l'orientation de bloc centré au pixel (i, j) , et $\theta(i, j)$ est l'angle d'orientation.

La figure (Fig.III.7) illustre le résultat d'estimation de champs d'orientation d'une image d'empreinte digitale.



(a) image originale

(b) image d'orientation

Fig.III.7. L'image d'orientation d'une empreinte digitale [47].

III.7.3.3. Segmentation

L'étape de segmentation consiste à éliminer les blocs appartenant au fond de l'image en fonction de leur valeur moyenne. La valeur moyenne locale M_l est calculée et comparée avec un seuil T :

- Si $M_l < T$, alors on considère que le bloc ne contient aucune information utile et il est éliminé.
- Si $M_l \geq T$, alors le bloc contient une partie de l'empreinte et est conservé.

Le seuil T a été choisi en fonction de la valeur moyenne M_g de l'image globale tel que

$$T = \frac{M_g}{2}.$$

- Tout bloc non bruité ayant au moins 3 blocs adjacents définis comme bruités est éliminé.

- L'ensemble des blocs au bord de l'image est supprimé.

Un exemple de segmentation d'image d'empreinte digitale est illustré sur la figure (Fig.III.8).

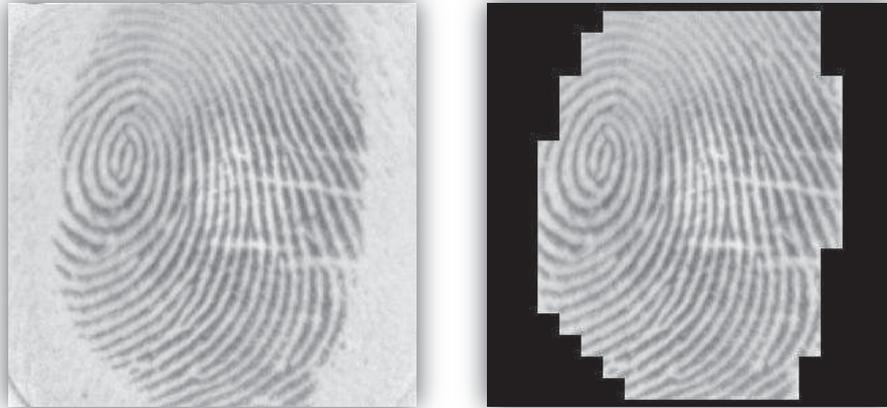


Fig.III.8. Effet de la segmentation sur l'image (a) Image d'entrée (b) Image segmentée [48].

III.7.3.4. Détection des points singuliers globaux

Il existe plusieurs algorithmes de détection des caractéristiques globales d'une empreinte digitale. Cependant, la plupart d'entre eux se basent sur l'orientation de l'empreinte digitale.

Dans cette thèse, on utilise l'un des plus pratiques, simples et élégants: c'est l'algorithme de l'index de Poincaré proposé par Kawagoe et Tojo [49].

Algorithme de l'index de Poincaré

Soit une courbe C dans un champ de vecteurs G . L'index de Poincaré $P_{G,C}$ est la somme des rotations des vecteurs de G le long de C . C'est - à - dire la somme des angles entre les vecteurs adjacents le long de la courbe C (voir Fig.III.9).

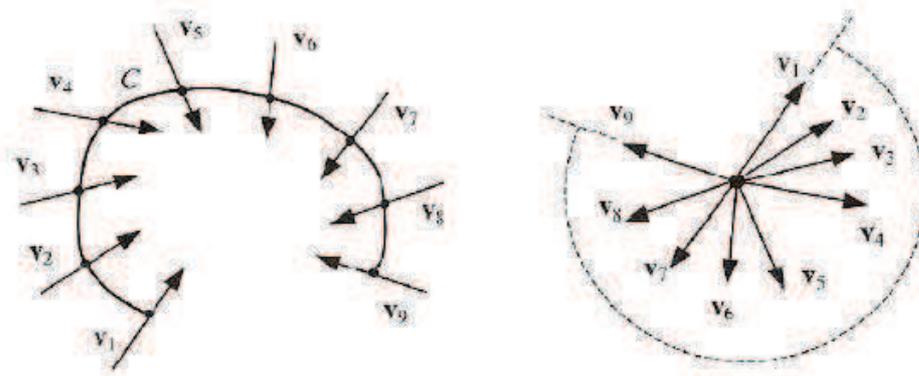


Fig.III.9. L'index de Poincaré calculé sur une courbe C [47].

Dans le cas des empreintes digitales, G est remplacé par l'orientation de l'empreinte digitale θ . Soit un pixel $[i, j]$ de la matrice θ , l'index de Poincaré $P_{G,C}(i, j)$ est calculé sur une courbe fermée C, entourant $[i, j]$, en faisant la somme des différences des orientations des lignes de l'empreinte aux pixels adjacents de C. On considère donc un certain nombre de pixels N entourant $[i, j]$ et on somme les angles entre les orientations aux pixels adjacents [53].

La figure (Fig.III.10) montre le calcul de l'index de Poincaré avec un voisinage de $N=8$ pixels autour du pixel $[i, j]$ appartenant respectivement (de gauche à droite) à une zone singulière en core, et delta.

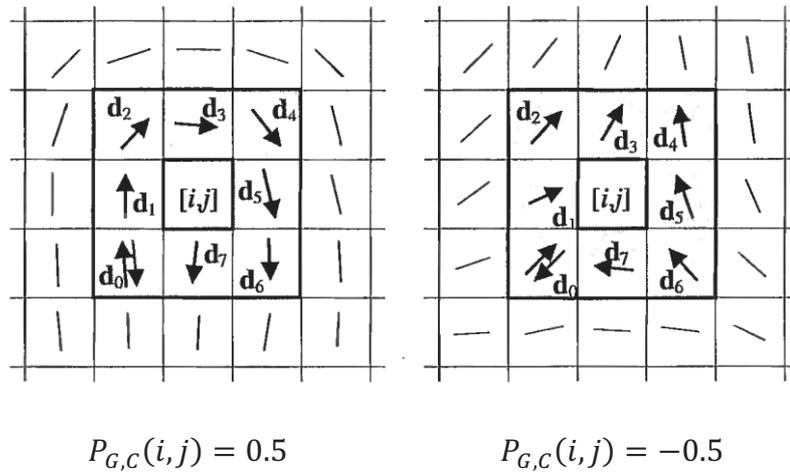


Fig.III.10. L'index de Poincaré calculé avec un voisinage de $N=8$ pixels autour du pixel $[i, j]$ [47].

L'index de Poincaré est alors calculé par la formule suivante :

$$Poincaré(i, j) = \frac{1}{2\pi} \sum_{k=0}^{N-1} \Delta(k) \tag{III.9}$$

$$\text{Où } \Delta(k) = \begin{cases} \delta(k) & |\delta(k)| < \frac{\pi}{2} \\ \delta(k) + \pi & \delta(k) \leq -\frac{\pi}{2} \\ \pi - \delta(k) & \delta(k) \geq \frac{\pi}{2} \end{cases} \tag{III.10}$$

$$\delta(k) = \theta(x_{(k+1) \bmod N}, y_{(k+1) \bmod N}) - \theta(x_k, y_k) \tag{III.11}$$

Et, il va dans le sens inverse des aiguilles d'une montre de 0 à N-1. dans notre cas on a pris la valeur N comme étant égale à 4 [50].

En fait, l'index de Poincaré sur une courbe fermée admet toujours l'une des valeurs suivantes : 0, 0.5 et -0.5. Pour ce qui est des empreintes digitales, le pixel $[i, j]$ se trouve dans une région singulière de core ou bien delta selon la valeur de l'index de Poincaré à ce pixel :

Si Poincaré $(i, j) = 0.5$, le pixel (i, j) se trouve dans une région singulière en core;

Si Poincaré $(i, j) = -0.5$, le pixel (i, j) se trouve dans une région singulière en delta;

Sinon, le pixel (i, j) ne se trouve pas dans une région singulière.

La figure (Fig.III.11) illustre un exemple de détection des points singuliers globaux par l'algorithme de l'index de Poincaré.

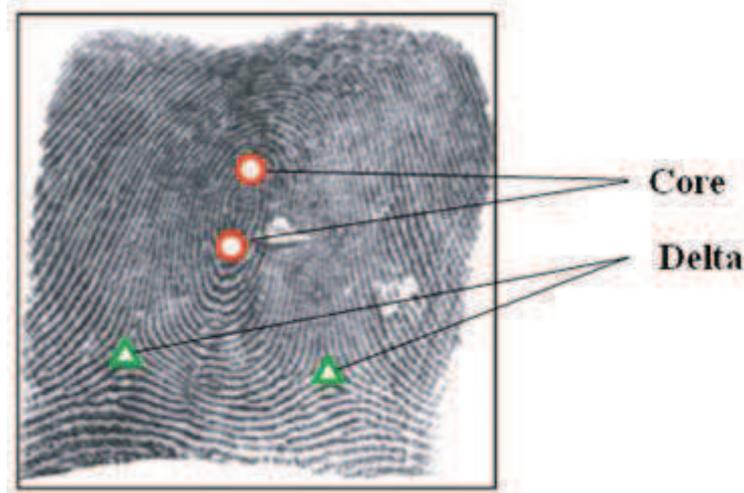


Fig.III.11. Résultat de détection des points singuliers globaux d'une image d'empreinte digitale spire par l'algorithme de l'index de Poincaré.

III.7.4. Classification

C'est la dernière étape d'un système de classification des empreintes digitales, qui consiste à recevoir les caractéristiques extraites de l'image d'empreinte digitale et de prendre une décision qui est la classe à laquelle appartient cette empreinte digitale.

III.8. Approches de classification des empreintes digitales

La plupart des approches existantes de classification des empreintes digitales peuvent être attribué à l'une des catégories suivantes : les approches statistiques, les approches structurales, les approches syntaxiques, les approches neuronales et les approches basées sur les règles.

III.8.1. Approches statistiques

Dans les méthodes statistiques, un vecteur numérique de caractéristiques de taille fixe est dérivé de chacun des empreintes digitales et un classificateur statistique est utilisé pour la classification. L'un des classificateurs statistiques les plus largement adoptés est le plus proche voisin (*k*-nearest neighbor (*k*-NN) en anglais).

De nombreuses approches utilisent directement l'image d'orientation de l'empreinte digitale comme un vecteur de caractéristiques, en codant chaque élément de l'image d'orientation avec les deux composantes $[r \cdot \cos 2\theta, r \cdot \sin 2\theta]$, par exemple une image d'orientation de (30 x 30) donne un vecteur de 1800 éléments, donc il a besoin de grandes quantités de données d'entraînement, de grande capacité de mémoire et de grand temps de calcul. Pour cette raison, les techniques statistiques de réduction de dimensionnalité sont souvent appliquées pour réduire la dimension du vecteur de caractéristiques [47].

III.8.2. Approches structurales

Les approches structurales sont basées sur l'organisation relationnelle des caractéristiques. Cette organisation relationnelle est représentée au moyen de structures de données symboliques, tels que des arbres et des graphiques, qui permettent une organisation hiérarchique de l'information.

L'image d'orientation est bien adaptée pour la représentation structurelle, en fait, il peut être partitionné en régions homogène des orientations ; ces régions et les relations entre eux contiennent des informations utiles pour la classification. C'est l'idée de base de la méthode proposée par Maio et Maltoni (1996), l'image d'orientation est divisée en régions par la minimisation d'une fonction de coût qui tient compte de la variance des orientations élémentaires dans chaque région (Fig.III.12).

Le point faible de cette approche est qu'il n'est pas facile de partitionner l'image d'orientation en régions homogènes, surtout les images d'empreintes de mauvaise qualité.

Le principal avantage de cette approche est qu'il repose uniquement sur l'information structurelle globale, donc il peut travailler sur des images très bruitées, où les points singuliers ou d'autres caractéristiques locales ne peuvent pas être exploités [42].

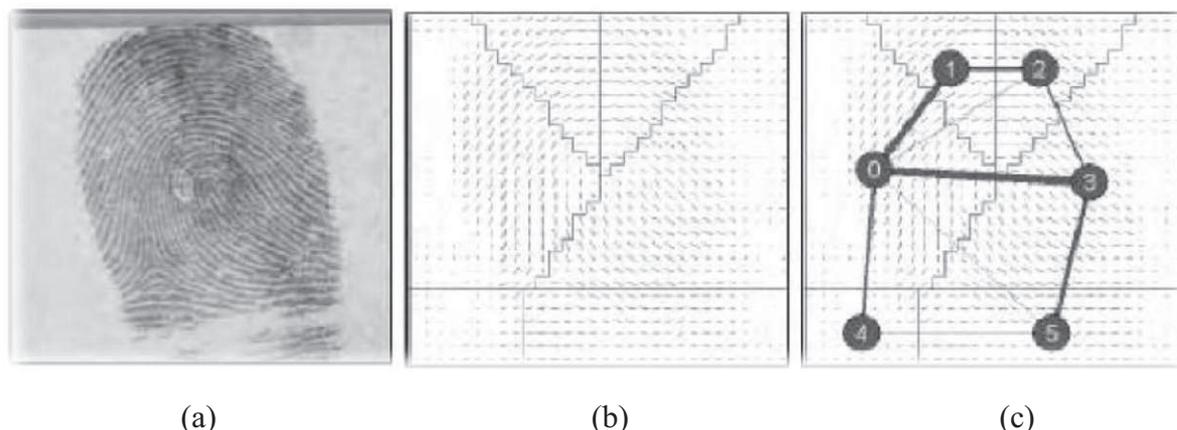


Fig.III.12. (a) image d'empreinte digitale, (b) partitionnement de son image directionnelle, (c) le graphe relationnel correspondant [42].

III.8.3. Approches syntaxiques

Parmi les premières méthodes élaborées, l'idée de base consiste à associer à chaque classe une grammaire qui décrit l'empreinte digitale de sorte que chaque empreinte est codée comme une phrase. Par la suite, une analyse syntaxique est réalisée afin que l'empreinte soit associée à la grammaire qui respecte les règles de l'empreinte digitale [51].

L'approche introduite par Rao et Balck (1980) est basée sur l'analyse des flux de ligne de crête, qui est représenté par un ensemble de lignes connectées (Fig.III.13). Ces lignes sont étiquetées conformément aux changements de direction, obtenant ainsi un ensemble de chaînes de caractères qui sont traitées par des grammaires [47].

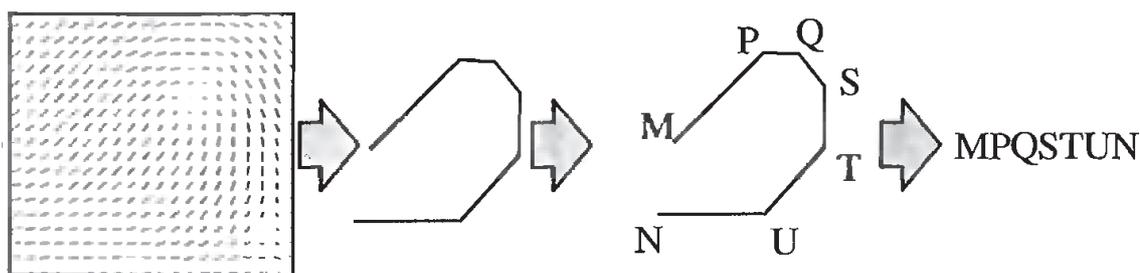


Fig.III.13. Un schéma de l'approche chaîne de la construction de Rao et Balck 1980.

En général, en raison de la grande diversité des types d'empreintes digitales, les approches syntaxiques exigent des grammaires très complexes, pour cette raison, l'utilisation des méthodes syntaxiques pour la classification des empreintes digitales a été presque abandonnée.

III.8.4. Approches des réseaux de neurones

Plusieurs approches de réseaux de neurones ont été proposées dans la littérature pour la classification des empreintes digitales, la plupart sont basées sur des perceptrons multicouches et utilisent les éléments de l'image directionnelle de l'empreinte digitale comme des entrées du réseau. Dans [42], Kamijo présente une architecture pyramidale constituée par plusieurs perceptrons multicouches, dont chacun est formé à reconnaître les empreintes digitales appartenant à des classes différentes. Les réseaux auto-organisés de Kohonen sont aussi utilisés pour la classification, ils sont formés pour trouver les points cores et delta [42].

Dans notre travail on s'intéresse à la classification des empreintes digitales par les Réseaux de Neurones Flous où les entrées de réseaux sont les caractéristiques globales de l'empreinte digitale, et les sorties sont les codes binaires des cinq classes des empreintes digitales.

III.8.5. Approches basées sur les règles

Ces approches sont les plus utilisées par les experts humains pour la classification manuelle, ils sont basés sur les points singuliers en utilisant les règles de type si – alors. Aujourd'hui, cette technique est adoptée pour la classification automatique des empreintes digitales.

Le tableau suivant illustre les classes des empreintes digitales et le nombre de leurs points singuliers :

Classe d'empreinte	Core	Delta	Position du Delta
Arc	0	0	-
Arc tendu	1	1	milieu
boucle à gauche	1	1	droite
boucle à droite	1	1	gauche
Spire	2	2	-

Tab.III.1. Les différentes classes d'empreintes digitales et leurs nombre de points singuliers.

III.9. Etat de l'art de la classification des empreintes digitales

La classification des empreintes digitales est un problème très difficile qui attire l'attention de plusieurs chercheurs dans les dernières années. Pour cette raison plusieurs approches de classification et différentes méthodes d'extraction des caractéristiques ont été proposées, mais aucune parmi ces approches n'est considérée comme complète.

