



N° d'ordre :

Université de M'sila

Faculté des Mathématiques et de l'Informatique

Département de l'Informatique

Mémoire présenté pour l'obtention du diplôme de

Master Académique en Informatique

Domaine : Mathématique et Informatique

Filière : Informatique fondamentale

Option : Systèmes d'information Avancés

Par

Ahlem ADJIMI

Thème

Classification des empreintes digitales
par les réseaux de neurones flous

Soutenu publiquement le : 27/06/2012 devant le jury composé de :

Mr. B.Bouderah	Prof. Université de M'sila	Rapporteur
Mme. O.Assas	M.A(A). Université de M'sila	Rapporteur
Mme. K.Malek	M.A(A). Université de M'sila	Président
Mme. M.Boudia	M.A(A). Université de M'sila	Examineur

Promotion : 2011/2012

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciement

Je rends mes profondes gratitude à dieu qui m'a aidé à réaliser ce travail.

Je tiens à exprimer mes sincères remerciements à mes encadreurs :

Monsieur BOUDERAH Ibrahin et Madame ASSAS Ouarda

pour leur conseils, leur aides et leur encouragements.

Je tiens à exprimer aussi mes sincères remerciements à :

Mes parents, pour leur encouragement, leur soutien et pour les sacrifices qu'ils ont enduré.

Je remercie très vivement les membres de jury pour avoir accepté de juger ce modeste travail.

Enfin, j'exprime mes remerciements à tous les professeurs de département d'Informatique.

Adjimi Ahlem

Table des Matières

Liste des figures	V
Liste des tableaux	VII
Introduction générale	1
<i>Chapitre I Généralités sur le traitement d'images</i>	
I.1. Introduction.....	3
I.2. Définition de l'image.....	3
I.3. Définition de l'image numérique.....	3
I.4. Types d'images.....	4
I.4.1. Image matricielle (bitmap)	4
I.4.2. Image vectorielle	4
I.5. Les images bitmap.....	4
I.5.1. Concepts et définitions de base.....	4
I.5.1.1. Pixel.....	4
I.5.1.2. Le codage de pixel.....	5
I.5.1.3. La taille d'une image	5
I.5.1.4. La résolution d'une image	5
I.5.1.5. La luminance	6
I.5.1.6. Le contraste	6
I.5.1.7. Le bruit	6
I.5.1.8. L'histogramme	6
I.5.1.9. La palette des couleurs	7
I.6. Le codage des couleurs	7
I.7. Formats de fichiers d'images	8
I.8. La structure d'un fichier BMP	9
I.9. Quelques traitements d'images	11
I.9.1. Binarisation	11
I.9.2. Squelettisation	12
I.9.3. Segmentation	12
I.9.4. Convolution	12
I.9.5. Filtrage	13

I.9.5.1. Filtres linéaires	13
I.9.5.2. Filtres non linéaires	15
I.9.6. Morphologie mathématique	16
I.9.7. La détection de contours	16
I.10. Conclusion	18
 <i>Chapitre II Classification d'empreintes digitales</i>	
II.1. Introduction	19
II.2. Historique	19
II.3. L'empreinte digitale	20
II.4. Caractéristiques des empreintes digitales	20
II.4.1. Les points singuliers globaux	20
II.4.2. Les points singuliers locaux (minuties)	21
II.5. Classes des empreintes digitales	21
II.6. Intérêts de l'empreinte digitale	23
II.7. Description d'un système de classification des empreintes digitales.....	23
II.7.1. L'acquisition de l'empreinte	24
II.7.2. Le prétraitement	25
II.7.3. L'extraction des caractéristiques	25
II.7.3.1. Normalisation	25
II.7.3.2. Estimation de champ d'orientation d'image d'empreinte digitale	26
II.7.3.3. Segmentation	28
II.7.3.4. Détection des points singuliers globaux.....	28
II.7.4. Classification	31
II.8. Approches de classification des empreintes digitales	31
II.8.1. Approches statistiques	31
II.8.2. Approches structurales	31
II.8.3. Approches syntaxiques	32
II.8.4. Approches des réseaux de neurones	33
II.8.5. Approches basées sur les règles	33
II.9. Conclusion	34

Chapitre III Les Réseaux de Neurones flous

III.1. Introduction	35
III.2. Notions fondamentales sur les réseaux de neurones	35
III.2.1. Introduction	35
III.2.2. Neurone biologique	35
III.2.3. Neurone artificiel	36
III.2.4. L'apprentissage	37
III.2.4.1. Types d'apprentissages	37
III.2.4.2. Les méthodes d'apprentissage	38
III.2.4.3. Les réseaux de neurones les plus utilisés	39
III.2.5. La rétro-propagation du gradient de l'erreur	42
III.2.5.1. Principe	42
III.2.5.2. Algorithme	42
III.2.5.3. Choix de critère à minimiser	43
III.2.5.4. Conditions d'arrêt de l'apprentissage	44
III.2.5.5. Avantages et inconvénients	44
III.2.6. Avantages et Inconvénients des réseaux de neurones	45
III.2.6.1. Avantage des réseaux de neurones	45
III.2.6.2. Inconvénients des réseaux de neurones	45
III.3. Les Réseaux de Neurones Flous	46
III.3.1. La logique floue	46
III.3.1.1. Notion de sous-ensemble flou	46
III.3.1.2. Variable et valeur de variable linguistique	46
III.3.1.3. Les fonctions d'appartenance	47
III.3.1.4. Règles d'inférence	48
III.3.1.5. Système d'inférence floue	48
III.3.2. Systèmes Neuro-Flous	50
III.3.2.1. Définition	50
III.3.2.2. Le perceptron flou	51
III.3.3. Les combinaisons des réseaux de neurones avec la logique floue	51
III.3.3.1. Réseau flou neuronal	51
III.3.3.2. Système neuronal/flou simultanément	51
III.3.3.3. Modèles neuro-flous coopératifs	51
III.3.3.4. Modèles neuro-flous hybrides	51

III.3.4. Les types des réseaux de neurones flous utilisés pour la classification.....	52
III.3.4.1. La structure de Pedrycz	52
III.3.4.2. La structure fusionnée compacte	52
III.3.4.3. Les systèmes neuro-flous de classifications NEFClass	53
III.4. Conclusion	54
<i>Chapitre IV Conception et réalisation</i>	
IV.1. Introduction.....	55
IV.2. Plateforme et outils de développement.....	55
IV.3. Présentation et implémentation du système.....	56
IV.3.1. Système de classification d’empreintes digitales.....	56
IV.3.1.1. La base de données des images utilisées	56
IV.3.1.2. Extraction d’image d’orientation de l’empreinte digitale.....	57
IV.3.1.3. Extraction des points singuliers globaux.....	57
IV.3.1.4. Le codage des données	59
IV.3.1.5. Classification.....	60
A. Réseaux de Neurones Artificiels.....	60
B. Réseaux de Neurones Flous.....	61
IV.4. Tests et résultats.....	66
IV.5. Quelques interfaces de l’application.....	69
IV.6. Conclusion.....	71
Conclusion générale et perspectives.....	72
Bibliographie	74

Liste des figures

Fig.I.1. <i>Imge matricielles</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.I.8. <i>Détection des contours d'une image par convolution de type Sobel</i>	18
Fig.I.9. <i>Détection des contours d'une image par convolution de type Laplacien</i>	18
Fig.II.1. <i>L'empreinte digitale</i>	20
Fig.II.2. <i>Les points singuliers globaux</i>	21
Fig.II.3. <i>Les différents types de minuties</i>	21
Fig.II.4. <i>Les cinq classes des empreintes digitales</i>	22
Fig.II.5. <i>Schémas générale d'un système de classification des empreintes digitales</i>	24
Fig.II.6. <i>L'orientation de la crête locale du pixel(x, y)</i>	26
Fig.II.7. <i>L'image d'orientation d'une empreinte digitale</i>	27
Fig.II.8. <i>Effet de la segmentation sur l'image</i>	28
Fig.II.9. <i>L'index de Poincaré calculé sur une courbe C</i>	29
Fig.II.10. <i>L'index de Poincaré calculé avec un voisinage de $N=8$</i>	29
Fig.II.11. <i>Détection des points singuliers par l'algorithme de l'index de Poincaré</i>	30
Fig.II.12. <i>Le graphe relationnel d'une image d'empreinte digitale</i>	32
Fig.II.13. <i>Un schéma de l'approche chaîne de la construction dans Rao et Balck 1980</i> ...	33
Fig.III.1. <i>Neurone biologique</i>	36
Fig.III.2. <i>Neurone artificiel</i>	36
Fig.III.3. <i>fonctions d'activations les plus utilisées</i>	37
Fig.III.4. <i>Schéma général de perceptron simple</i>	40
Fig.III.5. <i>Exemple de réseau de type perceptron Multicouche</i>	41
Fig.III.6. <i>Réseaux de neurones bouclés</i>	42
Fig.III.7. <i>Variable linguistique</i>	47
Fig.III.8. <i>Formes des fonctions d'appartenance usuelles</i>	48
Fig.III.9. <i>Structure d'un système d'inférence floue</i>	49
Fig.III.10. <i>principe du système neuro-flou</i>	50

Fig.III.11. <i>Architecteur de réseau de Pedrycz</i>	52
Fig.III.12. <i>Architecteur de réseau fusionnée compacte</i>	53
Fig.III.13. <i>Architecteur d'un système NEFClass</i>	53
Fig.IV.1. <i>L'interface de Microsoft Visuel Studio 2010</i>	55
Fig.IV.2. <i>Structure de système de classification des empreintes digitales</i>	56
Fig.IV.3. <i>L'extraction de limage d'orientation de l'empreinte digitale</i>	57
Fig.IV.4. <i>Détection des points singuliers avant la segmentation</i>	58
Fig.IV.5. <i>Détection des points singuliers après la segmentation</i>	59
Fig.IV.6. <i>Structure de réseaux de neurones artificiels utilisés</i>	61
Fig.IV.7. <i>Les fonctions d'appartenance des variables d'entrée du système</i>	61
Fig.IV.8. <i>Réseau fusionné compact</i>	62
Fig.IV.9. <i>Réseau NEFClass</i>	64
Fig.IV.10. <i>Réseau de Pedrycz</i>	65
Fig.IV.11. <i>Taux de classification des différents réseaux de neurones</i>	68
Fig.IV.12. <i>Interface principale</i>	69
Fig.IV.13. <i>Le menu Fichier</i>	70
Fig.IV.14. <i>Le menu Prétraitement</i>	70
Fig.IV.15. <i>Le menu Classification</i>	71