Maheswari et chandra [54] présentent une étude détaillée sur les approches existantes pour la classification des empreintes digitales qui sont appliquées dans le domaine d'identification par les empreintes digitales. Plusieurs approches ont été utilisées pour cet axe de recherche, parmi lesquelles on peut citer : approches statistiques [47], approches structurales [42], approches syntaxiques [51][47], approches basées sur les règles [55][56], approches basées sur les réseaux de neurones [42][58][59], approches basées sur les algorithmes génétiques[57] et approches basées sur les SVM [58][59]. Cependant les approches basées sur les réseaux de neurones ont des meilleurs résultats par rapport aux autres techniques. Les réseaux de neurones utilisant l'algorithme de propagation de gradient pour l'apprentissage donnent des bons résultats, mais ils consomment beaucoup de temps machine.

Kamijo[52] a employé un réseau de neurones pour la classification des images d'empreintes digitales, qui permettent de classer les images d'empreintes digitales complexes. Il utilise une méthode d'apprentissage qui consiste en deux étapes pour former les quatre sous-réseaux de neurones en couches (un sous- réseau pour chaque catégorie). Il effectue l'analyse en composantes principales (ACP) par rapport aux valeurs unitaires de la deuxième couche cachée et étudie l'état de classification des empreintes digitales représenté par l'état interne du réseau également. Par conséquent, le procédé confirme que les modèles d'empreinte digitale sont grossièrement classés dans chaque catégorie dans la seconde couche cachée, et l'efficacité du processus d'apprentissage de deux étapes.

L'approche de classification utilisée par Karu [55] consiste d'abord à trouver la direction de la crête à chaque pixel d'une image d'empreinte digitale donnée. Ensuite, l'algorithme extrait les caractéristiques globales telles que les points singuliers (cores et deltas) dans l'image d'empreinte digitale et effectue le classement basé sur le nombre et l'emplacement des points singuliers détectés. Ici, le ou les point singulier(s) de détection est un processus de régularisation itérative jusqu'à ce que les points singuliers valides sont détectés. Si les images sont de mauvaise qualité, l'algorithme classe les images comme des types inconnus.

Tan [57] utilise un algorithme de apprentissage-caractéristique avec la programmation génétique (GP) pour apprendre et découvrir les opérateurs composites et les caractéristiques qui se dégagent de combinaisons d'opérations primitives de traitement d'image pour la classification des empreintes digitales. Les opérateurs primitifs utilisés sont simples et faciles à calculer. Ces opérateurs ont été séparés en opérateurs de calcul et les opérateurs de génération de caractéristiques. Cette méthode de classification peut être efficace pour les images d'empreintes digitales de bonne qualité.

Park [59] a basé sur la transformée de Fourier discrète utilisée (DFT) et l'analyse non linéaire discriminante pour classer les empreintes digitales. En utilisant la DFT et les filtres directionnels, une image directionnelle fiable et efficace est construite à partir de chaque image d'empreinte digitale, et ensuite l'analyse discriminante non linéaire est appliquée sur les images directionnelles construites, ce qui réduit considérablement la dimension et l'extraction des caractéristiques discriminantes. La méthode explore la capacité de DFT et de filtrage directionnel dans le traitement des images de faible qualité et l'efficacité de la méthode non linéaire d'extraction de caractéristiques dans la classification des empreintes digitales.

Shah [58] a utilisé un vecteur des caractéristiques de dimension petite obtenu à partir de la rétroaction sur la base détecteur de ligne pour classer les empreintes digitales en cinq classes (arc, boucle gauche, boucle de droite, spirale, et l'arc de tentes). Le détecteur de ligne était un système dynamique coopérative qui donne des lignes orientées et conserve de multiples orientations aux points où les lignes orientées différemment répondent. L'extraction de caractéristiques est basée sur la caractérisation de la distribution des orientations autour de l'empreinte digitale. Trois types de classificateurs sont utilisés nommément machines à vecteurs de support, le plus proche voisin classificateur, et réseau de neurones.

Le travail du Mohamed et Nyongesa [46] est un des rares travaux (pour ne pas dire l'unique) réalisés pour la classification des empreintes digitales utilisant les réseaux neuro-floue. Ils ont proposés un système de classification des empreintes digitales qui est basé sur la classification de Henry en utilisant les termes linguistiques et les règles floues. Cette méthode donne des bons résultats de classification avec les réseaux de neurones flous.

On peut constater que :

- Plusieurs méthodes ont été proposées ces deux dernières décennies pour trouver une solution au problème de la classification des empreintes digitales, la plupart des ces travaux son basés sur les arêtes, les caractéristiques locales (minuties) et caractéristiques globales (points singuliers) ;

- Les méthodes basées sur les caractéristiques globales (points singuliers des empreintes digitales) sont plus efficace pour la classification
- Les méthodes qui utilisent l'image d'orientation de l'empreinte sont capables de classer les images en une des cinq classes
- La pluparts des approches classifient les empreintes en cinq classes : arc, arc tendu, boucle à droite, boucle à gauche et spire

Une étude très récente réalisée par Maheswari a permet de dresser la table récapitulative suivante :

Auteur	Technique utilisée	Caractéristiques	Classes
Kamijo (1993)	Neural Network Classifier	PCA, ridge lines, singular points	5C
Karu and Jain (1996)	Rule-based classifier	Cores and Delta	5C et type inconnue
Ballan (1998)	Rule-based classifier	Singular Points from directional histograms	Lasso and Wirbel
Cho <i>et al.</i> (2000)	Rule based classification		4C
Yao <i>et al.</i> (2001)	SVM and RNN	Finger code features and structural representation	5C
Wang <i>et al.</i> (2002)	K-Means and 3-Nearest Neighbour	Directional Fields, delta points and Singular Points	5C
Mohamed and Nvongesa	Fuzzy Neural Network	Singular points, positions and direction of core and delta	5C
Tan <i>et al.</i> (2003)	Genetic Programming and Bayesian Classifier	Image composite operations and features	5C
Shah and Shastry (2004)	SVM, Nearest neighbour and neural network		5C
Afsar <i>et al.</i> , (2004)	Rule based classifier	Singular points	5C
Park and Park (2005)	SVM, MLP, KNN, RNN	Fast Fourier transform coefficient and directional features	5C
Zhou <i>et al.</i> , (2007)	SVM Classifier	Block Features (Mean, variance, contrast, coherence and energy)	Lasso and Wirbel
Ji and Yi (2007)	RNN + SVM	Four dimensional feature vector of directional block classes	5C

Li <i>et al.</i> (2007)	SVM Classifier	coefficients of the orientation model and the singularity information	5C
Wei (2008)	rapid rule-based classification	Singular points using delta field Poincare index	5C
Tarjoman and Zarei (2008)	Relative super graph classifier	Segmented regions directional ridge lines	9 classes
Nain <i>et al.</i> (2008)	Ridge Tracing	Ridges around HRC (High Ridge Curvature) region	4C
Wei <i>et al.</i> (2008)	Rule based classifier	Ridge line type, singular points	5C
Kant and Nath (2009)	Henry Classification with Single Print	Singular Point (Delta)	4C
Suralkar <i>et al.</i> (2009)	Rule Based Classification	Singular point and maximum variation in local orientation Field	5Cb
Bhuyan <i>et al.</i> (2010)	K Means clustering and Apriori Classifier	numeric code sequence based on the ridge flow patterns.	5Ca

Table III.2. Classification des algorithmes de la classification des empreintes digitales [54]

III.10. Conclusion

La classification des empreintes digitales fournit un mécanisme très important pour l'identification automatique des individus par l'empreinte digitale. Elle consiste à affecter une empreinte digitale à une classe d'une manière cohérente et fiable.

Dans ce chapitre, nous avons d'abord présenté l'empreinte digitale, ces caractéristiques et ces classes, ensuite nous avons décrit les différentes étapes d'un système de classification des empreintes digitales qui sont : l'acquisition d'une image d'empreinte digitale, le prétraitement, l'extraction des caractéristiques globales d'empreinte et la classification. Finalement nous avons décrit les différentes approches utilisées pour la classification des empreintes digitales qui sont: les approches statistiques, les approches structurales, les approches syntaxiques, les approches neuronales et les approches basées sur les règles.

IV.1. Introduction

Les réseaux de neurones et la logique floue sont deux concepts totalement inspirés du raisonnement humain. Bien que ces deux modèles soient différents du point de vue structure, ils possèdent de nombreux points communs. En effet, l'utilisation de ces deux techniques ne nécessite pas de modèle mathématique bien défini car la résolution se base sur les valeurs numériques des entrées et des sorties pour les réseaux neuronaux et sur la logique du système dans le cas de la logique floue.

Donc ce chapitre est composé de deux parties, dans la première partie, on introduit les principales notions sur les réseaux de neurones, telle que les méthodes d'apprentissages, l'algorithme de rétro-propagation de gradient de l'erreur et autres notions. Dans la deuxième partie on présente en premier lieu, la logique floue, les ensembles flous et les fonctions d'appartenance, on passe ensuite aux réseaux de neurones flous et les différentes combinaisons des réseaux de neurones avec la logique floue, et en fin les types des réseaux de neurones flous les plus utilisées pour la classification.

IV.2. Notions fondamentales sur les réseaux de neurones

IV.2.1. Introduction

Les réseaux de neurones artificiels (RNA) sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Ils infèrent des propriétés émergentes permettant de résoudre les problèmes complexes.

IV.2.2. Neurone biologique

Le neurone biologique est le composant cellulaire de base du cerveau (voir Fig. IV.1), il comprend :

- Le corps cellulaire, qui fait la somme des influx par intermédiaire de l'axone ;
- L'axone qui permet de transmettre les signaux émis par le corps cellulaire aux autres neurones ;
- Les dendrites, qui sont les récepteurs principaux du neurone, captant les signaux qui lui parviennent ;
- Les synapses qui permettent aux neurones de communiquer avec les autres via les axones et les dendrites.

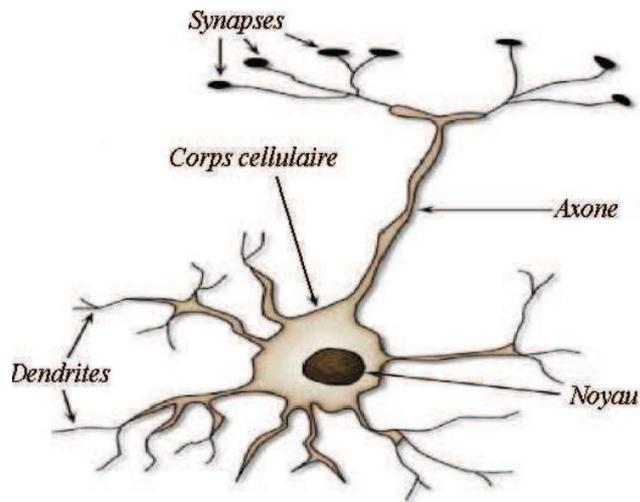


Fig. IV.1. Neurone biologique.

IV.2.3. Neurone artificiel

Le neurone artificiel est un modèle mathématique simplifié du neurone biologique, il présente un certain nombre d'entrées, un corps traitant les entrées, et un axone véhiculant la réponse du neurone.

La Figure (Fig. IV.2) montre la structure d'un neurone artificiel. Chaque neurone reçoit un nombre variable d'entrées en provenance des neurones amont. A chacune de ces entrées est associé un poids W_i abréviation de Weight (poids en Anglais) représentatif de l'entrée X_0 .

Chaque neurone élémentaire est doté d'une fonction de transfert (fonction d'activation) qui donne une sortie unique Y , et se ramifie ensuite pour alimenter un nombre variable de neurones avals [60].

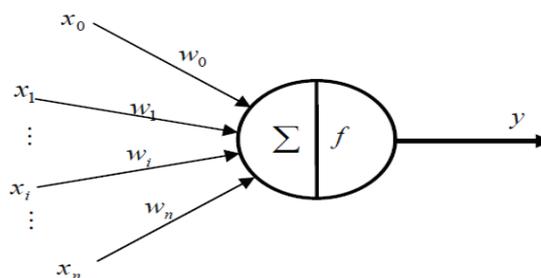


Fig. IV.2. Neurone artificiel [61].

$$z = \sum_{i=0}^n w_i x_i \text{ et } y = f(z) \quad (IV.1)$$

Les fonctions d'activations les plus utilisées sont les suivantes : (voir Fig. IV.3)

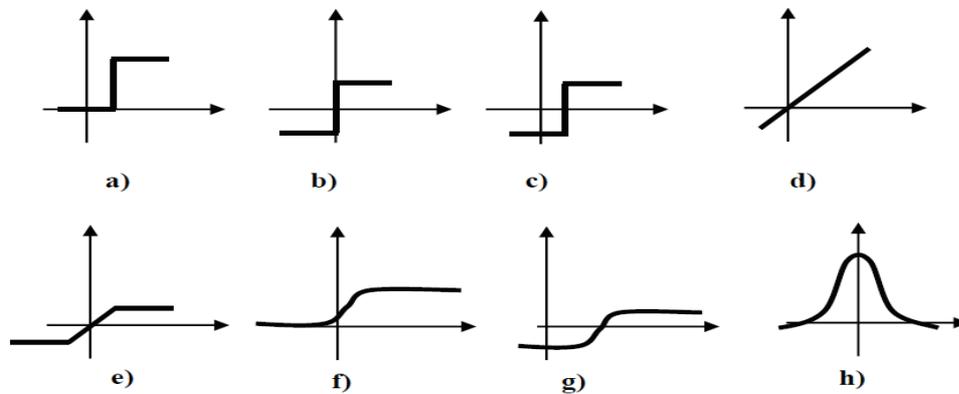


Fig. IV.3. fonctions d'activations les plus utilisées. a) tout ou rien ; b) fonction signe ; c) plus ou moins à seuil ; d) fonction affine ; e) saturation ; f) sigmoïde ; g) fonction arc tangente ; h) fonction gaussienne. [61].

IV.2.4. L'apprentissage

L'apprentissage dans le contexte des réseaux de neurones, est le processus de modification des poids de connexions (y compris les biais) ou plus rarement du nombre de couches et de neurones, afin d'adapter le traitement effectué par le réseau à une tâche particulière [62].

IV.2.4.1. Types d'apprentissages

On distingue trois familles d'apprentissage :

a. Apprentissage supervisé

Un superviseur, ou professeur, fournit au réseau des couples d'entrées-sorties. Il fait apprendre au réseau l'ensemble de ces couples, par une méthode d'apprentissage, comme la rétro-propagation du gradient de l'erreur, en comparant pour chacun d'entre eux la sortie effective du réseau et la sortie désirée. L'apprentissage est terminé lorsque tous les couples entrées-sorties sont reconnus par le réseau.

b. Apprentissage non supervisé

Cet apprentissage consiste à détecter automatiquement des régularités qui figurent dans les exemples présentés et à modifier les poids des connexions pour que les exemples ayant les mêmes caractéristiques de régularité provoquent la même sortie. Les réseaux auto-organiseurs de Kohonen [63] sont les réseaux à apprentissage non supervisé les plus connus.

c. Apprentissage auto-supervisé

Le réseau de neurones évalue lui-même ses performances, sans l'aide d'un « expert », un objet est présenté à l'entrée du réseau de neurones, à qui on a indiqué la classe à laquelle appartient cet objet. Si le réseau ne le classe pas correctement, il mesure lui-même l'erreur, et propage cette erreur vers l'entrée. Le réseau procède à autant d'itérations qu'il est nécessaire jusqu'à obtenir la réponse valide.

IV.2.4.2. Les méthodes d'apprentissage [63]

Dans les systèmes experts, les connaissances de l'expert ont une forme énumérée : elles sont exprimées sous forme de règles. Dans le cas des réseaux de neurones, les connaissances ont une forme distribuée : elles sont codées dans les poids des connexions, la topologie du réseau, les fonctions de transfert de chaque neurone, le seuil de ces fonctions et la méthode d'apprentissage utilisée. Il existe un certain nombre de méthodes d'apprentissage :

1. Règle d'apprentissage de Hebb

C'est la méthode d'apprentissage la plus ancienne (1949), elle est inspirée de la biologie. Elle traduit le renforcement des connexions liant deux neurones activés. Si un des deux neurones au moins n'est pas activé, le poids de la connexion n'est pas modifié.

L'apprentissage se fait par les équations suivantes :

$$w_{ij}^{t+1} = w_{ij}^t + O_i \cdot O_j \quad \text{ou} \quad w_{ij}^{t+1} = w_{ij}^t + (O_i^t - O_i^{t-1}) \cdot (O_j^t - O_j^{t-1})$$

2. Apprentissage de Rosenblatt

L'apprentissage du Perceptron de F. Rosenblatt (1958) est un apprentissage supervisé qui se fait par correction d'erreur. Il consiste à présenter au réseau une série d'exemples à apprendre, puis à minimiser l'erreur entre la sortie désirée S et la sortie effectivement obtenue Y .

L'algorithme est le suivant :

- Présenter au réseau un exemple (E, S) ;
- Calculer la sortie obtenue : $Y_j = \sum_i e_i \cdot w_{ij}$;
- Pour chaque neurone, calculer : $w_{ij}^{t+1} = w_{ij}^t + \mu \cdot (S_j - Y_j)$;

Avec

w_{ij} : Poids de la connexion entre le neurone i et le neurone j .

S_j : Sortie désirée pour le neurone j .

Y_j : Sortie obtenue pour le neurone j .

μ : Coefficient représentant la vitesse d'apprentissage.

t : Temps.