Liste des tableaux

Tab.I.1. <i>Entête du fichier BMP</i>	9
Tab.I.2. <i>Entête de l'image BMP</i>	10
Tab.I.3. <i>Masque de convolution passe-bas</i>	14
Tab.I.4. <i>Masque de convolution passe-haut</i>	14
Tab.I.5. <i>Masque de convolution Gaussien</i>	14
Tab.I.4. <i>Les différents masques de convolution utilisée pour la détection du contour.....</i>	17
Tab.II.1. <i>Les classes d'empreintes et leurs nombre de points singuliers.....</i>	34
Tab.IV.1. <i>Le codage de variables d'entrées.....</i>	60
Tab.IV.2. <i>Le codage binaire de variable de sortie.....</i>	60
Tab.IV.3. <i>Le nombre de couche et le nombre de neurones de chaque système.....</i>	65
Tab.IV.4. <i>Les sorties du réseau de neurones artificiels.....</i>	66
Tab.IV.5. <i>Les sorties du réseau fusionné Compact.....</i>	67
Tab.IV.6. <i>Les sorties du système NEFClass.....</i>	67
Tab.IV.7. <i>Les sorties de réseau de Pedrycz.....</i>	67
Tab.IV.8. <i>Les taux de classification des différents réseaux de neurones utilisés.....</i>	68

Introduction

Générale

Introduction générale

La biométrie est un ensemble des technologies qui exploitent les caractéristiques humaines physiques ou comportementales telles que l'empreinte digitale, la signature, l'iris, la voix, le visage...etc, pour différencier des personnes.

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 (par exemple, la base de données de FBI contient plus de 200 millions des empreintes digitales). Pour réduire le temps de recherche et la complexité, il est souhaitable de classifier ces empreintes digitales d'une façon précise et cohérente de sorte que l'empreinte digitale d'entrée doit être assortie seulement avec un sous-ensemble des empreintes digitales dans la base de données.

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 classification des empreintes digitales se réfère au problème de l'affectation d'une empreinte digitale à une classe d'une manière cohérente et fiable. Plusieurs méthodes de classification des empreintes digitales ont été proposées dans la littérature, la plupart sont basées sur la structure globale de l'empreinte digitale, dans notre travail on utilise les points singuliers globaux de l'empreinte digitale qui sont le core et le delta.

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. Ainsi dans notre travail, on utilise les réseaux de neurones flous pour classifier les empreintes digitales.

L'objectif de ce travail est donc de concevoir et mettre en œuvre un système de classification des empreintes digitales selon la classification de Henry, basé sur les réseaux de neurones flous en utilisant les caractéristiques globales de celles-ci.

Ce mémoire s'articule autour de quatre 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, le format d'un fichier BMP et les filtres les plus utilisées pour l'amélioration de la qualité des images.

Le deuxième 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 troisième chapitre présente en premier lieu des notions fondamentales sur les réseaux de neurones artificielles, ensuite décrit la logique floue et en fin les réseaux de neurones flous et leurs principaux types et modèles.

Et le dernier chapitre permettra d'évaluer les performances des différentes approches implémentées en présentant les résultats obtenus avec interprétation.

Chapitre I

Généralités sur le Traitement d'Images

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....

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

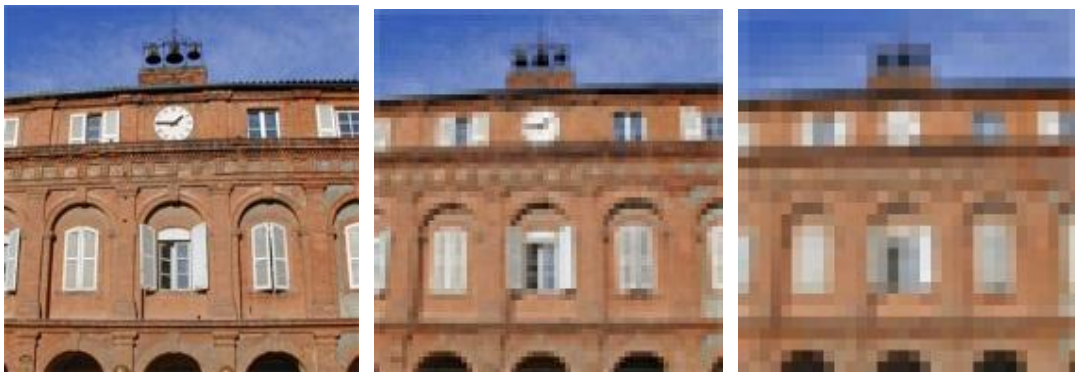
La taille d'une image est le nombre de pixels de cette image, la taille d'une image qui est représenté par (328×456) dont 328 est le nombre de lignes, et 456 est le nombre de colonnes, est égale à : $328 \times 456 = 149568$ pixels.

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].



(a)

(b)

(c)

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 :

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

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.

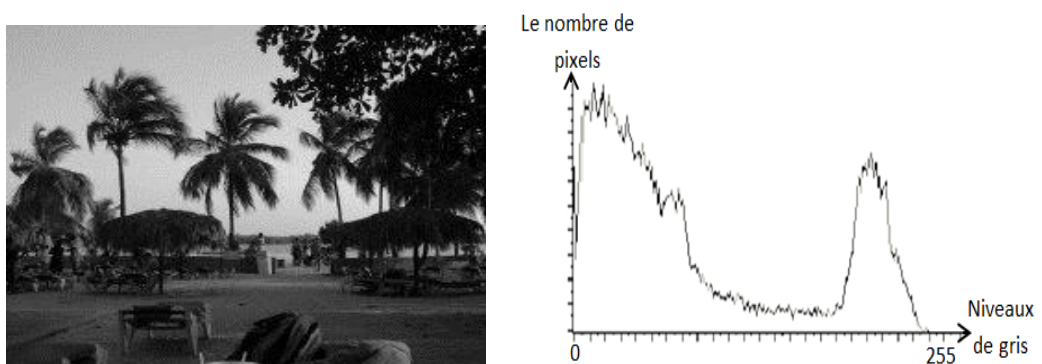


Fig.I.3. Exemple d'histogramme d'une image.

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).

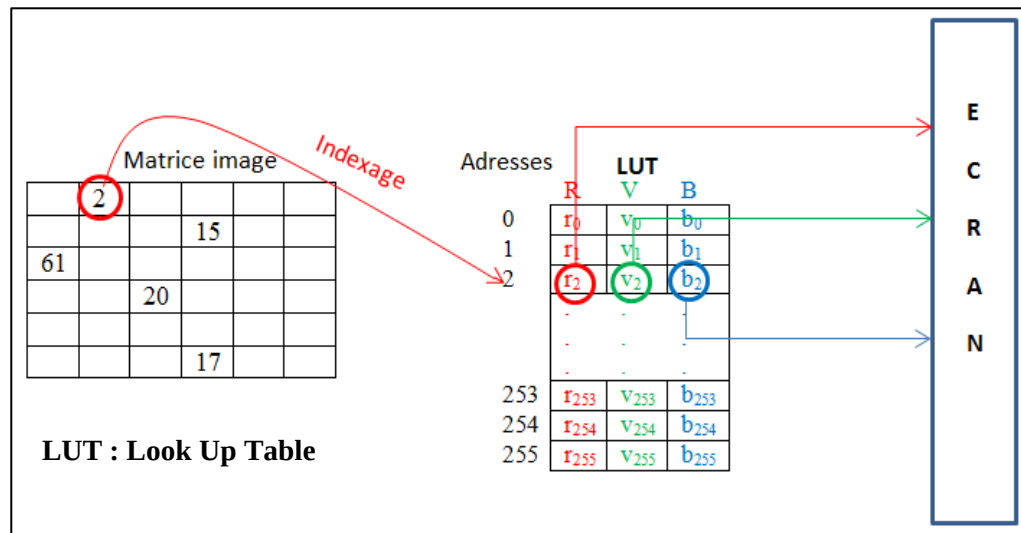


Fig.I.4. La transformation ponctuelle d'une image numérique.

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 $[0...255]$, 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,...ect.

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. La structure d'un fichier BMP

Un fichier BMP est un fichier bitmap, c'est-à-dire un fichier d'image graphique stockant les pixels sous forme d'un tableau bidimensionnel de pixels et gérant les couleurs soit en couleur vraie soit grâce à une palette indexée. La structure d'un fichier bitmap est la suivante :

I.8.1. En-tête du fichier

L'entête du fichier fournit des informations comme le montre le tableau suivant :

Nom du champ	Longueur en octets	Signification
Identifier	2 octets	Contient toujours l'octet 'B' suivi de l'octet 'M'.
FileSize	4 octets	Taille totale du fichier en octets.
Reserved	4 octets	Champ réservé, doit être égal à 0.
DataOffset	4 octets	Nombre d'octets séparant le début du fichier des données de l'image.

Tab.I.1. Entête du fichier BMP.

I.8.2. En-tête du bitmap

L'entête de l'image fournit des informations sur l'image, comme le montre le tableau suivant :

Nom du champ	Longueur en octets	Signification
HeaderSize	4 octets	La taille de l'entête de l'image en octets
Width	4 octets	Largeur de l'image en pixels
Height	4 octets	Hauteur de l'image en pixels
Planes	2 octets	Le nombre de plans. Cette valeur vaut toujours 1
BitsPerPixels	2 octets	Nombre de bits nécessaires pour représenter un pixel
Compression	4 octets	Type de compression
BitmapDataSize	4 octets	Taille en octets des données de l'image
HResolution	4 octets	Résolution horizontale de l'image en pixels par mètre
VResolution	4 octets	Résolution verticale de l'image en pixels par mètre
Colors	4 octets	Nombre de couleurs dans l'image
ImportantColors	4 octets	Nombre de couleurs importantes

Tab.I.2. Entête de l'image BMP.

I.8.3. La palette des couleurs

Une couleur est codée sur 4 octets : |R|V|B|Réservé|. La taille de la palette est donc de 1024 octets.

Pour les images BMP en mode :

- 24 bits, la palette n'est pas nécessaire (on peut représenter toutes les couleurs). Dans ce cas, la palette n'existe pas dans le fichier BMP.
- 8 bits, on ne peut représenter que 256 couleurs. Il faut définir une correspondance entre la couleur du pixel et les trois composantes (R, V et B). Dans ce cas une table a été ajoutée juste après l'en-tête du bitmap (octet 54), qui fournit pour chaque valeur (de 0 à 255) les trois composantes RVB qui y correspondent [3].