3. Apprentissage de Widrow-Hoff

Cet apprentissage est une extension du modèle du Rosenblatt à des sorties binaires à des sortie réelles. L'algorithme est le suivant :

- Affecter des valeurs aléatoires aux connexions du réseau;
- Présenter un vecteur d'entrée $X(x_1, \dots, x_n)$;
- Calculer les sorties: $S_j = \sum_i x_i \cdot w_{ij}$;
- Calculer l'erreur de chaque neurone: $\varepsilon_j = (Y_j - S_j)$;
- Modifier le poids des neurones: $w_{ij}^{t+1} = w_{ij}^t + \mu \cdot x_i \cdot \varepsilon_j$;

Avec

w_{ij} : Poids de la connexion entre le neurone i et le neurone j .

S_j : Sortie désirée pour le neurone j .

Y_j : Sortie obtenue pour le neurone j .

μ : Coefficient représentant la vitesse d'apprentissage.

t : Temps.

4. Rétro-propagation du gradient de l'erreur

L'apprentissage par rétro-propagation est une généralisation de la règle de Widrow-Hoff, il consiste à présenter au réseau un vecteur d'entrées, de procéder au calcul de la sortie par propagation à travers les couches, de la couche d'entrées vers la couche de sortie passant par les couches cachées. Cette sortie obtenue est comparée à la sortie désirée, une erreur est alors obtenue. A partir de cette erreur, est calculé le gradient de l'erreur qui est à son tour propagé de la couche de sortie vers la couche d'entrée, d'où le terme de rétro-propagation. Cela permet la modification des poids du réseau et donc l'apprentissage. L'opération est répétée pour chaque vecteur d'entrée et cela jusqu'à ce que le critère d'arrêt soit vérifié.

IV.2.4.3. Les réseaux de neurones les plus utilisés

a. Perceptron simple

Le perceptron est le premier modèle de réseau de neurones. Le but du perceptron est d'associer des formes en entrée à des réponses (Fig.IV.4.). Le perceptron se compose de deux couches : la couche d'entrée et la couche de sortie qui donne la réponse correspondant à la stimulation présente en entrée. Les cellules de la première couche répondent en oui/non. La réponse «oui» correspond à une valeur «1» et la réponse « non» correspond à une valeur «0» à la sortie du neurone.

Les cellules d'entrée sont reliées aux cellules de sortie grâce à des synapses d'intensité variable. L'apprentissage du perceptron s'effectue en modifiant l'intensité de ces

synapses. Les cellules de sortie évaluent l'intensité de la stimulation en provenance des cellules de la première couche en effectuant la somme des intensités des cellules actives.

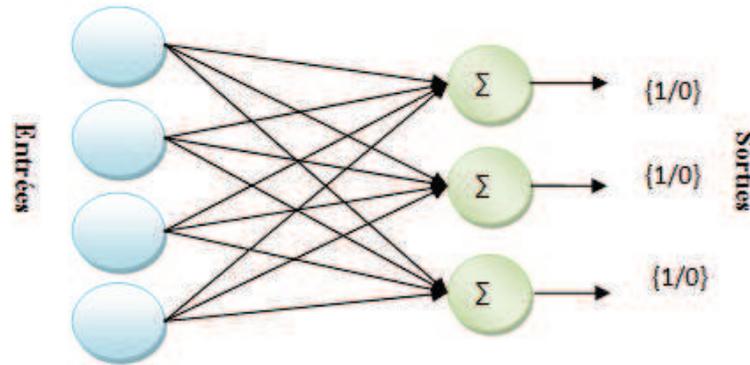


Fig.IV.4. Schéma général de perceptron simple.

b. Perceptron Multicouches (PMC)

Le perceptron Multicouche est un réseau orienté de neurones artificiels organisé en couches et où l'information voyage dans un seul sens, de la couche d'entrée vers la couche de sortie.

La figure (Fig. IV.5) donne l'exemple d'un réseau contenant une couche d'entrée, deux couches cachées et une couche de sortie. La couche d'entrée représente toujours une couche virtuelle associée aux entrées du système. Elle ne contient aucun neurone. Les couches suivantes sont des couches de neurones. Dans l'exemple illustre, il y a 3 entrées, 4 neurones sur la première couche cachée, trois neurones sur la deuxième couche cachée et quatre neurones sur la couche de sortie. Les sorties des neurones de la dernière couche correspondent toujours aux sorties du système.

Dans le cas général, un perceptron Multicouche peut posséder un nombre de couches quelconque et un nombre de neurones (ou d'entrées) par couche également quelconque.

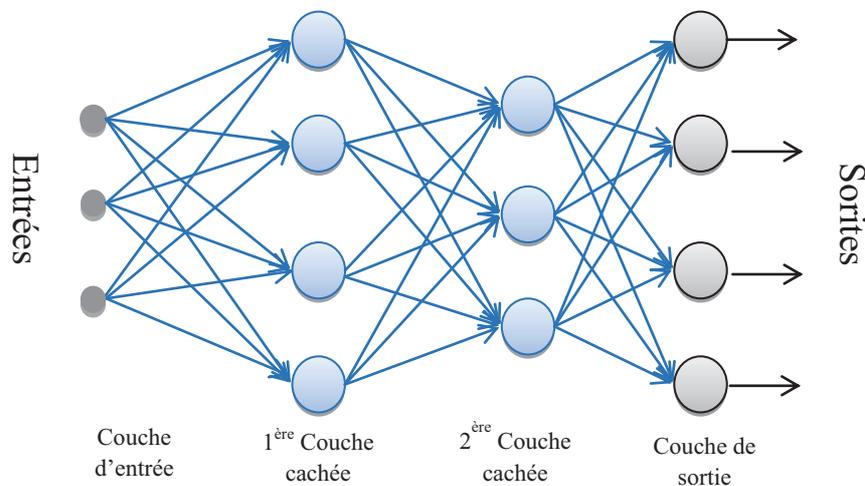


Fig.IV.5. Exemple de réseau de type perceptron Multicouche.

c. Les réseaux de neurones à Fonctions de Base Radiales (RBF)

Les réseaux à fonctions de base radiales (RBF) ou plus simplement réseaux à bases radiales ont été proposés par J. Moody et C. Darken. Ils comportent une couche d'entrée, une couche cachée et une couche de sortie.

Pour un réseau comportant n entrées et m unités cachées, l'activation des neurones cachés est donnée par une fonction de type gaussienne. Chacun de ces neurones ne s'active donc de manière significative que pour des valeurs d'entrée relativement proches des centres des gaussiennes. Les connexions provenant des neurones d'entrée ne sont pas pondérées.

L'apprentissage se fait dans ces réseaux par modification des poids des connexions entre les neurones cachés, les neurones de sortie, les centres et les variances des gaussiennes.

d. Les réseaux de neurones bouclés (récurrents)

Un réseau bouclé ou récurrent possède la même structure qu'un réseau multicouche munie de rétroactions (voir Fig.IV.6), les connexions rétroactives peuvent exister entre tous les neurones du réseau sans distinction, ou seulement entre certains neurones (les neurones de la couche de sortie et les neurones de la couche d'entrée ou les neurones de la même couche par exemple).

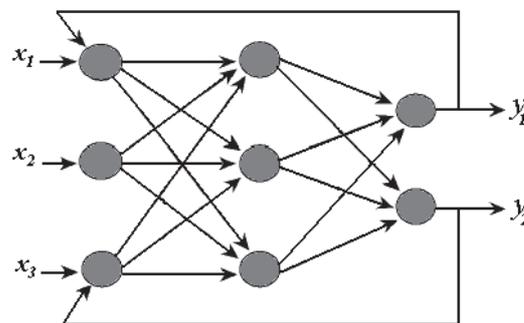


Fig.IV.6. Réseaux de neurones bouclés.

IV.2.5. La rétro-propagation du gradient de l'erreur

L'algorithme de rétro-propagation du gradient de l'erreur a été créé en généralisant les règles d'apprentissages de Widrow-Hoff [63], aux réseaux multicouches à fonction de transfert non linéaire. C'est un algorithme utilisé avec des réseaux de types feedforward pour l'apprentissage de fonction, la classification et la reconnaissance de forme.

IV.2.5.1. Principe

La rétro-propagation du gradient de l'erreur est utilisée pour ajuster les poids et les biais du réseau afin de minimiser l'erreur quadratique entre la sortie du réseau et la sortie désirée. A chaque couple entrée/sortie, une erreur est calculée. Ensuite les poids et les biais

sont modifiés en ligne sur le réseau. On réitéré ces calculs jusqu'à l'obtention du critère d'arrêt.

IV.2.5.2. Algorithme

L'algorithme du rétro-propagation du gradient de l'erreur se résume aux étapes suivantes :

1. Initialisation des poids $W^{[q]}$ à des petites valeurs aléatoires.
2. Présentation d'une entrée x_k et de la sortie désirée d_k .
3. Calcul de la sortie actuelle par propagation à travers les couches :

$$y_j^{[q]} = F(P_i) \quad (IV.2)$$

On a
$$P_i = \sum_i W_{ji}^{[q]} \cdot x_i^{[q-1]}$$

Où F est la fonction de transfert et $[q]$ la $q^{ième}$ couche du réseau.

4. Accumulation des erreurs en sortie :

$$\varepsilon = 0,5 \cdot \sum_k \left(d_k - y_k^{[s]} \right)^2 \quad (IV.3)$$

- d_k est la sortie désirée associée au vecteur d'entrée x_k .
- $y_k^{[s]}$ est la sortie obtenue sur la dernière couche au temps t .
- ε est l'erreur cumulée pour k présentations de couples (x_k, d_k)

5. Rétro-propagation du gradient de l'erreur (δ) depuis la dernière couche vers la première couche :

- Pour chaque cellule de sortie :

$$\delta_i^{[s]} = \left(d_i - y_i^{[s]} \right) \cdot F' \left(p_i^{[s]} \right) \quad (IV.4)$$

Où F' est la dérivée de la fonction de transfert.

- Pour chaque cellule cachée :

$$\delta_i^{[q]} = \sum_k \delta_k^{[q+1]} \cdot W_{ki} \cdot F' \left(p_i^{[s]} \right) \quad (IV.5)$$

6. Mise à jour des poids et des biais selon les règles :

$$\Delta W_{ij}^{[q]} = \alpha \cdot \left(\delta_i^{[q]} \cdot x_j^{[q]} \right) \quad (IV.6)$$

$$\Delta b_i^{[q]} = \alpha \cdot \delta_i^{[q]} \quad (IV.7)$$

Où α est le coefficient d'apprentissage compris dans l'intervalle $[0,1]$ et b est le biais.

7. Retour à 2 tant qu'il y a des couples à présenter.

IV.2.5.3. Choix de critère à minimiser

Dans le cas de la rétro-propagation de l'erreur, le critère à minimiser est une erreur quadratique. L'application de l'algorithme du gradient nécessite la dérivabilité de la fonction de transfert.

Le critère de minimisation d'erreur est le suivant :

$$\varepsilon = \sum_k \left(d_k - y_k^{[s]} \right)^2 \quad (IV.8)$$

L'algorithme présenté ici est de type «online », c'est-à-dire que l'on met à jour les poids pour chaque échantillon d'apprentissage présenté dans le réseau de neurones. Une autre méthode est dite en « batch », c'est-à-dire que l'on calcule d'abord les erreurs pour tous les échantillons sans mettre à jour les poids (on additionne les erreurs) et lorsque l'ensemble des données est passé une fois dans le réseau, on applique la rétro-propagation en utilisant l'erreur totale. Cette façon de faire est préférée pour des raisons de rapidité et de convergence.

IV.2.5.4. Conditions d'arrêt de l'apprentissage [64]

Le processus d'apprentissage du réseau peut être interrompu pour plusieurs raisons dont les plus importantes sont :

- Le dépassement du seuil fixé sur la moyenne des carrés d'erreurs entre la sortie du réseau et les données d'apprentissage. Dans ce cas, la précision du réseau est satisfaisante et le réseau est prêt à remplacer le système à modéliser.
- Le dépassement du seuil fixé sur le module du gradient. Quand le module du gradient de la fonction de performance est inférieur à une valeur fixée à l'avance, l'évolution des poids et biais devient négligeable, alors le processus d'apprentissage est interrompu et le réseau retient les poids et biais ajustés lors de la dernière époque.
- Le dépassement du seuil fixé sur le nombre d'époques (itérations). Par définition, une époque est le calcul des résultats du réseau pour toutes les entrées utilisées dans la base d'apprentissage, le calcul de la fonction de performance et des composants de son gradient et l'ajustement des poids et biais du réseau en se basant sur l'information fournie par le gradient. Donc le nombre d'époques correspond au nombre de fois que les poids et biais sont ajustés. Quand le nombre d'époques atteint une valeur fixée à l'avance, le processus d'apprentissage est interrompu et le réseau retient les poids et biais ajustés lors de la dernière époque.

IV.2.5.5. Avantages et inconvénients

Comme tout algorithme, l'algorithme de rétro-propagation de gradient possède des avantages et des inconvénients :

➤ Avantages

- Ce fût un des premiers algorithmes développés pour l'apprentissage des réseaux de neurones multicouches. Il permet de couvrir un inconvénient de l'algorithme du perceptron qui est incapable de modifier les poids des couches cachées;
- l'implémentation informatique ne présente pas de difficultés.

➤ Inconvénients

- l'algorithme de rétro-propagation du gradient de l'erreur suit la descente du gradient de l'erreur : un minimum local peut rapidement bloquer la recherche des optimaux globaux ;
- l'algorithme de rétro-propagation consomme beaucoup de temps machine;
- importance du choix du coefficient d'apprentissage, si le coefficient est trop grand la dynamique du réseau va osciller autour de l'optimum, s'il est trop petit, la convergence est lente.

IV.2.6. Avantages et Inconvénients des réseaux de neurones

IV.2.6.1. Avantage des réseaux de neurones

- Capacité de représenter n'importe quelle fonction, linéaire ou pas, simple ou complexe ;
- Faculté d'apprentissage à partir d'exemples représentatifs, par " rétro-propagation des erreurs ". L'apprentissage (ou construction du modèle) est automatique ;
- Résistance au bruit ou au manque de fiabilité des données ;
- Simple à manier, beaucoup moins de travail personnel à fournir que dans l'analyse statistique classique.
- Comportement moins mauvais en cas de faible quantité de données ;
- Pour l'utilisateur novice, l'idée d'apprentissage est plus simple à comprendre que les complexités des statistiques multi-variables.

IV.2.6.2. Inconvénients des réseaux de neurones

- L'absence de méthode systématique permettant de définir la meilleure topologie du réseau et le nombre de neurones à placer dans la (ou les) couche(s) cachée(s) ;
- Le choix des valeurs initiales des poids du réseau et le réglage du pas d'apprentissage, qui jouent un rôle important dans la vitesse de convergence ;

- Le problème du sur-apprentissage (apprentissage au détriment de la généralisation) ;
- La connaissance acquise par un réseau de neurone est codée par les valeurs des poids synaptiques, les réseaux de neurones sont donc des *boites noires* où les connaissances sont inintelligibles pour l'utilisateur.

IV.3. Les Réseaux de Neurones Flous

IV.3.1. La logique floue

La notion de logique floue permet d'étendre la notion de logique classique, associée aux variables booléennes ne prenant que deux valeurs 0 et 1. Il est alors possible d'associer à des variables des coefficients d'appartenance à des sous-ensembles flous prenant des valeurs dans l'intervalle $[0, 1]$ et quantifiant l'incertitude sur la variable. La logique floue a été développée et formalisée par Lofti Zadeh en 1965.

Un événement certain pour la variable se traduira par un coefficient d'appartenance au sous-ensemble flou, c'est-à-dire à la propriété, égal à 1 alors que la valeur sera inférieure à 1 en présence d'incertitudes. Il vient alors qu'une valeur nulle pour un coefficient d'appartenance indique que la possibilité d'appartenance au sous - ensemble sélectionné de la variable représentant la grandeur concernée est complètement rejetée [65].

IV.3.1.1. Notion de sous-ensemble flou

Dans la théorie classique des ensembles, un sous-ensemble A de B est défini par une fonction d'appartenance $\mu_A(x)$ qui caractérise tout élément x appartenant à B . Cette fonction prend la valeur 1 si x appartient à A et la valeur 0 dans le cas contraire :

$$\mu_A(x) = \begin{cases} 1 & \text{Si } x \in A \\ 0 & \text{Si } x \notin A \end{cases} \quad (\text{IV. 9})$$

En logique floue, un sous-ensemble flou A de B est défini par une fonction d'appartenance $\mu_A(x)$ pouvant prendre différentes valeurs comprises entre 0 et 1, selon le degré d'appartenance de l'élément x au sous-ensemble A .

$$\mu_A(x) \in [0, 1]$$

IV.3.1.2. Variable et valeur de variable linguistique

La variable linguistique représente un phénomène ou une grandeur, par des termes linguistique : « quelque, beaucoup, souvent », « négative, environ zéro et positive » ce sont les valeurs linguistique.

La figure (Fig.IV.7) illustre un exemple de la variable linguistique 'vitesse' avec trois termes linguistiques: petite, moyenne et grande.

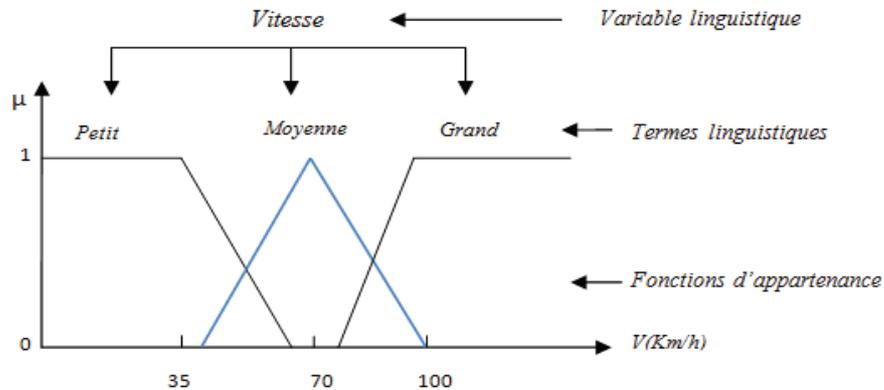


Fig.IV.7. Variable linguistique.