I.8.4. Les données de l'image

Les données de l'image représentent les pixels de l'image. Contrairement à la plupart des formats d'images, les pixels de l'image sont codés en partant de la ligne inférieure de l'image, où chaque ligne (codée de gauche à droite) doit toujours occuper un nombre d'octets multiple de 4, excepté si l'image est compressée. Si la ligne ne possède pas un nombre d'octets multiple de 4, on remplit la ligne en ajoutant des zéros [3].

I.9. Quelques traitements d'images

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

I.9.1. 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 :

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

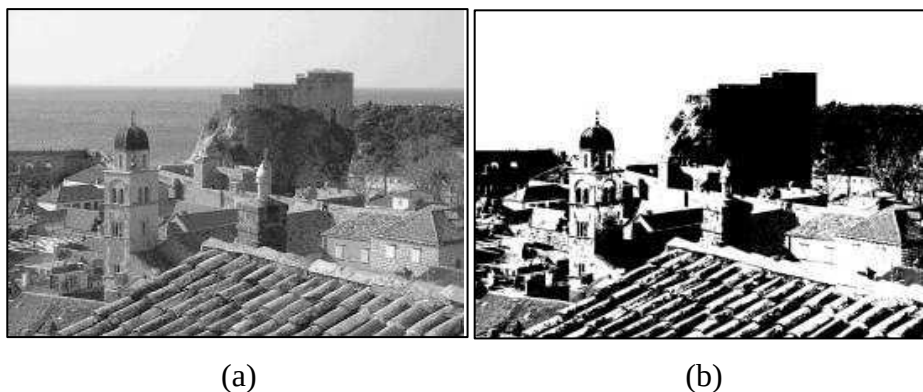


Fig.I.5. Binarisation d'une image (a) image originale, (b) image binarisée.

I.9.2. 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].

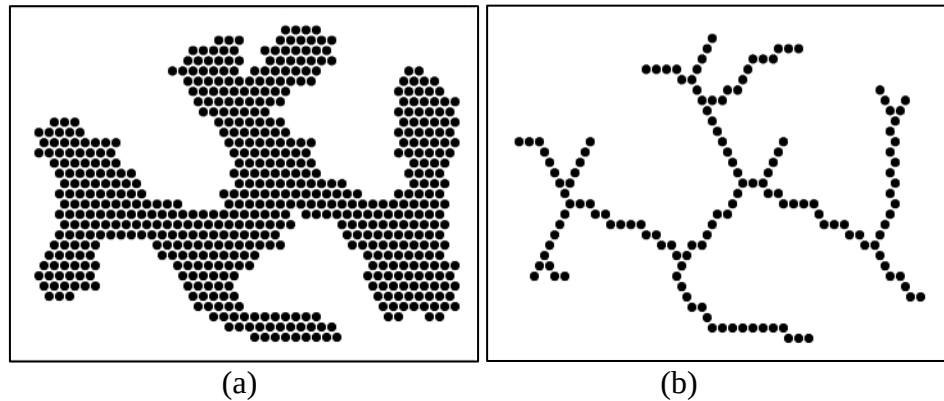


Fig.I.6. *Squelettisation d'une image (a) image originale, (b) image squelettisée.*

I.9.3. 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.9.4. 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 :

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

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.9.5. 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.9.5.1. 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.3. Masque de convolution 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.4. Masque de convolution 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 :

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

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

$\frac{1}{6}$	×	<table> <tr> <td>1</td><td>2</td><td>1</td></tr> <tr> <td>2</td><td>4</td><td>2</td></tr> <tr> <td>1</td><td>2</td><td>1</td></tr> </table>	1	2	1	2	4	2	1	2	1
1	2	1									
2	4	2									
1	2	1									

Tab.I.5. Masque de convolution Gaussien.

I.9.5.2. 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.

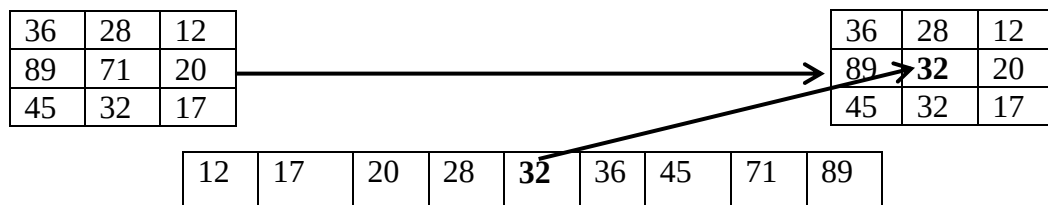


Fig.I.7. Principe du filtre médian.

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.9.6. 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.9.7. 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.6) 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 <table><tr><td colspan="2">Gradient selon x</td></tr><tr><td>-1</td><td>0</td></tr><tr><td>0</td><td>1</td></tr></table> <table><tr><td colspan="2">Gradient selon y</td></tr><tr><td>0</td><td>-1</td></tr><tr><td>1</td><td>0</td></tr></table>	Gradient selon x		-1	0	0	1	Gradient selon y		0	-1	1	0	$G_1 ; G_2$ Substitution du pixel supérieur gauche	$A = \sqrt{G_1^2 + G_2^2}$	$\theta = \frac{\pi}{4} + \arctan\left(\frac{G_2}{G_1}\right)$												
Gradient selon x																											
-1	0																										
0	1																										
Gradient selon y																											
0	-1																										
1	0																										
Masques de Sobel <table><tr><td colspan="3">Gradient selon x</td></tr><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> <table><tr><td colspan="3">Gradient selon y</td></tr><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>	Gradient selon x			1	0	-1	2	0	-2	1	0	-1	Gradient selon y			1	2	1	0	0	0	-1	-2	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
Gradient selon x																											
1	0	-1																									
2	0	-2																									
1	0	-1																									
Gradient selon y																											
1	2	1																									
0	0	0																									
-1	-2	-1																									
Masques de Prewitt <table><tr><td colspan="3">Gradient selon x</td></tr><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> <table><tr><td colspan="3">Gradient selon y</td></tr><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>	Gradient selon x			1	0	-1	1	0	-1	1	0	-1	Gradient selon y			1	1	1	0	0	0	-1	-1	-1	$G_1 ; G_2$	$A = \sqrt{G_x^2 + G_y^2}$	$\theta = \arctan\left(\frac{G_y}{G_x}\right)$
Gradient selon x																											
1	0	-1																									
1	0	-1																									
1	0	-1																									
Gradient selon y																											
1	1	1																									
0	0	0																									
-1	-1	-1																									
Masque Laplacien <table><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.6 : les différents types de masques de convolution utilisée pour la détection du contour d'une image.



Fig.I.8. Détection des contours d'une image par convolution de type Sobel.

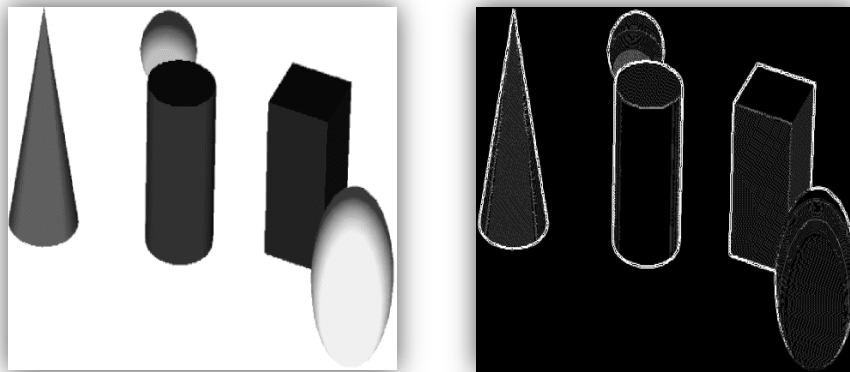


Fig.I.9. Détection des contours d'une image par convolution de type Laplacien.

I.10. 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.

Chapitre II

Classification des Empreintes Digitales

II.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. Loin des considérations probabilistes, elle est, de nos jours, le seul moyen d'identification formelle d'une personne. 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 sous ensemble de la base des empreintes et donc réduire le temps de recherche.

II.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 [8].

II.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.II.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) [9].

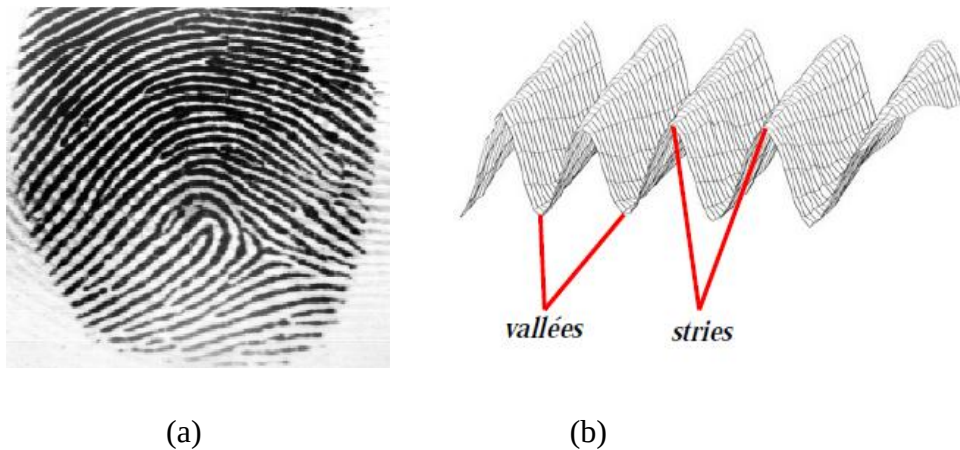


Fig.II.1. *L'empreinte digitale (a) Exemple d'une empreinte digitale (b) vallées et stries.*

II.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.

II.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 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. II.2).

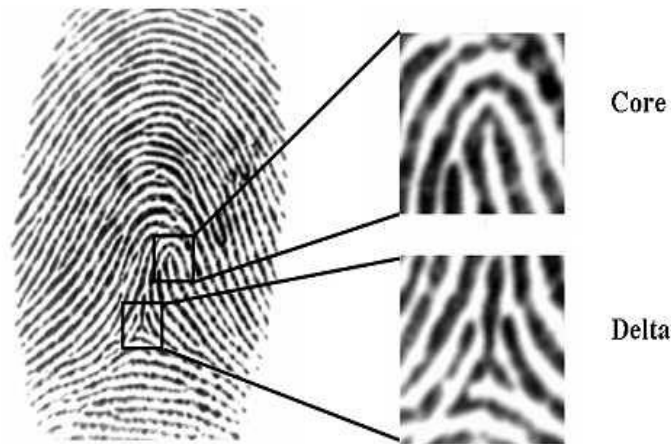


Fig. II.2. Les points singuliers globaux, core et le delta.