IV.3.1.3. Les fonctions d'appartenance [66]

On peut utiliser pour les fonctions d'appartenance des formes différentes (Fig.IV.8), le plus souvent, des formes triangulaires ou trapézoïdales, il s'agit des formes les plus simples composées par segments de droites. La forme rectangulaire est utilisée pour représenter la logique classique. Dans la plupart des cas, en particulier pour le réglage par logique floue, ces deux formes sont suffisantes pour déterminer des ensembles flous.

a. Fonction triangulaire

Elle est définie par trois paramètres $\{a, b, c\}$, qui déterminent les coordonnées des trois sommets.

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad (IV.10)$$

b. Fonction trapézoïdale

Elle est définie par quatre paramètres $\{a, b, c, d\}$, qui déterminent les coordonnées des quatre sommets.

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad (IV.11)$$

c. Fonction sigmoïde

Elle est définie par deux paramètres $\{a, b\}$:

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (IV.12)$$

d. Fonction gaussienne

Elle est définie par deux paramètres $\{\sigma, m\}$,

$$\mu(x) = \exp\left(-\frac{(x - m)^2}{2\sigma^2}\right) \quad (IV.13)$$

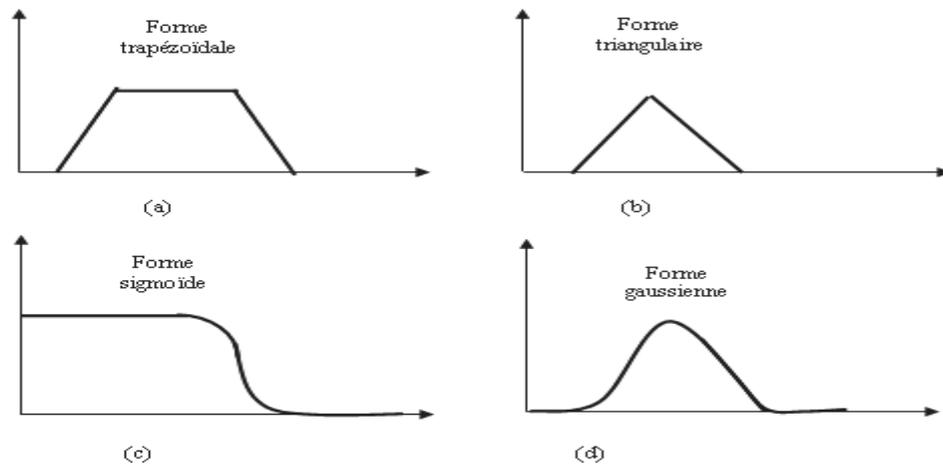


Fig.IV.8. Formes des fonctions d'appartenance usuelles [66].

IV.3.1.4. Règles d'inférence [67]

On appelle règles d'inférence l'ensemble des différentes règles reliant les variables floues d'entrée d'un système aux variables floues de sorties de ce système à l'aide des différents opérateurs. Ces règles se présentent sous la forme :

Si condition 1 ET/OU condition 2 (ET/OU...) alors action sur les sorties OU

Si condition 3 ET/OU condition 4 (ET/OU...) alors action sur les sorties OU

⋮

Si condition n ET/OU condition n+1 (ET/OU...) alors action sur les sorties OU

IV.3.1.5. Système d'inférence floue

La notion de règle floue permet de définir un système expert flou comme une extension d'un système expert classique, manipulant la proposition floue. Donc un système d'inférence floue (SIF) est formé de trois blocs comme l'indique la figure (Fig.IV.9).

Le premier, bloc de fuzzification transforme les valeurs numériques en degrés d'appartenance aux différents ensembles flous de la partition. Le second bloc est le moteur d'inférence, constitué de l'ensemble de règles. Enfin, le bloc de défuzzification permet, si nécessaire, d'inférer une valeur nette, à partir du résultat de l'agrégation des règles [70].

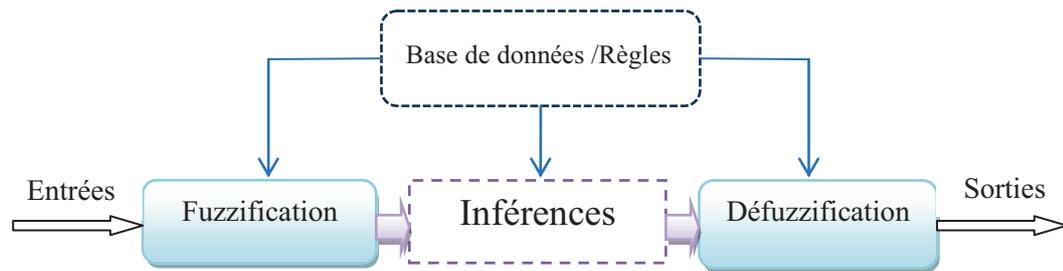


Fig.IV.9. Structure d'un système d'inférence floue.

- **Fuzzification**

La fuzzification consiste à calculer, pour chaque valeur d'entrée numérique, les degrés d'appartenance aux ensembles flous associés et prédéfinis dans la base de données du système flou. Ce bloc réalise la transformation des entrées numériques en informations symboliques floues utilisables par le mécanisme d'inférence [62].

- **Inférence floue**

La phase d'inférence consiste à calculer le degré de vérité des différentes règles du système, en utilisant les formules données dans la phase de fuzzification, et à associer à chaque règle une valeur de sortie. Cette valeur de sortie dépend de la partie conséquente des règles qui peut prendre plusieurs formes. Donc, on peut dire que l'inférence floue est l'opération d'agrégation des règles floues [62].

- **Défuzzification**

En sortie de l'inférence, le système flou ne peut pas communiquer des valeurs floues qu'il peut seul exploiter. Il lui est donc nécessaire de fournir des valeurs précises, c'est le rôle de la défuzzification.

Cette étape s'effectue toujours à l'aide des fonctions d'appartenance. A partir des degrés d'appartenance, on obtient autant de valeurs qu'il y a d'états [68].

IV.3.2. Systèmes Neuro-Flous

Les réseaux de neurones (RN) multicouches constituent des approximateurs universels grâce à leurs capacités d'apprentissage, mais leurs structures et leurs paramètres n'ont pas toujours d'interprétation physique. De plus, la connaissance humaine ne peut pas être exploitée pour les construire.

En revanche, les systèmes d'inférence flous possèdent deux avantages par rapport aux réseaux de neurones. D'une part, ils sont généralement construits à partir de la

connaissance humaine, d'autre part, ils ont une capacité descriptive élevée due à l'utilisation de variables linguistiques. L'approche traditionnelle pour la conception des systèmes d'inférences flous est basée sur les connaissances acquises par des opérateurs experts. Cette approche a prouvé son efficacité dans plusieurs applications. Il se peut cependant que les opérateurs ne puissent pas traduire leurs connaissances et expérience sous forme de contrôleur à logique floue. Il arrive parfois que le domaine d'expertise ne soit pas disponible. Il est donc naturel de construire des systèmes hybrides qui combinent les concepts des systèmes d'inférence flous et des réseaux de neurones, ce sont des réseaux neuro-flous, dont l'apprentissage permet d'ajuster les paramètres et de s'adapter aux changements de conditions.

IV.3.2.1. Définition

Un système Neuro-Flou est un réseau de neurones qui est typologiquement équivalent à la structure d'un système flou. Les entrées/sorties du réseau ainsi que les poids sont des nombres réels, mais les nœuds implémentent des opérations spécifiques aux systèmes flous : fuzzification, opérateurs flous (conjonction, disjonction) et défuzzification. En d'autres termes, un système Neuro-flou peut être vu comme un système flou pour lequel les opérations sont implémentées de façon parallèle par un réseau de neurone [68].

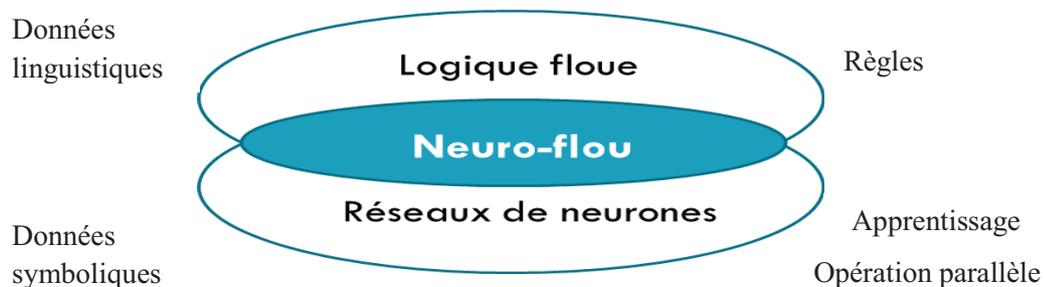


Fig.IV.10. *principe du système neuro-flou.*

La figure (Fig.IV.10) résume le principe du système neuro-flou qui représente l'intersection entre la logique floue et les réseaux de neurones.

IV.3.2.2. Le perceptron flou

L'architecture du perceptron flou est identique à celle du perceptron Multi-Couche usuel, mais les poids sont modélisés par des ensembles flous. Les activations, les sorties, et les fonctions de propagation seront changées. Aussi, l'intention de ce modèle est qu'il soit

interprétable par des règles linguistiques et peut utiliser des bases de connaissance des règles à priori, donc l'apprentissage peut ne pas démarrer à zéro (la base de règles n'est pas vide) 70].

IV.3.3. Les combinaisons des réseaux de neurones avec la logique floue

Il existe quatre grandes catégories de combinaisons des réseaux de neurones avec la logique floue 70].

IV.3.3.1. Réseau flou neuronal

Dans ces réseaux, les techniques floues sont employées pour augmenter les possibilités du processus d'apprentissage et de la réalisation des réseaux de neurones.

IV.3.3.2. Système neuronal/flou simultanément

Le réseau de neurone et le système flou fonctionnent ensemble sur la même tâche, mais sans s'influencer, c'est employé pour déterminer les paramètres de l'autre. Habituellement, le réseau neuronal traite les entrées, ou post-traite les sorties du système flou.

IV.3.3.3. Modèles neuro-flous coopératifs

Le réseau de neurone est employé pour déterminer les paramètres (les règles et les ensembles flous) d'un système flou. Après la phase d'apprentissage. Le système flou fonctionne sans le réseau de neurone. C'est une forme simple des systèmes neuro-flous.

IV.3.3.4. Modèles neuro-flous hybrides

Les approches Neuro-Floues modernes sont de cette forme. Un réseau neuronal et un système flou sont combinés dans une architecture homogène. Le système peut être interprété comme un réseau neuronal spécial avec des paramètres flous ou comme un système flou mis en application sous une forme distribuée parallèle.

IV.3.4. Les types des réseaux de neurones flous utilisés pour la classification

Plusieurs architectures des réseaux flous ont été proposées dans la littérature. Parmi ces architectures, on peut citer : le réseau de Pedrycz, le réseau fusionné compact, le réseau NEFClass :

IV.3.4.1. La structure de Pedrycz

C'est un réseau de neurones flous de trois couches, il comporte une partie de fuzzification des entrées suivie du réseau neuronal qui détermine les décisions à prendre (Fig.IV.11).

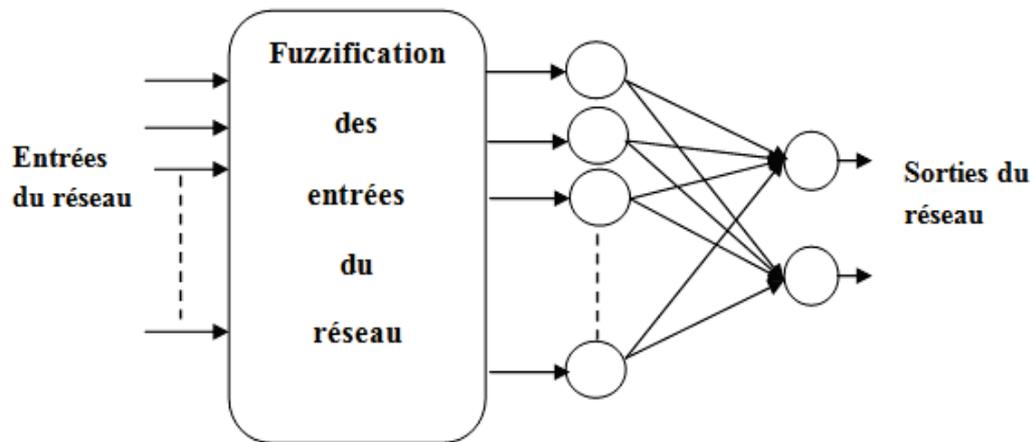


Fig.IV.11. Architecture de réseau de Pedrycz.

L'apprentissage de ce réseau implique le changement des relations floues entre les synapses d'entrées et la sortie en agissant sur les poids des entrées. L'adaptation suit en général, la méthode de rétro-propagation de gradient.

IV.3.4.2. La structure fusionnée compacte

Cette structure est un réseau de neurones de quatre couches. Chacune correspond à une étape du raisonnement flou (Fig.IV.12).

Les neurones de la première couche correspondent aux entrées du système, ceux de la deuxième couche correspondent aux variable linguistiques, les neurones de la troisième couche correspondent aux règles flous alors que les neurones de la dernière couche correspondent aux sorties du système.

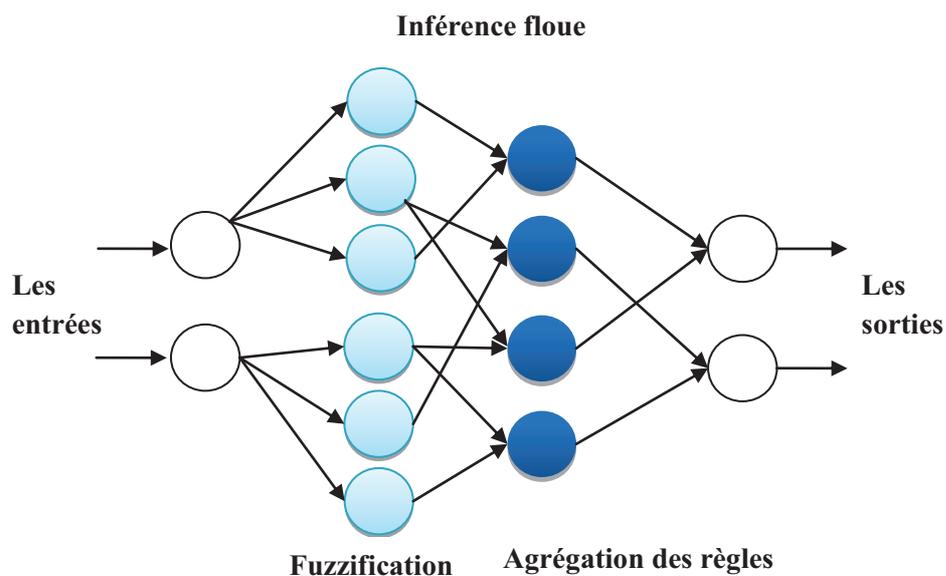


Fig.IV.12. Architecteur du réseau fusionné compact.

IV.3.4.3. Les systèmes neuro-flous de classifications NEFClass

Les systèmes NEFClass (**NE**uro **F**uzzy **CL**ASSification) sont souvent appelés les réseaux perceptrons flous de trois couches (three layers fuzzy perceptron networks). Les neurones de la couche d'entrées représentent les caractéristiques des entrées, ceux de la couche cachée représentent les règles d'inférences alors que les neurones de la couche de sortie donnent les différentes classes. Dans un système NEFClass la connexion entre la couche d'entrée et la couche cachée se fait par le biais des poids flous alors que la couche cachée et la couche de sortie sont reliées par des poids booléens (Fig.IV.13) [70].

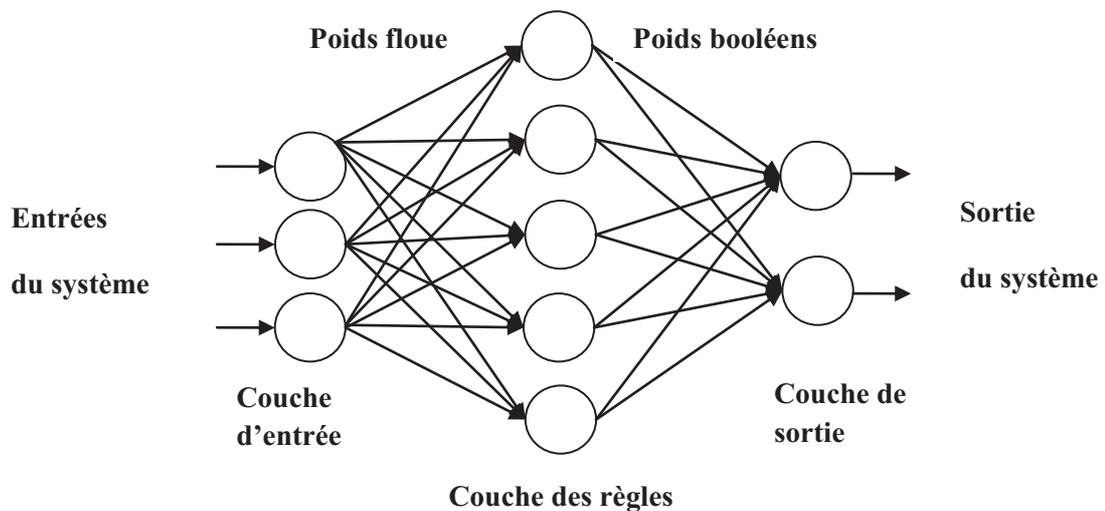


Fig.IV.13. Architecteur d'un système NEFClass.