II.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.II.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))

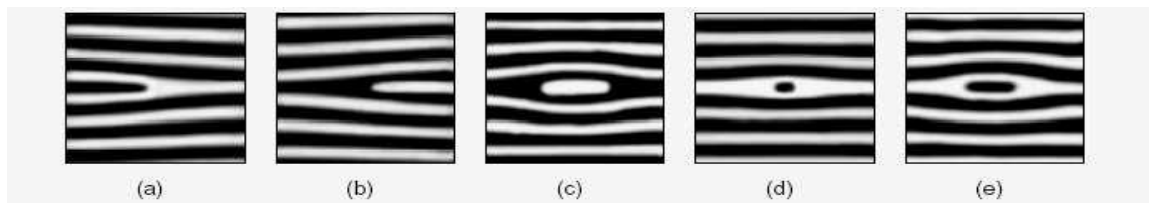


Fig. II.3. Les différents types de minuties.

II.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. [9]



a. Arc

b. Arc tendu

c. Boucle à gauche



d. Boucle à droite

e. Spire

Fig. II.4. Les cinq classes des empreintes digitales [14].

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 [10].

II.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 piqure), ni contraignante pour l'utilisateur (juste le doigt à appliquer sur un capteur).

II.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.II.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.

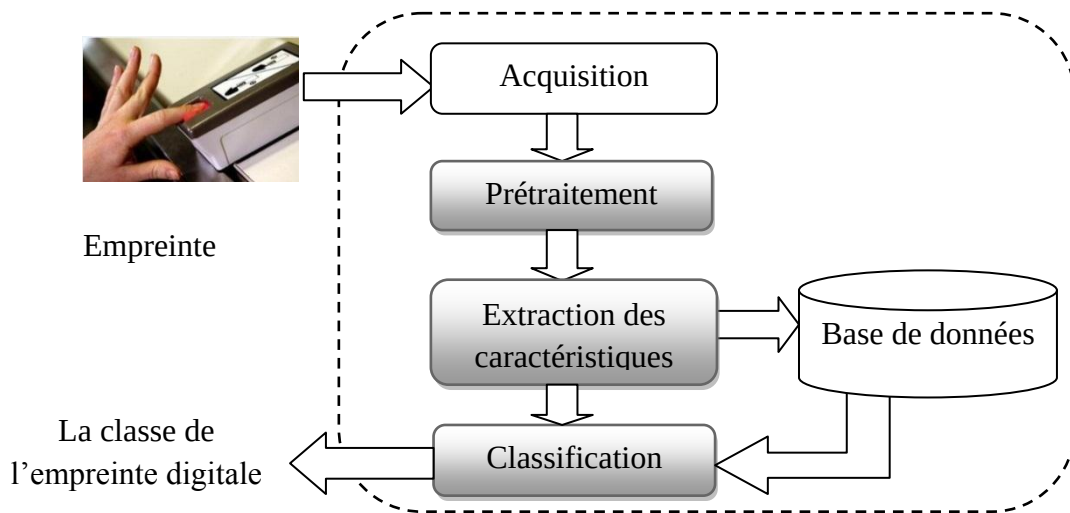


Fig.II.5. Schémas générale d'un système de classification des empreintes digitales.

II.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 travail 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 vide 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 [11].

II.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.

II.7.3. L'extraction des caractéristiques

La très grande majorité des techniques de classification des empreintes digitales sont basées 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 :

II.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 définie par :

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

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

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

L'image originale est normalisée en utilisant sa valeur moyenne $M(I)$ 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) .

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

M_0 et VAR_0 sont les valeurs de moyenne et de variance désirées. [12]

II.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.II.6) [13].

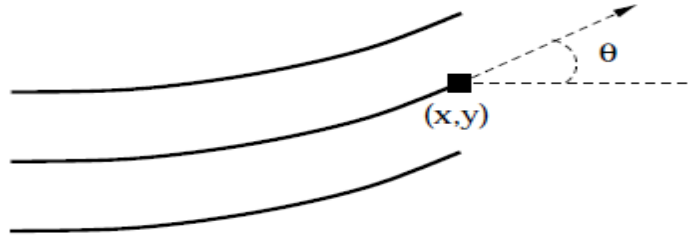


Fig.II.6. L'orientation de la crête locale du pixel(x, y).

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:

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

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

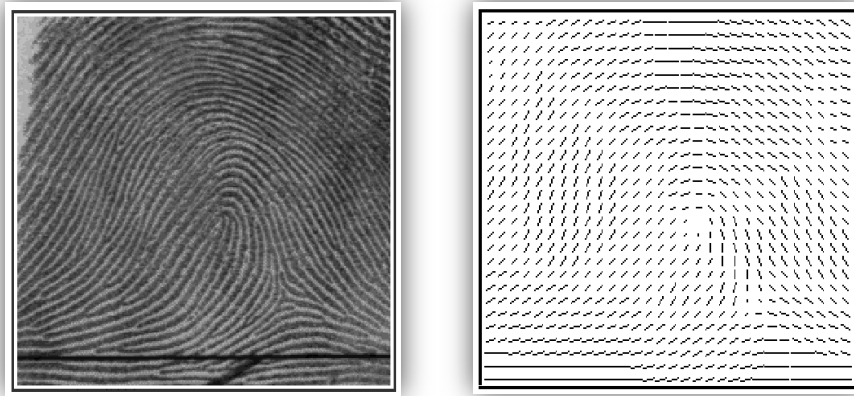
$$O(i, j) = \frac{V_x(i, j)}{V_y(i, j)} \quad (6)$$

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

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

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

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



(a) image original

(b) image d'orientation

Fig.II.7. L'image d'orientation d'une empreinte digitale [15].

II.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.II.8).

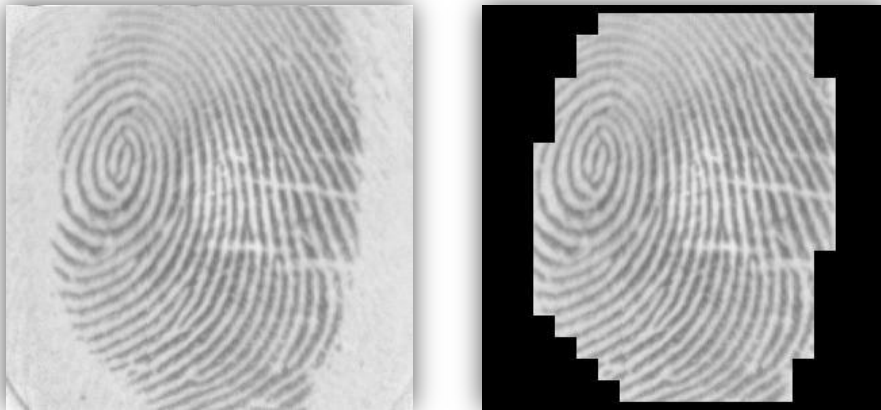


Fig.II.8. Effet de la segmentation sur l'image (a) Image d'entrée (b) Image segmentée [16].

II.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 ce mémoire, on utilise l'un des plus pratiques, simples et élégants: c'est l'algorithme de l'index de Poincaré proposé par Kawagoe et Tojo [17].

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.II.9).

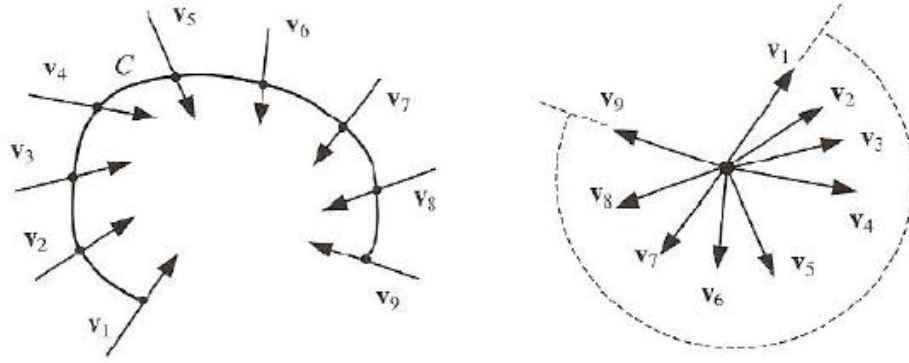


Fig.II.9. L'index de Poincaré calculé sur une courbe C [15].

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 [21].

La figure (Fig.II.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.

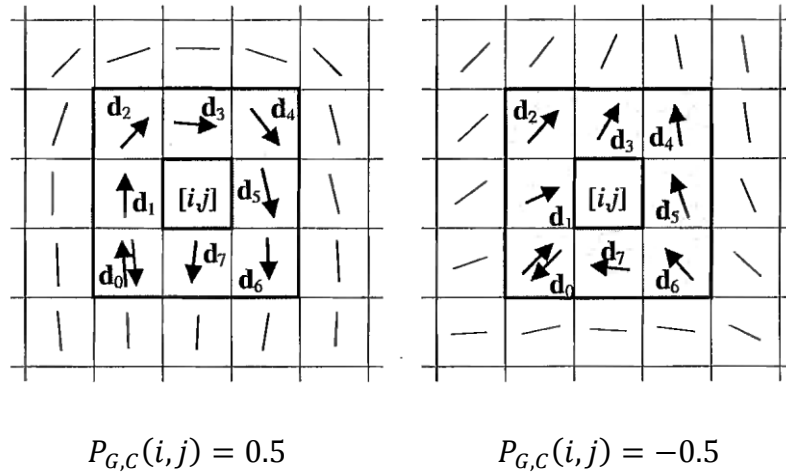


Fig.II.10. L'index de Poincaré calculé avec un voisinage de $N=8$ pixels autour du pixel $[i, j]$ [15].

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

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

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

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

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 [18].

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.II.11) illustre un exemple de détection des points singuliers globaux par l'algorithme de l'index de Poincaré.