IV.4. Conclusion

Les réseaux de neurones flous sont apparus de l'association des réseaux de neurone avec la logique floue, de manière à tirer profit des avantages de chacune de ces deux techniques. La principale propriété des réseaux de neurones flous est leur capacité à traiter dans un même outil des connaissances numériques et symboliques d'un système. Ils permettent donc d'exploiter les capacités d'apprentissage des réseaux de neurones d'une part et les capacités de raisonnement de la logique floue d'autre part.

Dans ce chapitre nous introduisons en premier lieu les réseaux de neurones artificiels, ensuite on décrit la logique floue, les sous-ensembles flous, les variables linguistiques et les fonctions d'appartenance. En fin nous présentons les réseaux de neurones flous, leurs modèles et les types les plus utilisées pour la classification.

V.1. Introduction

L'utilisation de l'empreinte digitale comme moyen d'identification d'une personne n'est pas nouvelle. En fait, les corps policiers utilisent cette technique depuis plus de 100 ans. Le recours à l'empreinte digitale compte pour plus du tiers du marché des procédés biométriques. La classification des empreintes digitales est une étape très importante dans le système de l'identification automatique des empreintes digitales.

Dans ce chapitre, on va présenter le travail réalisé, qui aborde le problème de la classification des empreintes digitales. Celui-ci est basé sur les réseaux de neurones flous en utilisant les points singuliers globaux des empreintes digitales. On va donc montrer, discuter et évaluer les résultats obtenus pour l'extraction des caractéristiques, ainsi que pour la classification basée sur les réseaux de neurones artificiels et les réseaux de neurones flous adoptés.

V.2. Présentation du système implémenté

Le système qu'on a réalisé est constitué de plusieurs modules, chacun correspond à une tâche spécifique. Il a pour but de classer les empreintes digitales en cinq classes : Arc, Arc tendu, Boucle à gauche, Boucle à droite et Spire. La structure du système réalisé est présentée par la figure suivante (Fig.V.1) :

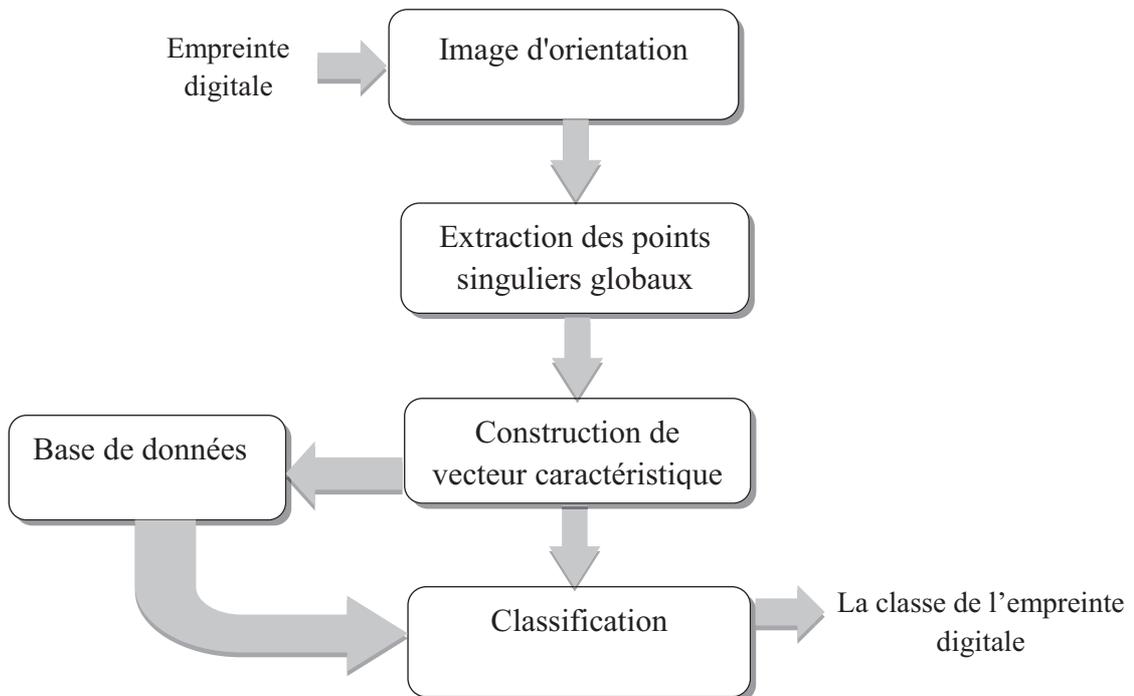


Fig.V.1. Structure de système de classification des empreintes digitales.

V.2.1. La base de données des images utilisées

Pour construire, valider et tester les structures des réseaux adoptés, on s'est basé sur une source de données qui est la base de données CASIA-FingerprintV5. Elle a été choisie dans la mesure où elle est fréquemment utilisée dans la littérature. Elle permet donc de pouvoir fournir un point de comparaison plus aisé avec d'autres travaux.

Les images des empreintes digitales qu'on a utilisé sont extraites de la base de données CASIA Fingerprint Image Version 5.0 (ou CASIA-FingerprintV5), qui contient 20,000 images d'empreintes digitales de 500 sujets. Les images d'empreintes digitales CASIA-FingerprintV5 ont été capturées à l'aide d'URU4000, qui est un capteur d'empreintes digitales en une seule session. Les bénévoles de CASIA-FingerprintV5 incluent des étudiants diplômés, les travailleurs, les serveurs, ...etc. Chaque bénévole a contribué 40 images d'empreintes digitales de ses huit doigts (5 images par doigt). Les volontaires ont été invités à tourner leurs doigts avec différents niveaux de pression. Toutes les images d'empreintes digitales sont de 8 bits en niveaux de gris des fichiers BMP et la résolution de l'image est de 328 * 356. La figure V.2 représente des échantillons de la base de données CASIA-FingerprintV5.

L'organisation de la base de données CASIA- FingerprintV5 est la suivante :

Elle contient 500 dossiers, le nom de chaque dossier est unique, qui correspond à une personne spécifiée. Les images de chaque dossier sont nommées par un code alphanumérique représentant des informations liées à l'image tel que : l'identifiant de la personne, la main gauche ou droite, le nom du doigt (pouce, index, majeur, l'annulaire) [71]. Les images de CASIA-FingerprintV5 sont stockées sous la forme suivante :
YYY_HX_KKK.bmp

YYY : l'identifiant unique de la personne, allant de 000 à 499

H : « L » désigne la main gauche, « R » désigne la main droite.

X : « 0 » désigne le pouce, « 1 » désigne index, « 2 » désigne majeur, « 3 » désigne l'annulaire.

K : c'est l'indice qui désigne la session, ces images ont été acquises en 5 sessions, cet indice varie de 0 à 4.

Pour notre application, on a besoin de deux bases de données, une pour l'apprentissage et l'autre pour le test. La première base est pour construire les modèles de classification, elle est constituée de 400 empreintes digitales de bonne qualité (complète), et

non bruitées, alors que la deuxième est pour l'évaluation des performances de celui-ci et qui comporte 200 empreintes.

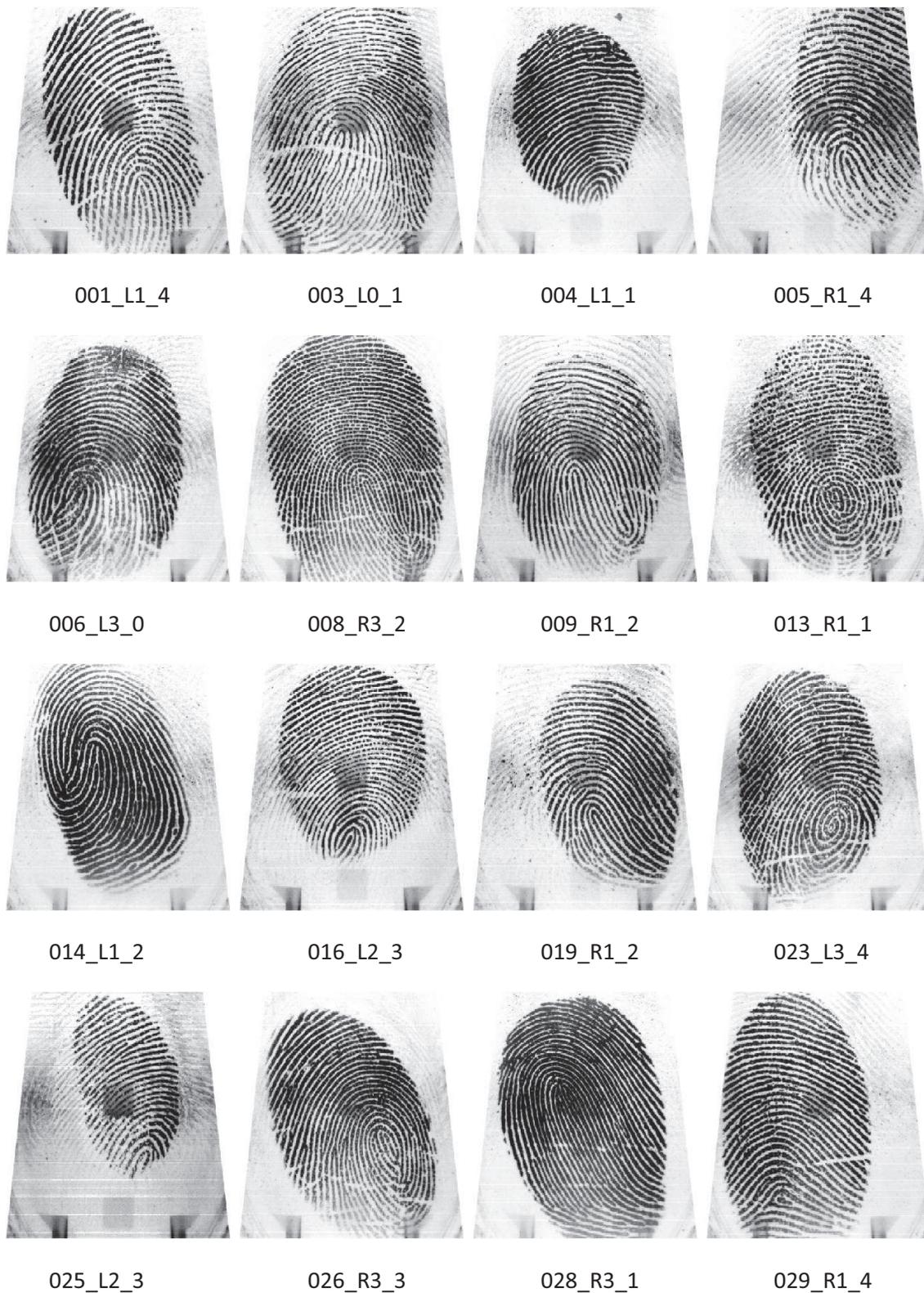


Fig.V.2. Echantillons de la base de données CASIA-Fingerprint Image Version 5.0.

V.2.2. Extraction d'image d'orientation de l'empreinte digitale

L'image d'orientation de l'empreinte digitale est obtenue en utilisant l'algorithme présenté dans le paragraphe (III.7.3.2, page 38) , la figure (Fig.V.3) illustre les résultats du gradient du Sobel et l'image d'orientation de l'empreinte digitale.

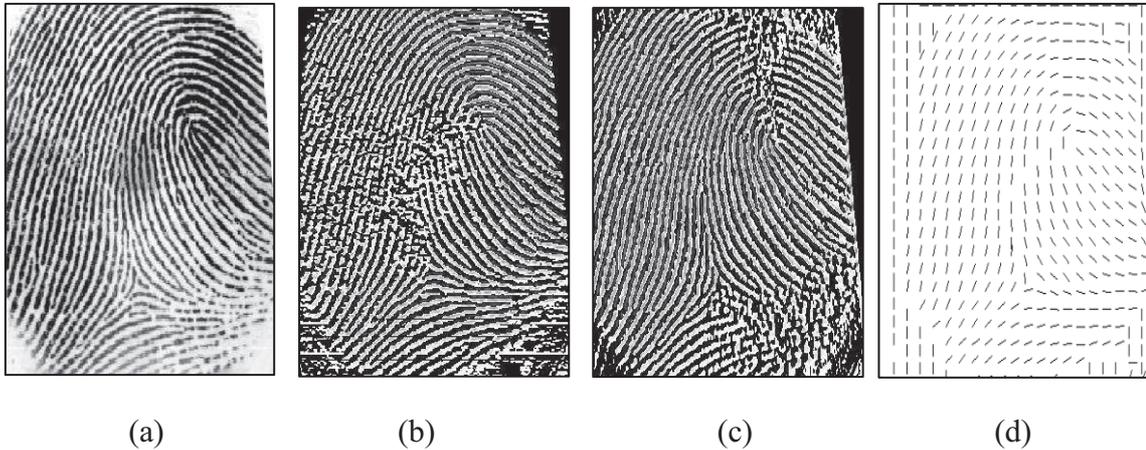


Fig.V.3. L'extraction de l'image d'orientation (a) L'image originale, (b) Le gradient selon x, (c) Le gradient selon y, (d) l'image d'orientation.

Les résultats obtenus pour l'extraction des images d'orientations sont très satisfaisants, et surtout pour les images de bonnes qualités.

V.2.3. Extraction des points singuliers globaux

Les points singuliers, le centre et le delta sont amplement appliqués dans la classification des empreintes digitales. De nombreux systèmes actuels de classification des empreintes digitales sont basés sur l'extraction de tels points singuliers. Néanmoins, dans certaines empreintes digitales, le centre et le delta ne sont pas facilement distingués et extraits. Par conséquent ces empreintes digitales ne sont pas exactement classées. Ces points singuliers représentent bien les caractéristiques globales de l'empreinte digitale.

Alors, l'extraction des points singuliers globaux de l'empreinte digitale consiste en deux étapes: la détection des points singuliers globaux et le post-traitement pour l'élimination des faux cores et deltas:

Etape1 : Détection des points singuliers globaux

Pour détecter les points singuliers globaux (core et delta) à partir de l'image d'orientation de l'empreinte digitale, on applique l'algorithme de l'index de Poincaré (voir le paragraphe III.7.3.4, page 40).

Cet algorithme donne de bons résultats pour la détection des points singuliers globaux (voir Fig.V.4), mais des faux cores et deltas apparaissent , ils appartiennent au fond de l'image de l'empreinte digitale et c'est pour cette raison qu'on a utilisé la segmentation afin de les éliminer.

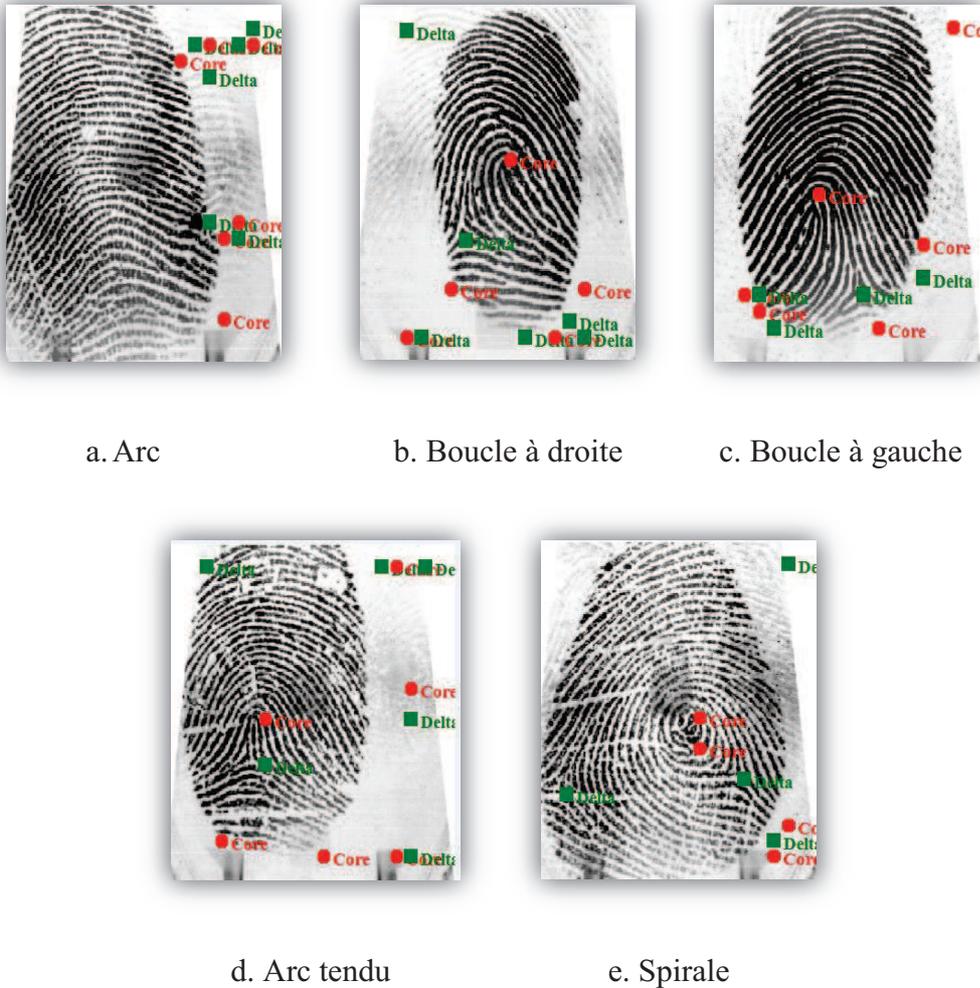


Fig.V.4. Le résultat de détection des points singuliers par l'algorithme de Poincaré pour les différentes classes.