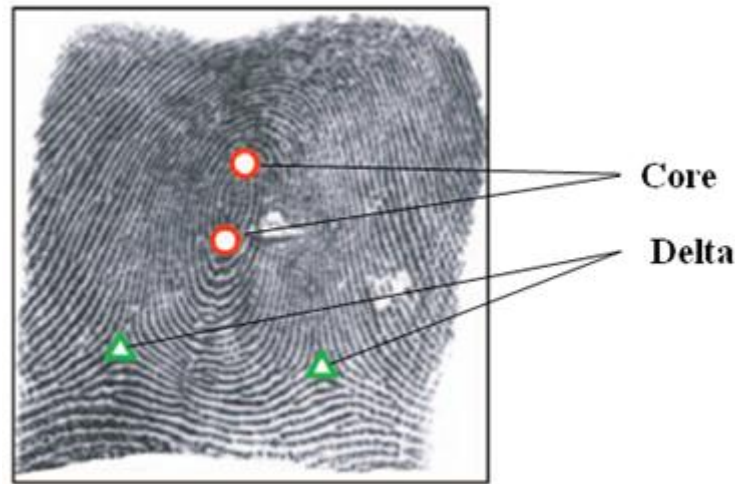


Fig.II.11. Résultat de détection des points singuliers globaux d'une image d'empreinte digitale spire par l'algorithme de l'index de Poincaré.

II.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 auxquelles appartient cette empreinte digitale.

II.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.

II.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 adoptées 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 $[r.\cos 2\theta, r.\sin 2\theta]$, par exemple une image d'orientation de (30 x 30) donne un vecteur de 1800 éléments, donc il 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 [15].

II.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 élément dans chaque région (Fig.II.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 exploité [10].

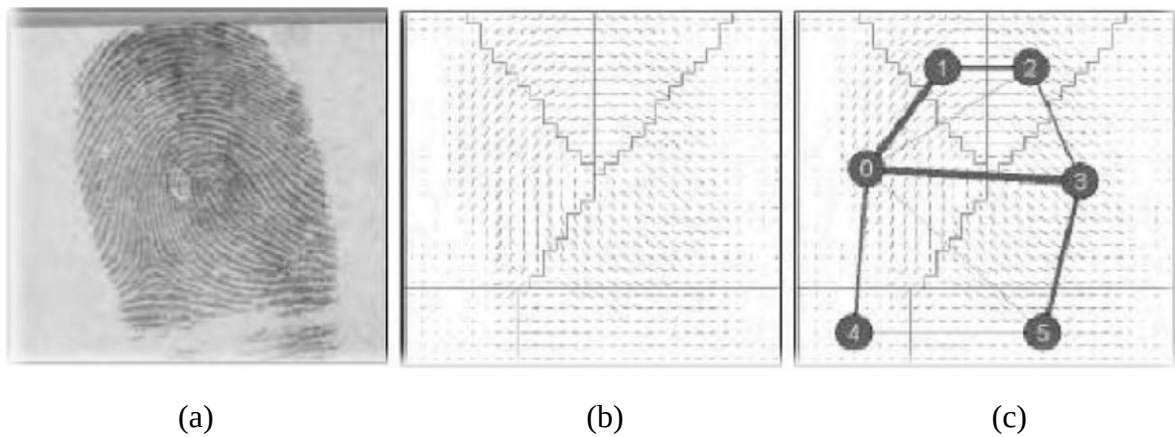


Fig.II.12. (a) image d'empreinte digitale, (b) partitionnement de son image directionnelle, (c) le graphe relationnel correspondant [10].

II.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 [19].

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.II.13). Ces lignes sont étiquetées conformément aux changements de direction, obtenant ainsi un ensemble de chaînes de caractères qui sont traitées par des grammaires [15].

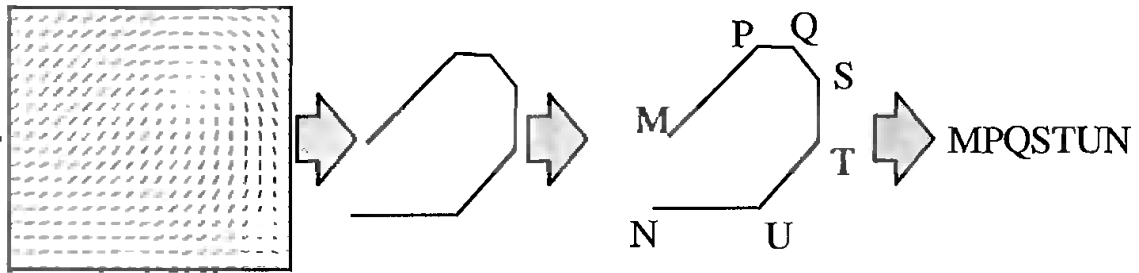


Fig.II.13. Un schéma de l'approche chaîne de la construction de Rao et Balck 1980.

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.

II.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 d'entrées du réseau. Dans [20], Kamijo présente une architecture pyramidale constitué 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 [10].

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.

II.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
Arc	0	0
Arc tendu	1	1(milieu)
boucle à gauche	1	1(gauche)
boucle à droit	1	1(droit)
Spire	2	2

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

II.9. 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.

Chapitre III

Réseaux de
Neurones Flous

III.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.

III.2. Notions fondamentales sur les réseaux de neurones

III.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 solutionner les problèmes complexes.

III.2.2. Neurone biologique

Le neurone biologique est le composant cellulaire de base du cerveau (voir Fig.III.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.

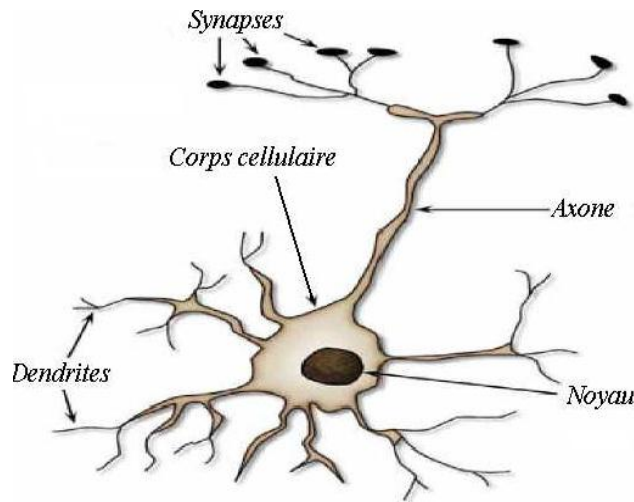


Fig.III.1. Neurone biologique [22].

III.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.III.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 [22] [23].

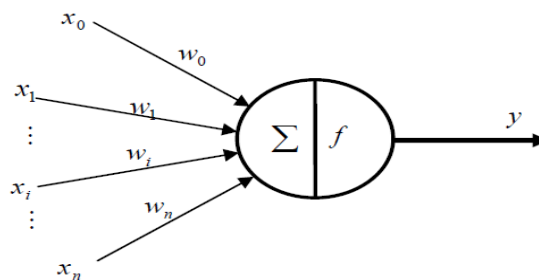


Fig.III.2. Neurone artificiel [24].

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

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

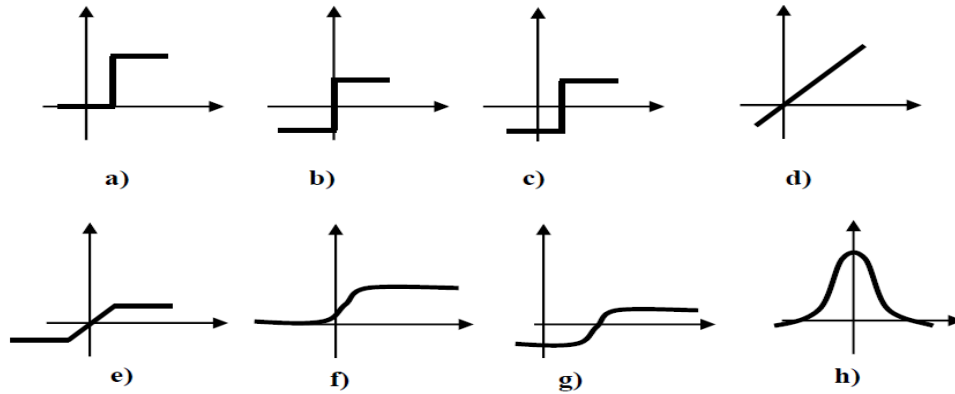


Fig.III.3. fonctions d'activations les plus utilisées. a) tout ou rien ; b) fonction signe ; c) plus ou moins à seuil ; d) fonction affine ; e) saturation ; f) sigmoïde ; g) fonction arc tangente ; h) fonction gaussienne. [24].

III.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 [25].

III.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 [26] 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 « professeur », 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 correcte [22].

III.2.4.2. Les méthodes d'apprentissage [22] [27]

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 sorties binaires à des sorties 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.

III.2.4.3. Les réseaux de neurones les plus utilisés [28]

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.III.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.

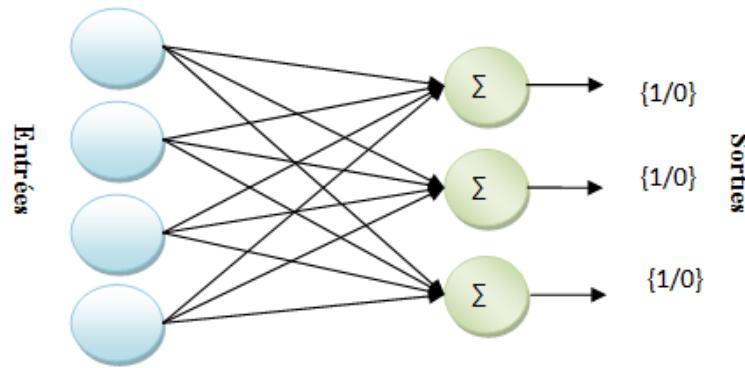


Fig.III.4. Schéma général de perceptron simple.

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.III.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 illustré, 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.

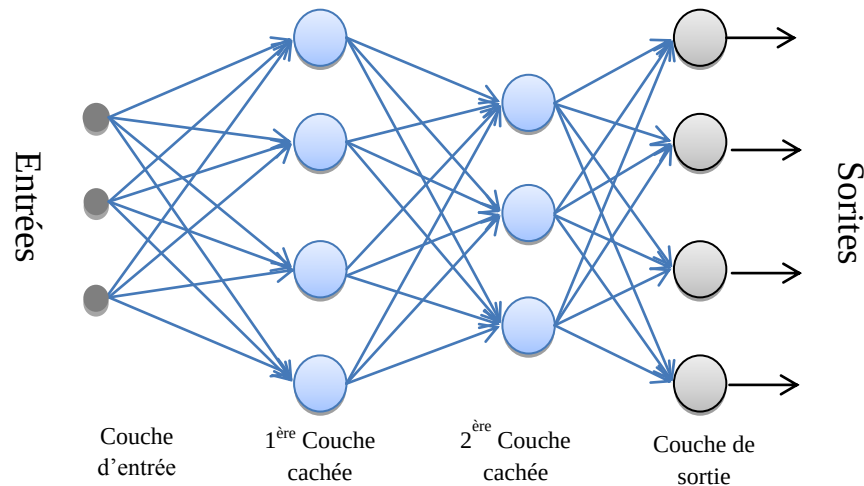


Fig.III.5. Exemple de réseau de type perceptron Multicouche.