Etape2 : Post traitement pour l'élimination des faux cores et deltas

On remarque que des faux cores et deltas apparaissent au fond de l'image de l'empreinte digitale. Afin de les éliminer, une opération de segmentation est nécessaire. Cette opération permet d'éliminer les blocs appartenant au fond et donc d'éliminer les faux cores et deltas, pour ce faire, on applique l'algorithme de segmentation qui est présenté dans le paragraphe (III.7.3.3, page 39) du troisième chapitre. Les résultats obtenus pour cette étape sont très satisfaisants.

La figure suivante (Fig.V.5) montre le résultat d'élimination des faux cores et deltas par la segmentation pour les différentes classes d'empreintes digitales.

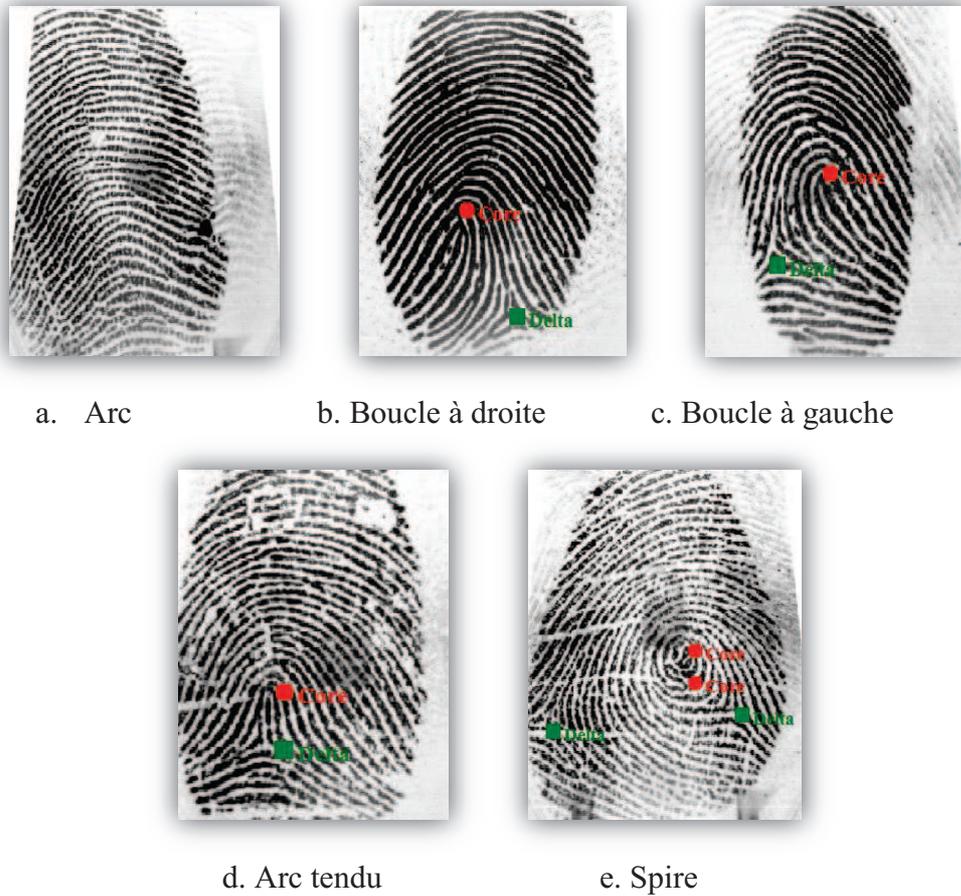


Fig.IV.5. Les résultats de l'extraction des points singuliers après la segmentation pour les différentes classes.

V.2.4. Construction du vecteur Caractéristique

Cette étape associe à chaque empreinte digitale donnée un ensemble d'attributs qui la caractérise appelé vecteur caractéristique X . Ce vecteur caractéristique d'entrée contient trois composantes $X = [\text{Nombre de core (NbCore)}, \text{Nombre de delta (NbDelta)}, \text{la position de delta par rapport au core (PosDelta)}]$.

Le nombre de cores et le nombre de deltas varient de 0 à 2. La troisième composante de vecteur caractéristique qui est la position du delta par rapport au core, est codée comme suit (voir Tab.V.1) :

Position du delta par rapport au core	Code
Droite	1
Milieu	2
Gauche	3

Tab.V.1. *Le codage la position du delta par rapport au core.*

V.2.5. Classification

Il s'agit de la dernière étape d'un système de classification des empreintes digitales. Celle-ci consiste à recevoir des caractéristiques extraites de l'empreinte digitale et de prendre une décision en ce qui concerne la classe à laquelle appartient cette empreinte digitale (arc simple, arc tendu, boucles à gauche, boucle à droite et les spires (tourbillons)).

Le vecteur de sortie représente les cinq décisions qui peuvent être prises et qui correspondent aux classes d'empreintes digitales : $Y = [\text{Arc (A)}, \text{Arc tendu (AT)}, \text{Boucle à gauche (BG)}, \text{Boucle à droite (BD)}, \text{Spire (S)}]$. Chaque composante représente le degré d'appartenance à cette classe.

On a codé la variable de sortie ou la décision par un codage binaire pur comme illustre le tableau suivant :

Classe de l'empreinte	Code binaire
Arc	10000
Boucle à droite	01000
Arc tendu	00100
Boucle à gauche	00010
Spire	00001

Tab.V.2. *Le codage binaire de variable de sortie.*

Pour cette étape on a utilisé deux approches: la première basée sur les réseaux de neurones artificiels et la deuxième basée sur les réseaux de neurones flous.

V.3. L'architecture du Réseau De Neurones Artificiels Adopté

Le réseau de neurones artificiels utilisé comporte trois couches : une couche d'entrée de trois neurones qui correspondent au vecteur caractéristique de l'empreinte digitale (nombre de cores, nombre de deltas et la position de delta par rapport au core), une couche cachée de quatre neurones et une couche de sortie de cinq neurones qui correspond au code binaire de la classe de l'empreinte digitale (voir Fig.V.6).

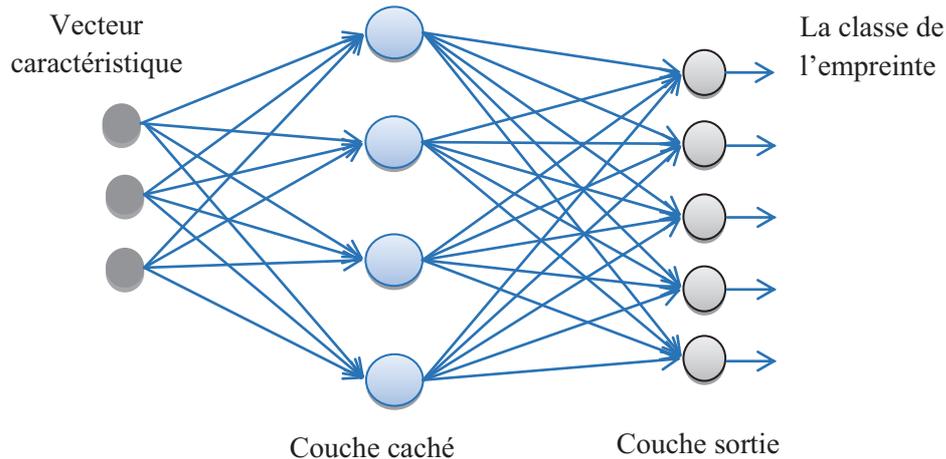


Fig. V.6. Structure de réseaux de neurones artificiels utilisés.

V.4. Les Architectures des Systèmes Neuro-Flous Adoptés

Afin de classifier les empreintes digitales, on utilise des modèles de réseaux neuronaux flous. Le choix de tels modèles se justifie par le fait qu'il est souvent utile lors de la phase d'identification, d'associer à l'empreinte, une probabilité d'appartenance à une classe donnée et de classer les différentes probabilités dans l'ordre décroissant. La classe la plus probable est celle qui correspond à la probabilité maximale.

Pour représenter les ensembles flous des variables linguistiques (NbCore, NbDelta, PosDelta) on a utilisé les fonctions d'appartenance de type triangulaire et de type trapézoïdale qui sont présentés par la figure (voir Fig. V.7) :

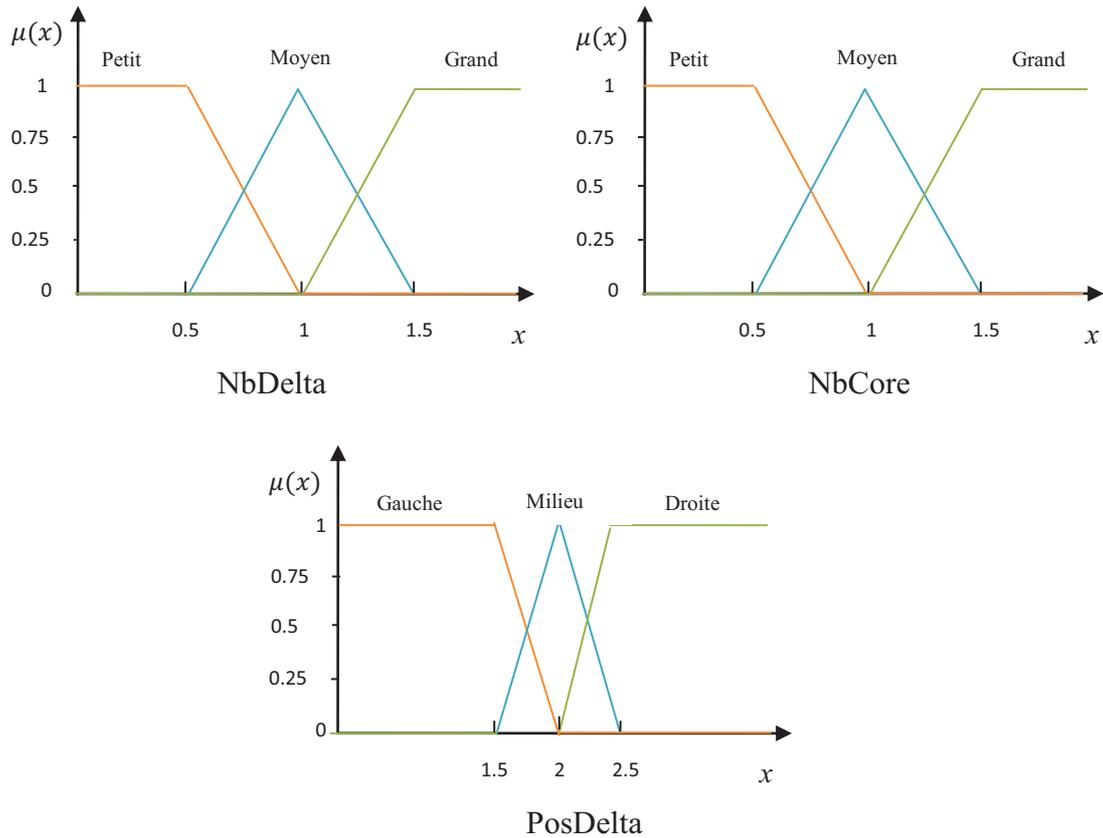


Fig. V.7. Les fonctions d'appartenance des variables d'entrée du système.

Comme les points singuliers représentent bien les caractéristiques globales de l'empreinte digitale. On établit les règles de classification en termes de points singuliers et leurs positions relatives. Pour les modèles neuro-flous, on a utilisé les cinq règles suivantes:

- \mathcal{R}_1 **Si** NbCore est *petit* et NbDelta est *petit* **Alors** Class est Arc.
- \mathcal{R}_2 **Si** NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *droite* **Alors** Class est Boucle à gauche.
- \mathcal{R}_3 **Si** NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *gauche* **Alors** Class est Boucle à droite.
- \mathcal{R}_4 **Si** NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *milieu* **Alors** Class est Arc tendu.
- \mathcal{R}_5 **Si** NbCore est *grand* **Alors** Class est Spire.

Pour la classification basée sur les réseaux de neurones flous, on a implémenté trois modèles : le réseau fusionné compact, le réseau NEFClass et le réseau de Pedrycz.

V.4.1. Le réseau fusionné compact :

Cette structure est un réseau de neurones de quatre couches. Chacune correspond à une étape du raisonnement flou (voir Fig.V.8). La couche d'entrée possède trois neurones. Chacun représente l'une des trois variables linguistiques (NbCore, NbDelta, PosDelta).

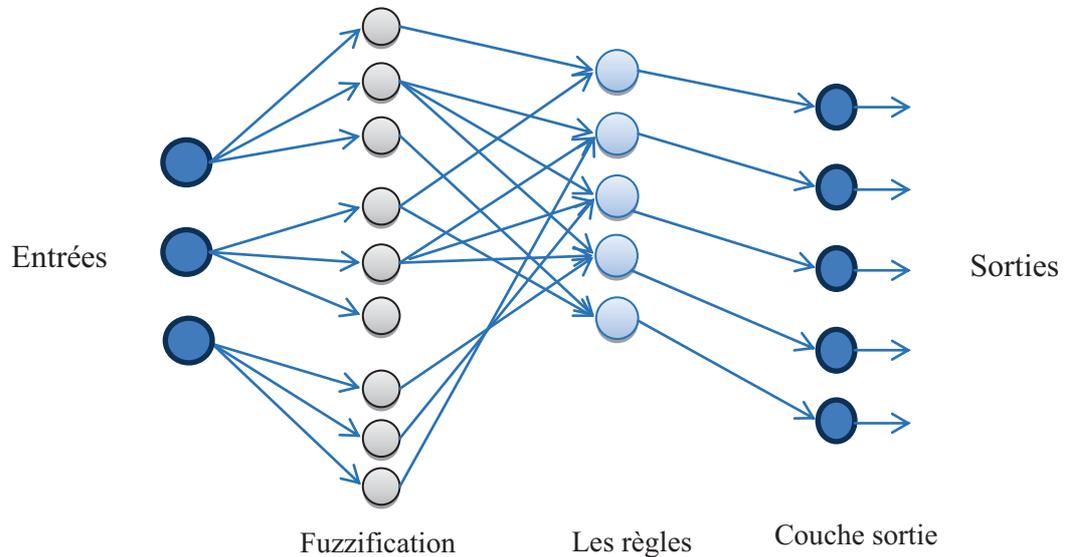


Fig.V.8. Réseau fusionné compact.

La seconde couche contient 9 neurones qui correspondent à l'ensemble des sous-ensembles flous définis pour toutes les entrées. Ainsi, les trois premiers neurones de cette couche représentent les trois sous-ensembles flous de la variable linguistique NbCore : petit, moyen, grand, même pour la variable NbDelta, et ceux de la variable linguistique PosDelta sont : gauche, milieu, droite. Les fonctions de sorties des neurones de cette couche sont les fonctions d'appartenances de ces variables linguistique à ses ensembles flous.

L'initialisation des poids de cette première couche cachée découle de ce raisonnement : les poids qui relient le neurone : NbCore aux neurones : petit, moyen et grand, sont égaux à 1 et ceux qui le relient aux autres termes linguistiques sont nuls. Le même principe est utilisé pour les autres variables linguistiques.

La troisième couche modélise les inférences floues. Elle est constituée de cinq neurones qui correspondent aux cinq règles floues définies.

Chaque neurone est lié à certains neurones de la couche précédente de telle sorte que la combinaison des sous-ensembles flous représentés par chaque neurone de la seconde couche donne la règle représentée par le neurone de cette couche. Ainsi, la combinaison du premier neurone (le nombre de core est petit), du quatrième neurone (le nombre de delta est petit), constitue la première règle représentée par le premier neurone de cette couche. Le poids initial de ces liaisons est donc 1 et le poids initial de toutes les autres liaisons est 0.

La couche de sortie comporte les cinq neurones qui correspondent aux cinq classes possibles, chaque neurone de cette couche est relié aux neurones des règles dont la partie conséquente correspond à la décision qu'il représente, le premier neurone de la couche des règles sera donc relié au premier neurone de sortie (qui correspond à la classe : arc). Le poids initial de cette liaison sera donc 1 et tous les autres seront nuls.

L'étape de défuzzification n'est pas modélisée car le système réalisé est un système de classification dont la sortie est une décision. La réponse qu'on désire obtenir du système est le degré d'appartenance de l'empreinte digitale à chacune des classes de sortie (arc, arc tendu, boucle à droite, boucle à gauche et spire).

V.4.2. Le réseau NEFClass :

C'est un réseau de neurones flous de trois couches, couche d'entrées, couche des règles, couche de sortie, les poids de la première couche sont calculées par les fonctions d'appartenance et les poids de la deuxième couche sont des poids booléennes (voir Fig.V.9).

Par exemple les poids reliant les trois neurones de couche d'entrée avec le premier neurone du couche des règles sont respectivement les valeurs des fonctions d'appartenance : $\mu_{NbCore}(Petit)$, $\mu_{NbDelta}(Petit)$, 0, ce qui donne la première partie de la première règle d'inférence.

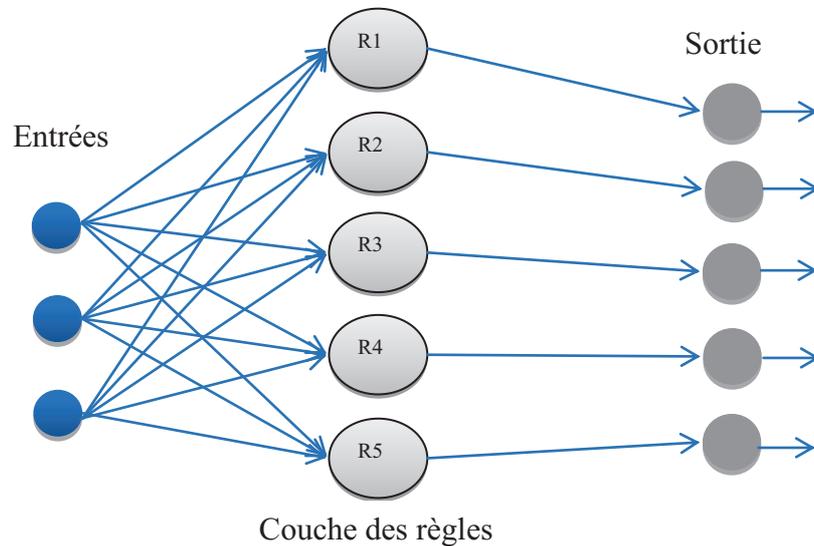


Fig. V.9. Réseau NEFClass.