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 comportant 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.III.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).

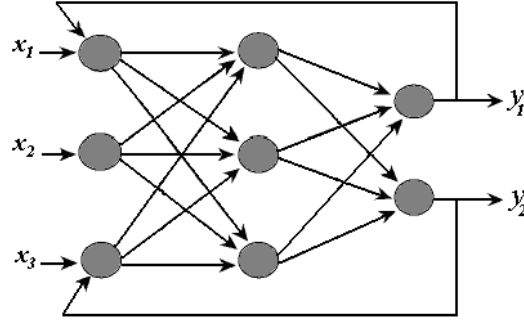


Fig.III.6. Réseaux de neurones bouclés.

III.2.5. La rétro-propagation du gradient de l'erreur [22]

L'algorithme de rétro-propagation du gradient de l'erreur a été créé en généralisant les règles d'apprentissages de Widrow-Hoff [26], 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.

III.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.

III.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 :

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

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

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

4. Accumulation des erreurs en sortie :

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

- 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 :

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

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

- Pour chaque cellule cachée :

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

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

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

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

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.

III.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 :

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

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 passée 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.

III.2.5.4. Conditions d'arrêt de l'apprentissage [29]

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.

III.2.5.5. Avantages et inconvénients [22]

L'algorithme de rétro-propagation de gradient a des avantages et des inconvénients :

➤ **Avantages**

- Ce fût un des premiers algorithmes développés pour l'apprentissage des réseaux de neurones multicouches. Il permet de pallier une carence 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 est gourmand en temps de calcul ;
- 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.

III.2.6. Avantages et Inconvénients des réseaux de neurones

III.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. Aucune compétence en mathématiques, informatique statistique requise ;
- 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.

III.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 *boîtes noires* où les connaissances sont inintelligibles pour l'utilisateur [22].

III.3. Les Réseaux de Neurones Flous

III.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 Zahed 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 [30].

III.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 :

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

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

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

III.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.III.7) illustre un exemple de la variable linguistique 'vitesse' avec trois termes linguistiques: petite, moyenne et grande.

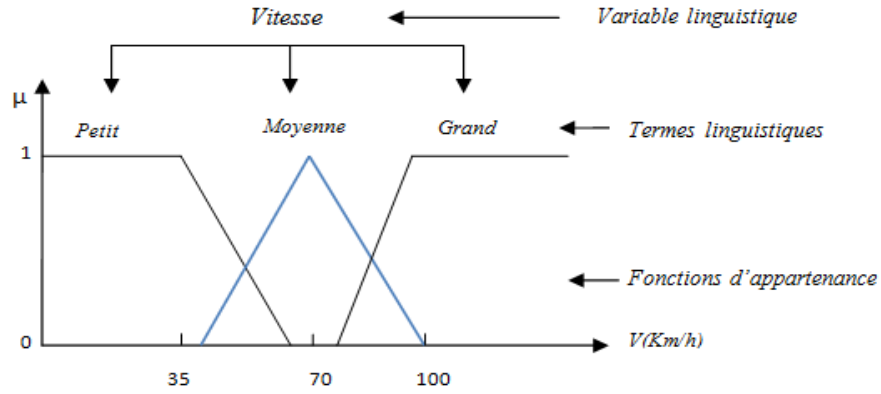


Fig.III.7. Variable linguistique.

III.3.1.3. Les fonctions d'appartenance [24]

On peut utiliser pour les fonctions d'appartenance des formes différentes (Fig.III.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.

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

b. Fonction trapézoïdale

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

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

c. Fonction sigmoïde

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

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

d. Fonction gaussienne

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

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

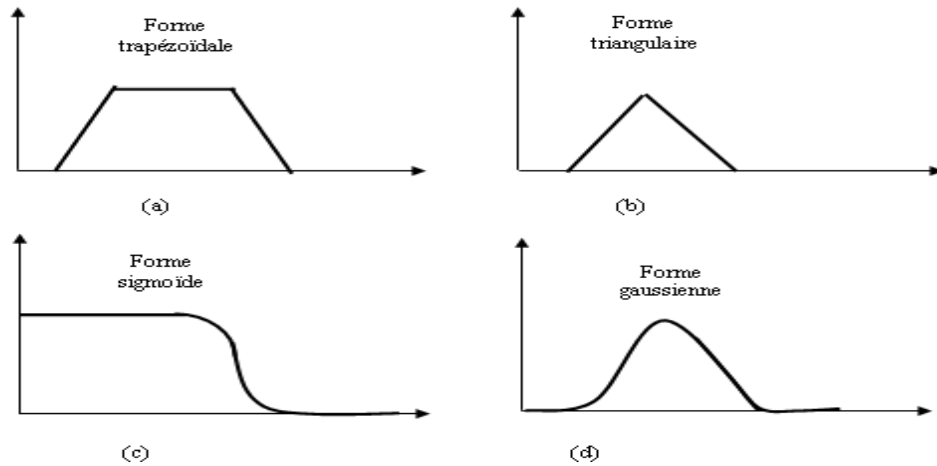


Fig.III.8. Formes des fonctions d'appartenance usuelles [24].

III.3.1.4. Règles d'inférence

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*

III.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.III.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 [28].

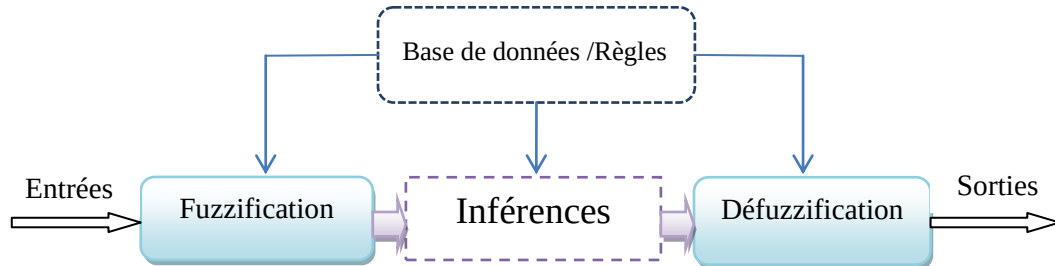


Fig.III.9. Structure d'un système d'inférence floue.

- **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 [25].

- **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 conclusion des règles qui peut prendre plusieurs formes. Donc, on peut dire que l'inférence floue est l'opération d'agrégation des règles floues [28].

- **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. À partir des degrés d'appartenance, on obtient autant de valeurs qu'il y a d'états [32].

III.3.2. Systèmes Neuro-Flous

Les réseaux de neurones (RN) multicouches constituent des approximateurs universels grâce à leur capacité 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 tarder 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 [24].

III.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 [32].

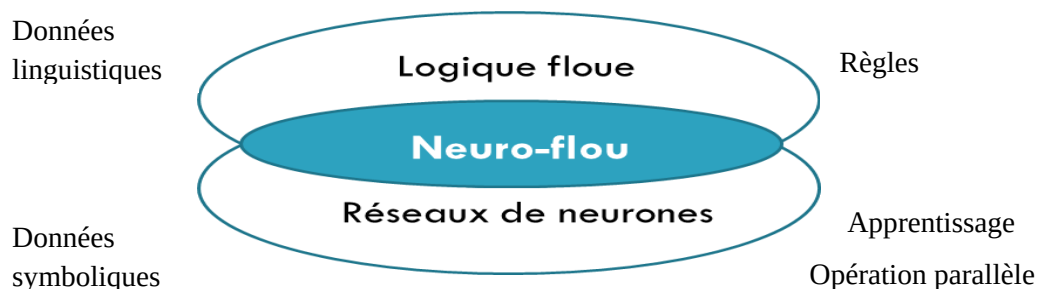


Fig.III.10. principe du système neuro-flou.

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

III.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) [28].

III.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.

III.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.

III.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.

III.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.

III.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.

III.3.4. Les types des réseaux de neurones flous utilisés pour la classification [33]

Plusieurs architectures ont été proposées dans la littérature. Parmi ces architectures, il y a le réseau de Pedrycz, le réseau fusionné compacte, le réseau NEFClass :

III.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.III.11).

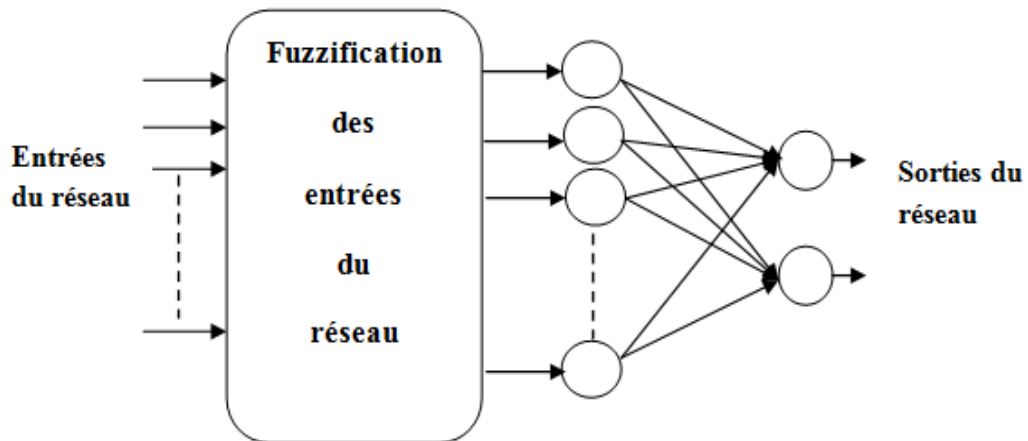


Fig.III.11. Architecteur de réseau de Pedrycz.

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.

III.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.III.12).

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

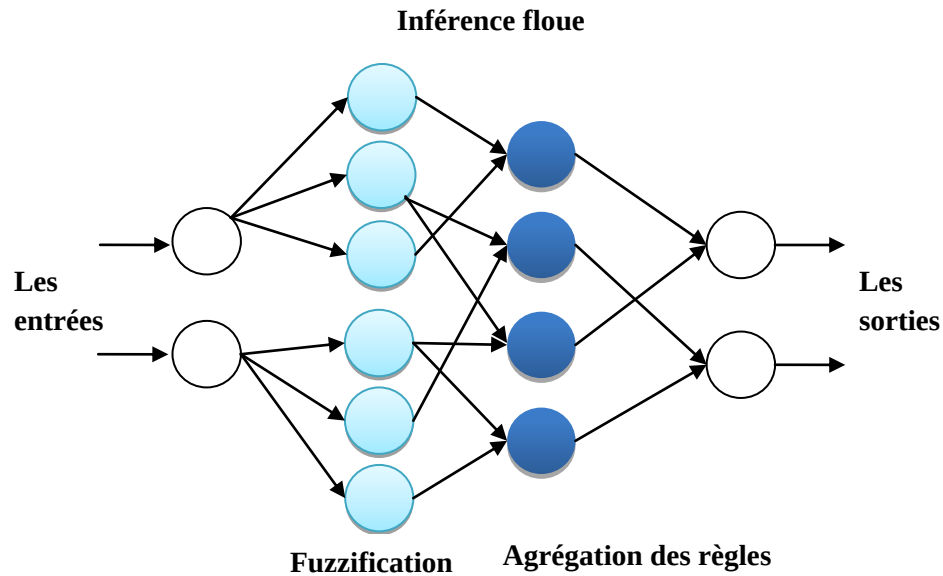


Fig.III.12. Architecte de réseau fusionnée compacte.

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

Les systèmes NEFClass (**NEuro Fuzzy CLASSification**) 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.III.13).

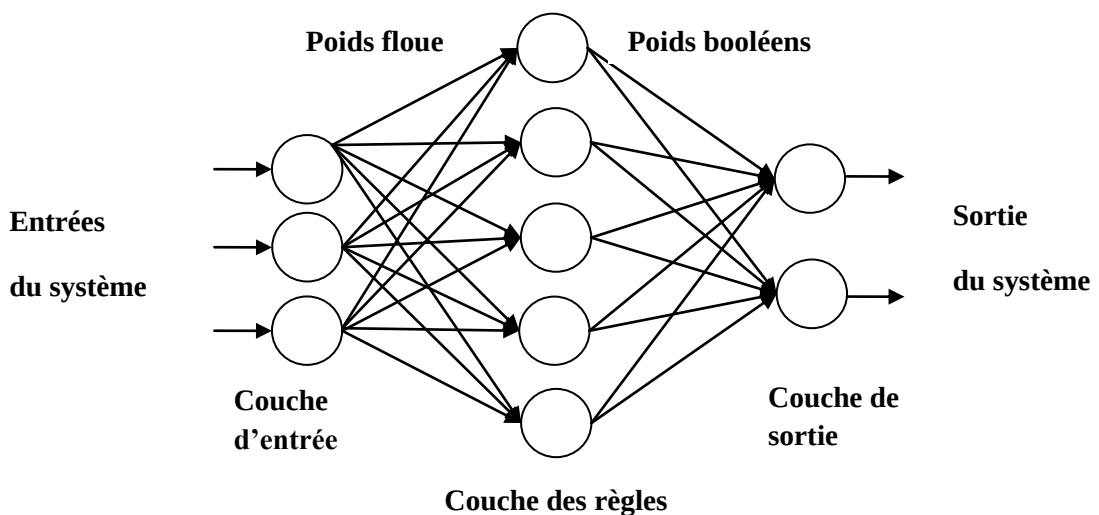


Fig.III.13. Architecte d'un système NEFClass.

III.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.

Chapitre IV

Conception et réalisation

IV.1. Introduction

Dans ce chapitre, nous allons présenter notre travail qui consiste à concevoir et à réaliser un système de 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 présenter les résultats concernant l'extraction des caractéristiques, ainsi que les résultats qu'on a obtenus pour les réseaux de neurones artificiels et les réseaux de neurones flous qu'on a optés.

IV.2. Plateforme et outils de développement

Pour l'implémentation de notre application, nous avons opté pour le langage C# en utilisant l'EDI (Environnement de développement intégré) Microsoft visual studio 2010 sous Windows 7, la figure (Fig.IV.1) présente l'interface de Microsoft visual studio 2010.

Pour la conception des bases de données, on a utilisé le système de gestion de base de données Access 2010.

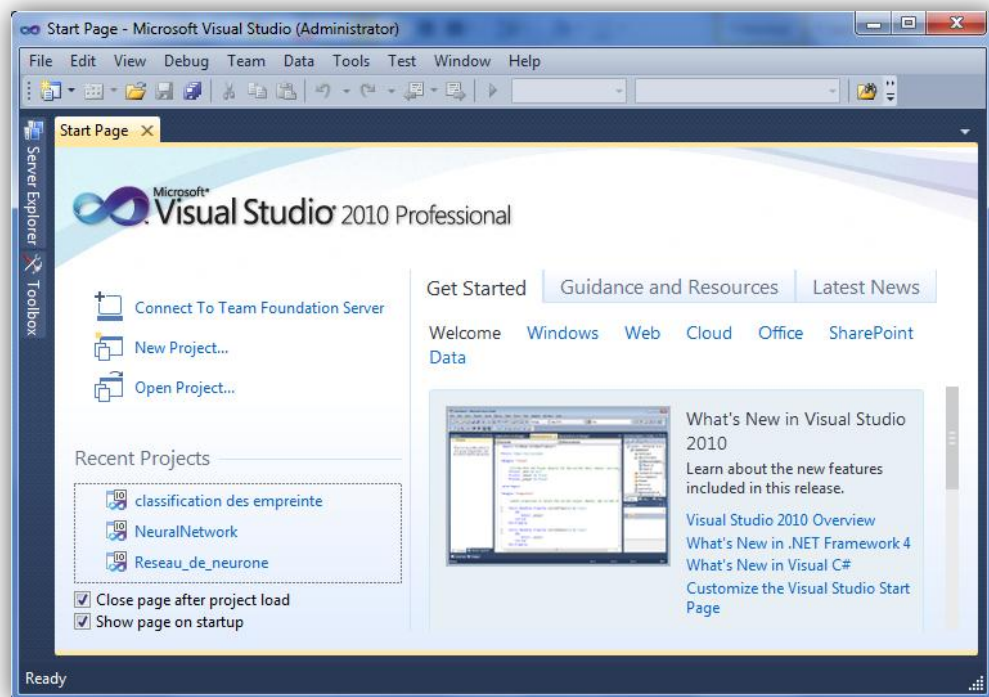


Fig.IV.1. *L'interface de Microsoft Visual Studio 2010.*

IV.3. Présentation et implémentation du système

Le système qu'on a réalisé est constitué de plusieurs modules, chacun correspond à une tâche spécifique. Il a pour but de classifier les empreintes digitales en cinq classes : Arc, Arc tendu, Boucle à gauche, Boucle à droite et Spire.

IV.3.1. Système de classification d'empreintes digitales

La structure de notre système est présentée par la figure suivante (Fig.IV.2) :

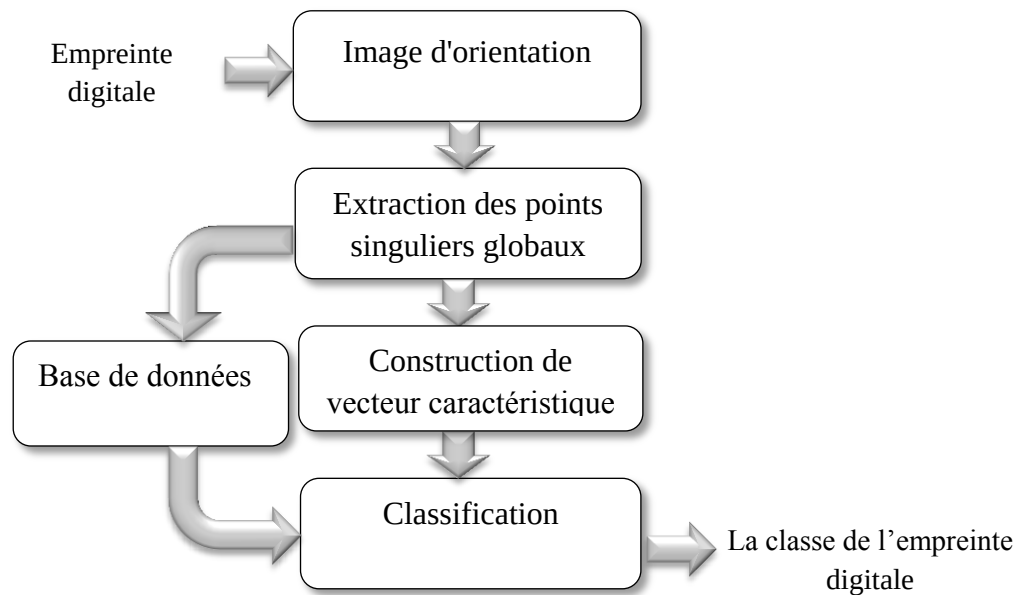


Fig.IV.2. Structure de système de classification des empreintes digitales.

IV.3.1.1. La base de données des images utilisées

Les images d'empreintes digitales qu'on a utilisé sont extraite de la base de données CASIA Fingerprint Image Version 5.0 (ou CASIA-FingerprintV5), qui contient 20,000 images d'empreintes digitales. 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 inclus 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.

Dans 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 notre modèle de classification, elle est constituée de 400 empreintes digitales, alors que la deuxième est pour l'évaluation des performances celui-ci et qui comporte 200 empreintes.

IV.3.1.2. 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 (II.7.3.2, page 27), la figure (Fig.IV.3) illustre les résultats du gradient du Sobel et l'image d'orientation de l'empreinte digitale.