Le poids qui relie le premier neurone de la couche des règles avec le premier neurone de la couche de sortie qui correspond à la partie conséquente de cette règle est égal à 1, et ceux qui le relient avec les autres neurones sont nuls. Le même principe est utilisé pour les autres neurones.

V.4.3. Le réseau de Pedrycz :

Le système comporte une partie de fuzzification des entrées suivie du réseau neuronal qui détermine les décisions à prendre. C’est un réseau de neurones flous composé de trois couches (voir Fig.V.10), la première couche contient les neurones qui correspondent aux entrées du système, la deuxième couche est la couche de fuzzification, elle possède neuf neurones qui correspondent aux sous-ensembles flous des entrées. La troisième couche comporte cinq neurones.

Les poids initiaux de la première couche ont le même raisonnement que le réseau fusionné compact, et ceux de la deuxième couche sont des poids aléatoires.

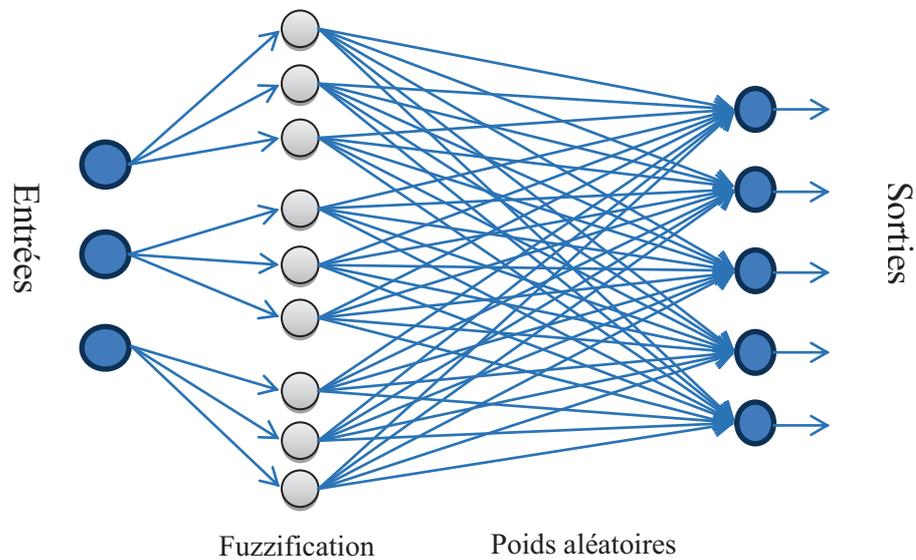


Fig. V.10. Réseau de Pedrycz.

Le tableau suivant présente pour chaque réseau de neurones le nombre de couches ainsi que le nombre de neurones dans chaque couche obtenue.

Le système	Nombre de couche	Nombre de neurones dans chaque couche
RNA	3	3
		4
		5
NEFClass	3	3
		5
		5
Compact	4	3
		9
		5
		5
Pedrycz	3	3
		9
		5

Tab.V.3. Le nombre de couches et le nombre de neurones de chaque système.

V.5. Tests et résultats

Pour le réseau de neurones artificiels et le réseau NEFClass, la fonction d'activation des neurones des couches est la fonction sigmoïde. Par contre, pour les deux réseaux : fusionné compact et Pedrycz les fonctions d'activation des neurones des couches de fuzzification sont les fonctions d'appartenance, et la fonction d'activation des neurones des autres couche est la fonction sigmoïde.

L'entraînement des différents systèmes implémentés se fait par la méthode de rétro-propagation de gradient de l'erreur. Le taux d'apprentissage est égal à 0,6 pour tous les systèmes.

En vue d'étudier, avec plus de détails, les performances des modèles obtenus et de pouvoir les comparer, on va commencer par les tester sur un ensemble d'exemples. Ce choix est dicté par un souci de clarté et de meilleure illustration des résultats obtenus. Et aussi, Pour déterminer la capacité de généralisation et de prise de décision, six exemples (Fig. V.11) sont présentés à chacune des structures des réseaux adoptées :

- **L'exemple 1** : empreinte digitale d'arc, qui ne comporte ni cores ni deltas.
- **L'exemple 2** : empreinte digitale de boucle à gauche, qui comporte un core à gauche et un delta.
- **L'exemple 3** : empreinte digitale de boucle à droite, qui comporte un core à droite et un delta.
- **L'exemple 4** : empreinte digitale d'arc tendu, qui comporte un core et un delta sur la même ligne droite.
- **L'exemple 5** : empreinte digitale de spire, qui comporte deux cores et ne comporte pas de deltas.
- **L'exemple 6** : empreinte digitale de spire, qui comporte quatre cores et trois deltas.

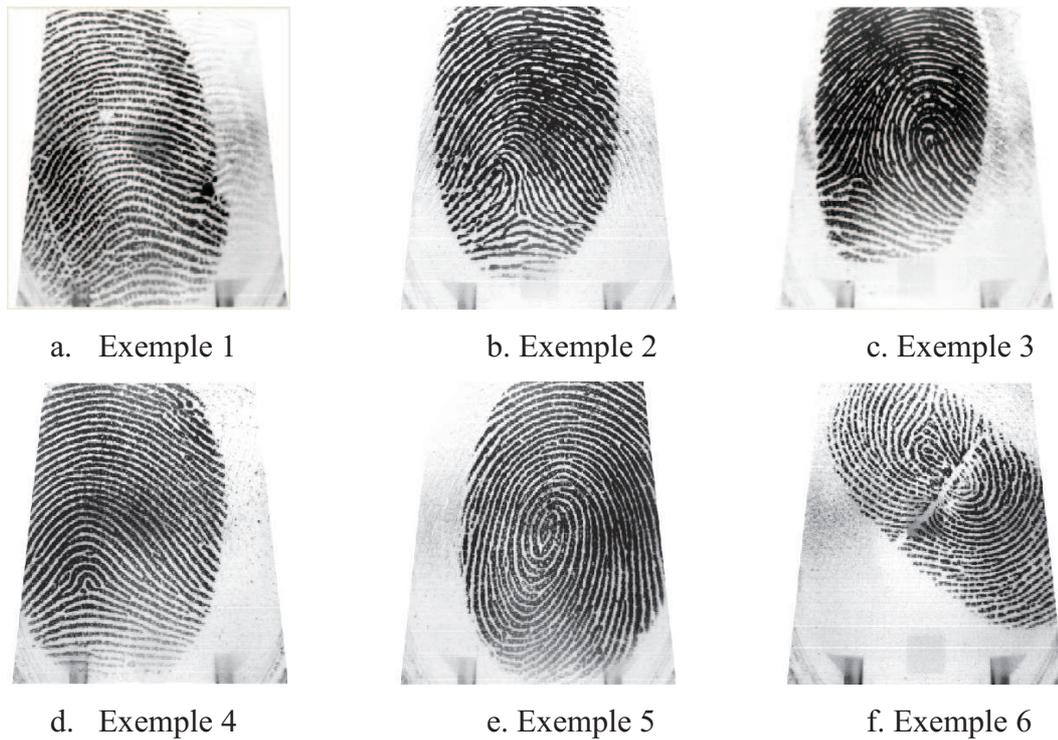


Fig. V.11. Les six exemples du test des quatre réseaux.

Les tableaux (Tab.V.4, Tab.V.5, Tab.V.6 et Tab.V.7) présentent pour chaque exemple les sorties des différents réseaux de neurones utilisés.

Exemple	Classe	Couche de sortie				
		1	2	3	4	5
1	Arc	0.9852	0.0115	0.0000	0.0141	0.0117
2	Boucle à gauche	0.0040	0.0000	0.0105	0.9875	0.0000
3	Boucle à droite	0.0075	0.9902	0.0077	0.0009	0.0014
4	Arc tendu	0.0002	0.0077	0.9865	0.0122	0.0045
5	Spire	0.0046	0.0043	0.0030	0.0000	0.9912
6	Spire	0.0348	0.9986	0.0026	0.0000	0.0018

Tab.V.4. Les sorties du réseau de neurones artificiels.

Exemple	Classe	Couche de sortie				
		1	2	3	4	5
1	Arc	0.9992	0.0001	0.0002	0.0000	0.0008
2	Boucle à gauche	0.0008	0.0009	0.0015	0.9976	0.0016
3	Boucle à droite	0.0006	0.9975	0.0017	0.0009	0.0014
4	Arc tendu	0.0026	0.0021	0.9962	0.0031	0.0007
5	Spire	0.0005	0.0005	0.0000	0.0005	0.9994
6	Spire	0.0000	0.0245	0.0001	0.0016	0.9994

Tab.V.5. Les sorties du réseau fusionné Compact.

Exemple	Classe	Couche de sortie				
		1	2	3	4	5
1	Arc	0.9992	0.0001	0.0002	0.0000	0.0008
2	Boucle à gauche	0.0007	0.0000	0.0012	0.9990	0.0000
3	Boucle à droite	0.0001	0.9970	0.0027	0.0001	0.0005
4	Arc tendu	0.0003	0.0043	0.9954	0.0015	0.0000
5	Spire	0.0007	0.0003	0.0000	0.0000	0.9993
6	Spire	0.0000	0.0257	0.0000	0.0000	1.0000

Tab.V.6. Les sorties du système NEFClass.

Exemple	Classe	Couche de sortie				
		1	2	3	4	5
1	Arc	0.9995	0.0004	0.0001	0.0001	0.0005
2	Boucle à gauche	0.0004	0.0014	0.0012	0.9984	0.0004
3	Boucle à droite	0.0008	0.9977	0.0013	0.0013	0.0007
4	Arc tendu	0.0006	0.0023	0.9972	0.0021	0.0006
5	Spire	0.0004	0.0003	0.0001	0.0001	0.9995
6	Spire	0.0003	0.1898	0.0007	0.0012	0.9992

Tab.V.7. Les sorties de réseau de Pedrycz.

Pour les exemples :1, 3, 4 et 5 les décisions prises par les systèmes utilisés sont tous valides, mais la sortie de réseau de Pedrycz est la plus proche au code binaire de sortie.

Pour l'exemple 2, la sortie de réseau NEFClass est la plus proche au code binaire du classe boucle à gauche.

Les décisions prises par les systèmes : NEFClass, Pedrycz et fusionné compact pour le sixième exemple sont valides, alors que le réseau de neurones artificiels présente un cas d'indécision, et la meilleure décision pour cet exemple est prise par le système NEFClass.

La performance d'un système de classification des empreintes digitales est généralement mesurée en termes de taux de classification. Le taux de classification est défini comme étant le rapport entre le nombre des empreintes correctement classées et le nombre total d'échantillons dans l'ensemble de test :

$$\text{Taux de classification} = \frac{\text{nombre des empreintes correctement classées}}{\text{nombre total d'échantillons dans la base de test}} \quad (\text{V.1})$$

Le tableau suivant (voir Tab.IV.8) présente les taux de classification des différents systèmes utilisés.

Modèle	Réseau de Neurons Artificiels	Le réseau fusionné compact	Le réseau de Pedrycz	Le système NEFClass
Taux de Classification	96%	100%	100%	100%

Tab.V.8. Les taux de classification des différents réseaux de neurones utilisés.

La représentation graphique des taux de classification des réseaux de neurones utilisés est donnée par la figure (Fig. V.12).

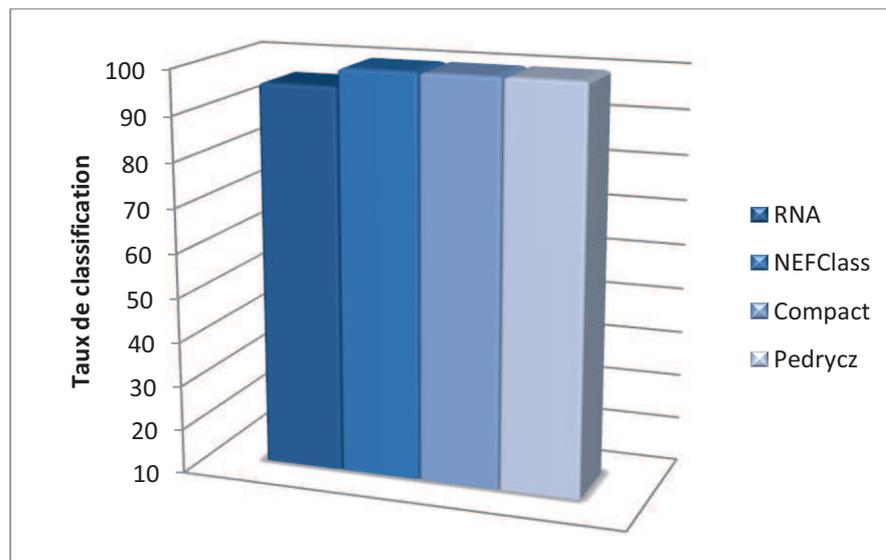


Fig. V.12. Représentation graphique de taux de classification des différents réseaux de neurones utilisés.

On remarque que les approches basées sur les réseaux de neurones artificiels et celles basées sur les réseaux de neurones flous fournissent des taux de classification très satisfaisants, et peuvent être utilisées pour la classification des empreintes digitales et que la performance des réseaux de neurones flous dépasse celle des réseaux de neurones artificiels. De plus, par comparaison, on peut conclure que le réseau de Pedrycz est le meilleur dans la prise de décision que les deux autres systèmes flous : NEFClass et réseau fusionné compact.

V.6. Discussion

D'après les résultats qu'on a obtenus, on peut faire les commentaires qui suivent :

- La logique floue est une technique adaptée pour la classification. En revanche, elle requiert une base de règles complète dont la détermination n'est pas toujours évidente. Si la base des règles n'est pas suffisante le système devient incapable de traiter tous les cas possibles. Ses résultats risquent d'être incohérents.

- Les réseaux de neurones artificiels sont également un bon outil d'aide à la décision, mais les cas d'indécision et les réponses erronées sont assez fréquentes.
- La base de données est un élément décisif dans le comportement des structures à architecture neuronale. Lorsqu'elle est petite, la capacité de généralisation du réseau est réduite. La majorité des sorties générées est erronée. L'augmentation de la taille de la base de données engendre une stagnation de l'erreur d'apprentissage. Le réseau devient lent et peut même diverger. Il y a donc un compromis à faire lors du choix du nombre d'exemples d'entraînement. Il doit y en avoir assez pour que le système génère les bonnes décisions et pas trop pour que le calcul de l'erreur ne stagne pas et que les décisions ne soit pas faussées.
- L'introduction des neurones flous ne donne de bons résultats que si l'architecture du réseau a un sens et qu'elle a été choisie en s'inspirant du raisonnement flou. Dans le cas contraire, aucune amélioration n'est apportée au système. Ainsi, un réseau de neurones flous dont la structure est arbitrairement choisie, génère des résultats obsolètes. Par contre, la structure de Pedrycz donne de meilleurs résultats lorsque les neurones sont flous.
- L'initialisation des poids et des biais joue un rôle primordial dans le bon apprentissage du système de classification et sa convergence.
- L'initialisation floue des poids a conduit à une amélioration de la génération en éliminant les cas d'indécisions. Ce résultat n'est pas systématique à cause des valeurs aléatoires des biais. Une détermination floue des biais devrait régler ce problème de manière définitive car toutes les valeurs seraient significatives.
- Il serait intéressant d'envisager une technique d'apprentissage qui permettrait de réadapter les poids à l'aide d'un système flou.
- L'initialisation aléatoire des poids et des biais pour le réseau de neurones artificiels peut mener à des cas d'indécisions. De plus, les résultats obtenus par les réseaux de neurones flous, sont plus intéressants que ceux des réseaux neuronaux. Ceci est dû au fait que les architectures de ces systèmes n'a été choisie arbitrairement mais de manière à modéliser le raisonnement flou. Alors, plus le réseau est assisté, plus sa capacité de généralisation s'améliore.

V.7. Conclusion

Dans ce travail, on a utilisé les réseaux de neurones flous dans le but de traiter le problème de classification des empreintes digitales en cinq classes selon la classification de Edward Henry. On a également fait une comparaison entre les réseaux de neurones artificiels et les réseaux neuronaux flous pour la prise de décision. On remarque que la fusion des réseaux de neurones avec la logique flous donne des très bons résultats.

L'opération de la classification d'une empreinte digitale est de déterminer correctement la classe à laquelle elle appartient. En se basant sur l'approche neuro-flou et l'utilisation des points singuliers globaux et leurs positions relatives, nos objectifs principaux sont atteints par l'obtention de bons taux de classification pour les différents modèles adoptés.

Nos perspectives visent à la modification des méthodes d'apprentissage par l'utilisation des approches méta-heuristiques d'optimisation.

Conclusion générale et perspectives

Dans ce travail, le problème de la classification des empreintes digitales a été abordé, ce problème est considéré complexe. L'approche proposée est une approche hybride. Le but de cette recherche est de construire une méthode de classification floue d'images, tirant profit d'une part, d'un modèle de classification floue basé l'estimation simultanée des classes, et d'autre part, de deux concepts d'apprentissage symbolique : systèmes d'inférence flous et les réseaux neuronaux.

De point de vue conception, les modèles proposés s'avèrent performants, dans le sens où ils bénéficient, des caractéristiques intrinsèques et avantageuses de chacun des outils cités précédemment. Ces caractéristiques peuvent se résumer par le parallélisme de traitement et la puissance de calcul qu'offrent les réseaux de neurones et la capacité de la logique floue de traiter l'imprécision et l'imperfection de données. Et aussi, l'utilisation des caractéristiques globales de l'empreinte digitale (le core et le delta) qui permettent de différencier d'une manière efficace deux empreintes digitales.

Le travail réalisé consiste en deux étapes fondamentales : la première consiste à l'extraction des points singuliers globaux (core et delta) de l'image de l'empreinte digitale et la deuxième c'est l'étape de classification purement dite. Pour cette dernière, on a adopté quatre modèles différents de réseaux :

- Le réseau de neurones artificiels ;
- Le réseau fusionné Compact ;
- Le réseau de Pedrycz ;
- Le réseau NEFClass ;

Afin de bien fixer les meilleurs paramètres des modèles de classification adoptés, une évaluation automatique est effectuée sur les résultats obtenus en faisant varier les paramètres : nombre de couches cachées, nombre de neurones dans chaque couche, fonction d'activation et taux d'apprentissage.

Après le développement, implémentation de ces modèles et l'analyse des résultats obtenus, on a pu constater que tous les modèles fournissent des taux de classification très satisfaisants, et que la performance des réseaux de neurones flous dépasse celle des réseaux de neurones artificiels, nous remarquons aussi que le meilleur parmi ces réseaux

dans la prise de décision est le réseau de Pedrycz. Nous concluons que l'utilisation de l'approche neuro-floue pour la classification augmente la crédibilité de la sortie du système de classification

L'opération de la classification d'une empreinte digitale est de déterminer correctement la classe à laquelle elle appartient. En se basant sur l'approche neuro-flou et l'utilisation des points singuliers globaux et leurs positions relatives, nos objectifs principaux sont atteints par l'obtention de bons taux de classification pour les différents modèles adoptés.

A la lumière de ce qui a été fait, l'objectif préliminaire a été en substance atteint. Mais en outre, des idées futures se dessinent. Nous croyons pouvoir affirmer que les performances du système réalisé pourraient être nettement améliorées si on ajoute d'autres règles d'inférences à la base des règles et/ou bien, ajouter d'autres informations au vecteur caractéristique. Aussi, nos perspectives visent à la modification des méthodes d'apprentissage par l'utilisation des approches méta-heuristiques d'optimisation : Recuit simulé, Algorithmes génétiques, Algorithmes des fourmis, Optimisation par essaim particulaire, ...etc.

Références Bibliographiques

- [1] M. ANDRE, *Introduction aux techniques de traitement d'images*, Eyrolles 1987.
- [2] <http://www.map.toulouse.archi.fr/works/panoformation/imagenum/imagenum.htm>, vu le 02/01/2013.
- [3] http://fr.wikipedia.org/wiki/R%C3%A9solution_spatiale_des_images_matricielles vu le 02/01/2013.
- [4] <http://humbert-lorent.tp-developpez.com/algorithmique/traitement/traitement.pdf>, vu le 01/02/2013.
- [5] R. C.GONZALES, P. WINTZ, *Digital Image Processing*, Addison Wesley, 1977.
- [6] E. TISSERAND, J. PAUTTEX, P. SCHWEITZER, *Analyse et traitement des signaux, Méthodes et applications au son et à l'image*, Dunod, Paris, pp 281-285, 2008.
- [7] B. GOSSELIN, " *Application des réseaux de neurones artificielles aux reconnaissances automatique de caractères manuscrits* ", Thèse de Doctorat, Faculté Polytechnique de Mons, 1996.
- [8] C. LURETTE, " *Développement d'une technique neuronale auto-adaptative pour la classification dynamique de données évolutives. Application à la supervision d'une presse hydraulique* ", Thèse de Doctorat, Université des sciences et technologies de LILLE 2003.
- [9] J. Bezdek, L. Hall, L. Clarke, " *Review of MR Image Segmentation Techniques using Pattern Recognition* ", *Medical Physics*, 20:1033-1048, 1993.
- [10] L. KHODJA, " *Contribution à la Classification Floue non Supervisée* ". Thèse de Doctorat, université de Savoie, France, 1997.
- [11] J. P. COCQUEREZ, S. PHILIPP, *Analyse d'images et segmentation*. Masson, 1995.
- [12] R. DUDA, P. HART, " *Pattern Classification and Scene Analysis* ". Wiley, New York, 1973. 3796-4/1997. IEEE.
- [13] M. SEMCHEDINE, L. TOUMI, A. MOUSSAOUI, " *Système Coopératif de Classification floue Possibiliste avec Rejet d'Ambiguïté Application à la Segmentation d'Images IRM* ", 4th International Conference on Computer Integrated Manufacturing CIP'2007. Sétif, Algérie.
- [14] J. MACQUEEN, " *Some methods for classification and analysis of multivariate observations* ". Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pages 281-297, 1967.
- [15] Y. LIM, S.U. LEE, " *On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques*". *Pattern Recognition*, 23(9), pages 935-

952, 1990.

- [16] R. Krishnapuram, J. Keller, "A possibilistic Approach to Clustering", *IEEE Trans. on Fuzzy Systems*, vol. 1, n. 2, May 1993.
- [17] M. SEMCHEDINE, L. TOUMI, A. MOUSSAOUI, " Classification Adaptative Robuste pour la Segmentation d'Images IRM Cérébrales ", JIG'2007, 3èmes Journées Internationales sur l'Informatique Graphique.
- [18] M. Ménard, " he fuzzy means: Solving The Extended Ambiguity Reject in Clustering ". In *IEEE Transactions on fuzzy systems*, vol. 1, N°.2, pages 195-203, 1998.
- [19] M. S. BOUGUELID, " Contribution à l'application de la reconnaissance des formes et la théorie des possibilités au diagnostic adaptatif et prédictif des systèmes dynamiques". Thèse de Doctorat, Université de Reims, champagne-ardenne école doctorale sciences, technologies, santé, France, 2007.
- [20] S. Nebti, " Optimisation par écosystème artificiels : application à la segmentation d'images ", thèses de Doctorat, Université de Constantine, 2007.
- [21] B. U Shanker, " Novel Classification and segmentation techniques with application to remotely sensed images ". *Lecture note on computer science Series 4400; Transactions on Rough Sets VII Springer Berlin /Heidelberg*, pp. 295-380, 2007.
- [22] S. Ameer, Z. Ameer, " Revue des approches de segmentation d'images textures: exemple des images météorologiques", 3rd International Conférence : Science of Electronic, Technologies of Information and télécommunication, Tunisia, 2005.
- [23] D. Guo, V. Atluri, N. Adam, " Texture based remote sensing image segmentation ". *Proceeding of IEEE International Conference on Multimedia and Expo*, pp1472-1475, 2005.
- [24] W. Wigner, B. Székely, " A review on image segmentation techniques with remote sensing perspective ", *ISPRS TC VII Symposium, Vienna, Austraiia, Vol XXXVIII, Part 7A*, 2010.
- [25] J. A. Canny, " Computational approach to edge detection ". *IEEE Pattern Ana. And Mach. Intelli.*, Vol. 8, 6, pp. 679-698. 1986
- [26] A. Herbulo, " Mesures statistiques non-paramétriques pour la segmentation d'images et de vidéos et minimisation par contours actifs ", Thèse de Doctorat, Université de Nice, France, 2007.
- [27] U. C. Benz, P. Hofmann, G. Willhauck, I. Lingenfelder, M. Heynen, " Multiresolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information ". *ISPRS Journal of Photogrammetry & Remote Sensing*, 58(3-4), pp.239-258, 2004.
- [28] S. L. Horowitz, T. Pavlidis, " Picture segmentation by tree transversal algorithm ". *J.*

- ACM., Vol. 32, 2, pp. 368-388, 1976.
- [29] A. Nakib, " Conception de métaheuristiques d'optimisation pour la segmentation d'images. Application à des images biomédicales". Thèse de Doctorat. Université de paris 12-val de marne France. 2007.
- [30] N. Monmarché, " Algorithmes des fourmis artificielles: applications à la classification et à l'optimisation ". Thèse de Doctorat. Université de Tours, France. Décembre 2000.
- [31] K. Takahashi, H. Nakatani, K. Abe. " Color Image Segmentation using ISODATA clustering method ". 2nd Asian Conf. On computer Vision, vol 1, pp 523-527, 1995.
- [32] A. P. Dempster, N. M. Laird, D. B. Rubin, " Maximum likelihood from incomplete data via the EM algorithm ". Journal of the Royal Statistical Society, Vol. 39, 1, pp. 1-38, 1977.
- [33] J. K. Fwu, P. M. Djuric, " EM algorithm for image segmentation initialized by tree structure scheme ". IEEE Trans. on Image Processing, Vol. 6, pp. 349-352, 1997.
- [34] Y. Bazi, Bruzzone, F. Melgani, " Image thresholding based on the EM algorithm and the generalized Gaussian distribution". Pattern Recognition, Vol. 40, pp. 619-634, 2007.
- [35] J. N. Kapur, P. K. Sahoo, A. C. K. Wong, " A new method for gray-level picture thresholding using the entropy of the histogram ". Computer Vision, Graphics and Image Processing, Vol. 29, pp. 273-285, 1985.
- [36] J. Kittler, and J. Illingworth, " Minimum error thresholding ". Pattern Recognition, Vol. 19, 1, pp. 38-52 1986.
- [37] M. Sezgin, B. Sankur, " Survey over image thresholding techniques and quantitative performance evaluation ". J. of Electronic Imaging, Vol. 13, 1, pp.146-165,2004.
- [38] N. Otsu, " A Threshold Selection Method from Gray-Level Histograms ". IEEE Trans. On Syst., Man and Cyb., Vol. 9, 1, pp. 62-66, 1979.
- [39] N. K. Pal, S. K. Pal., " Entropy: A new definition and its applications ". IEEE Trans. Syst. Man. Cybern., Vol. 21, pp. 1260-1270, 1991.
- [40] N. GALY, " *Etude de système complet de reconnaissance d'empreintes digitales pour un capteur microsysteme à balayage*", Thèse de Doctorat, Institut National Polytechnique de GRENOBLE, 2005.
- [41] H. V. DANG, " *Biométrie pour l'Identification*". Rapport final du tipe, Institut de la Francophonie pour l'Informatique, 2005.
- [42] R. Cappelli, D. Maio, " *The State of the Art in Fingerprint Classification* ", Automatic Fingerprint Recognition Systems, p183-193, 2004.

- [43] J. DES POMMAIRE, " *Analyse et classification des empreintes digitales* ", rapport du projet de recherche, Université de Caen Département d'informatique, 2009.
- [44] B. LE DUC, " *Authentification des empreintes digitales dans un système BioPKI* ", L'Institut de la Francophonie pour l'Informatique, 20 janvier 2007.
- [45] T. RAYMOND, " *Fingerprint image enhancement and minutiae extraction* ", Thèse PhD, Université de Western, Australie, 2003.
- [46] M. MOHAMED, H. NYONGESA, " *Automatic Fingerprint Classification System Using Fuzzy Neural Techniques*". School of Computing and Management Sciences, Sheffield Hallam University, Sheffield, IEEE, 2002.
- [47] D. MALTONI, D. MAIO, K. JAIN, S. PARABHAKAR, " *Handbook of Fingerprint Recognition* ", Springer. British Library of Cataloging in Publication Data, ISBN: 978-1-84882-253-5, 2009.
- [48] X. GUOL, Y. YIN, Z. SHI, " *Personalized Fingerprint Segmentation*", Springer-Verlag Berlin Heidelberg, pp 806, 2009.
- [49] M. KAWAGOE, A. TOKO, " *Fingerprint Pattern Classification*". Pattern Recognition, vol.17, pp. 295-303, 1984.
- [50] J. BO, T. HUA, P. XING LAN, " *Fingerprint Singular Point Detection Algorithm by Poincaré Index* ", la collection des ingénieurs d'Electronique et d'informatique, CHINA, 2008.
- [51] S. ALSSANDRA, " *Structural and graph-based methods for automatic fingerprint classification* ", thèse PhD, Université de Degli Studi di Cagliari, Département d'électronique et d'informatique.2010.
- [52] M. KAMIJO, " *Classifying fingerprint images using neural network: Deriving the classification state* ", *Neural Network*, pp. 1932-1937, 1993.
- [53] T. DUSENGE, " *La Reconnaissance des Empreintes Digitales* ", BA3-INFO Université Libre de Bruxelles, 25 mai 2009.
- [54] S. U. Maheswari, E. Chandra, " *A Review Study on Fingerprint Classification Algorithm used for Fingerprint Identification and Recognition* ", International Journal of Computer Science and Technology IJCST Vol. Issue 1, Jan-March 2012.
- [55] K. Karu, A. K. Jain, " *Fingerprint Classification* ", Proceedings of Pattern Recognition', Vol. 29, No. 3, pp.389-404, 1996.
- [56] F. A. Afsar, M. Azir, M. Hossain, " *Fingerprint Identification and Verification System using Minutiae Matching*", National Conference on Emerging Technologies, pp.141-146, 2004.
- [57] X. Tan, B. Bhanu, Y. Lin, " *Learning Features for Fingerprint Classification* ", AVBPA

- 2003, LNCS-2688, pp. 318-326. 2003.
- [58] S. Shesha, P. S. Sastry, " Fingerprint classification using a Feedback-Based Line Detector ", IEEE Transactions on Systems, Man and Cybernetics, part B: Cybernetics, vol. 34, no. 1, pp.85-94, 2004.
- [59] C. Park, H. Park, " Fingerprint classification using fast Fourier transform and nonlinear discriminant analysis", Pattern Recognition, Vol. 38, no. 4, pp. 495-503, 2008.
- [60] R. GHAYOULA, " *Contribution à l'Optimisation de la Synthèse des Antennes Intelligentes par les Réseaux de Neurones* ", Thèse de Doctorat, Laboratoire de Physique de la Matière Molle, Unité de Recherche des Circuits et Systèmes d'Electroniques à Haute Fréquence, Tunisie, 2008.
- [61] P. BORNE, J. HAGGEGE, *Les réseaux de neurones présentation et applications*, Paris, Edition Technip, 2007.
- [62] O. GUENOUNOU, " *Méthodologie de conception de contrôleurs intelligents par l'approche génétique- application à un bioprocédé* ", Thèse de Doctorat, Systèmes Automatiques, Université Toulouse III - Paul Sabatier, 2009.
- [63] R. JEAN-PHILLIPE, *Réseaux neuronaux, Une introduction accompagnée d'un modèle Java*. Vuibert, Paris, pp 56-57, 2010.
- [64] O. RABIH. " *Modélisation du moteur Diesel, en vue de sa commande dynamique par réseaux de neurones pour ministre les émissions* ", Thèse de Doctorat, L'école centrale de lyon, 2007.
- [65] J. FAUCHER, " *Les plans d'expériences pour le réglage de commandes à base de logique floue* ", Thèse de Doctorat, Institut national Polytechnique de Toulouse. Spécialité : Génie Electrique et Automatique, 2006.
- [66] L. GACOGNE, " *Eléments de la logique floue* ", Paris, Hermès 1997.
- [67] H. GERALD, " *Processeurs flous* ", he-arc baptiste savoye 26, ch-2610 st-imier, Mai 2008.
- [68] M. MORDJAOUI, " *Modélisation des effets électromagnétiques, Apport de la logique floue et neuro-floue* ". Thèse de Doctorat, Université El-hadj Lakhdar –BATNA. Juin 2008.
- [69] R. MAHDAOUI, H.L.MOUSS, O.CHOUHAL, H.HOUASSI, " *Apprentissage des systèmes Neuro-Flous par des données numériques et/ou symboliques pour la reconnaissance dynamique de formes Application au diagnostic industriel* ", ICIEM'10 : International Conference on Industrial Engineering and Manufacturing Batna-, Algérie, 2010.

- [70] F. Hoffmann, B. Baesens, J. Martens, F. Put, J. Vanthienen, " *Comparing a genetic fuzzy and a Neuro fuzzy Classifier for credit scoring* ", International journal of intelligent Systems (15), pp. 785-800, 2000.
- [71] Note on CASIA-Fingerprint, Fingerprint identification Research Group, Center for Biometrics and Security Research. National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences.
<http://www.idealtest.org/dbDetailForUser.do?id=7>

Liste des contributions

Publications Scientifiques:

O. Assas et al. "Comparison of Neuro-fuzzy Models for Classification Fingerprint Images", International Journal of Computer Applications,(0975-8887), volume 65-No9, March 2013.

Digital Library URI : <http://www.ijcaonline.org/archives/volume65/number9/10951-5911> ISBN : 973-93-80873-49-1

Communication Scientifiques :

1/ Artificial Neural Network Models for predicting Daily Global Solar Radiation Using Meteorological Data in Djelfa, Algeria (ICRE'12 University of Bejaia).

2/ Modeling of global solar radiation in Algeria based on geographical and all climatic parameters (sienr2012. Unité de Recherche Appliquée en Energies Renouvelables. URAER. Ghardaïa.)

3/Fuzzy C-Partition Using Particle Swarm Optimization Algorithm (ICCS'12 International Conférence en Complex Systems Agadir, Morocco. 978-1-4673-4766-2/12/\$31.00 ©2012 IEEE. <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6422825>

4/Image Thresholding Using Particle Swarm Optimization Algorithm Conférence Internationale des Technologies de l'Information et de la Communication CITIC'2009. SETIF Algerie, du 04-06, Mai 2009