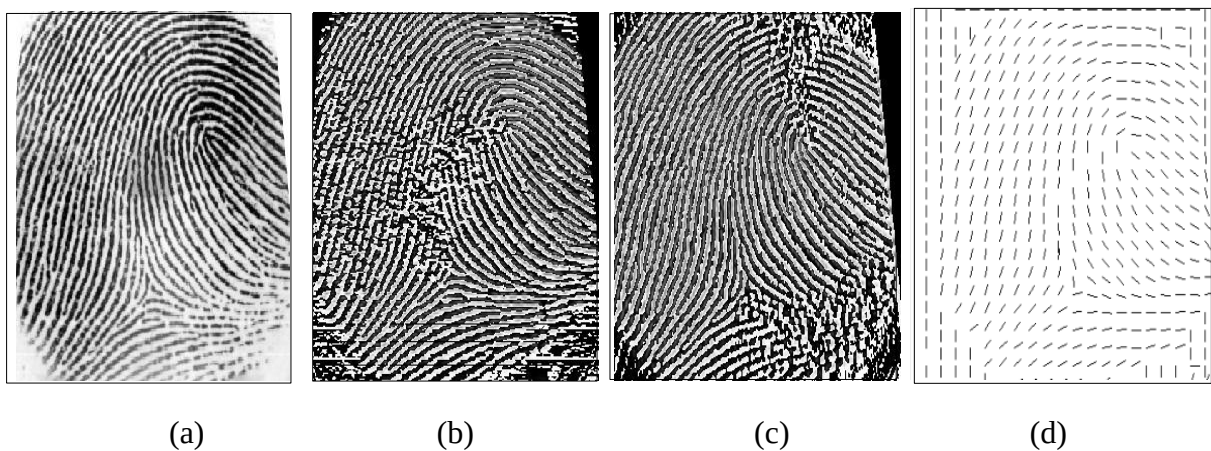


Fig.IV.3. L'extraction de l'image d'orientation (a) L'image originale, (b) Le gradient selon x , (c) Le gradient selon y , (d) l'image d'orientation.

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

IV.3.1.3. Extraction des points singuliers globaux

Pour extraire les points singuliers globaux de l'empreinte digitale, il y a deux étapes à réaliser : la détection des points singuliers globaux et l'élimination des faux cores et deltas par la segmentation :

1. 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 II.7.3.4, page 28).

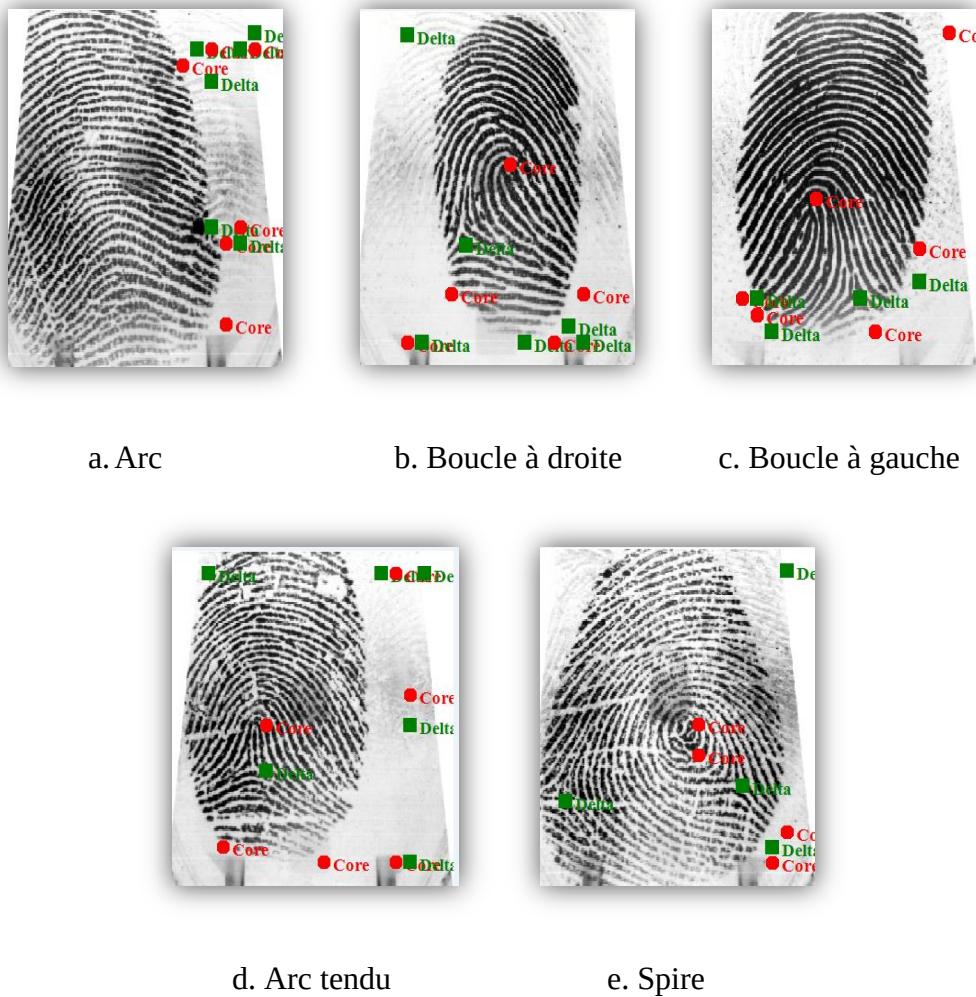


Fig.IV.4. Le résultat de détection des points singuliers par l'algorithme de Poincaré avant la segmentation pour les différentes classes.

Cet algorithme nous donne de bons résultats pour la détection des points singuliers globaux (voir Fig.IV.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.

2. Elimination des faux cores et deltas par la segmentation

Afin d'éliminer les faux cores et deltas, 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 (II.7.3.3, page 28) du deuxième chapitre. Les résultats de cette étape sont très satisfaisants.

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

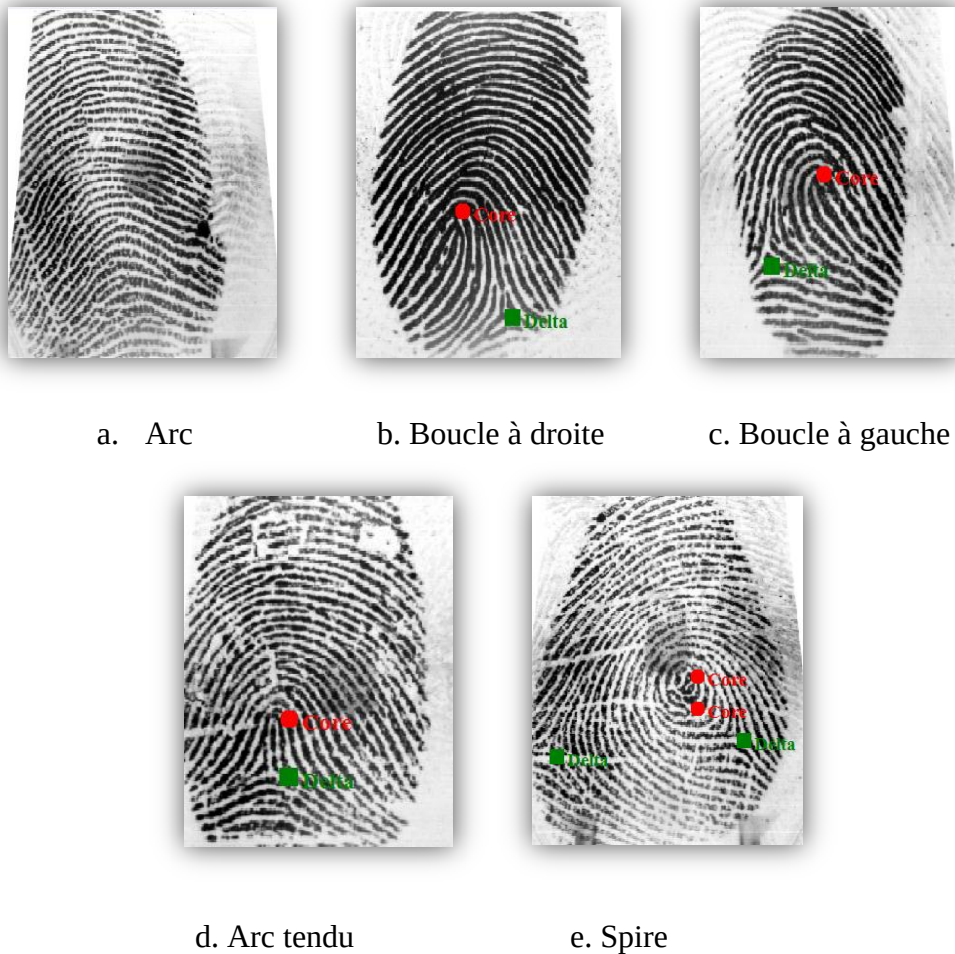


Fig.IV.5. *Le résultats de détection des points singuliers après la segmentation pour les différentes classes.*

IV.3.1.4. Le codage des données

1. Le codage des variables d'entrées du système

Le vecteur d'entrée est constitué des 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 varie de 0 à 2. La troisième composante de vecteur caractéristique qui est la position de delta par rapport au core, est codée comme suit (voir Tab.IV.1) :

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

Tab.IV.1. Le codage de variables d'entrées.

2. Le codage de variable de sortie du système :

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

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

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

Tab.IV.2. Le codage binaire de variable de sortie.

IV.3.1.5. Classification

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

A. Réseaux de Neurones Artificiels

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.IV.6).

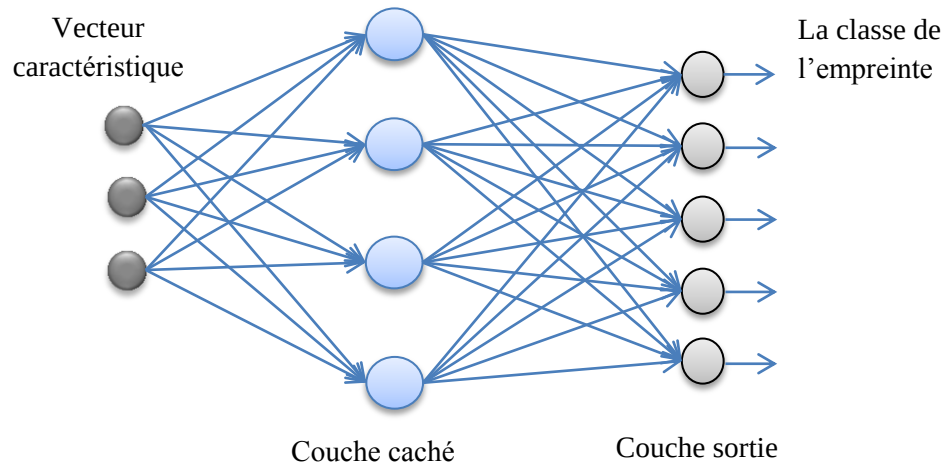


Fig. IV.6. Structure de réseaux de neurones artificiels utilisés.

B. Réseaux de Neurones Flous

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. IV.7) :

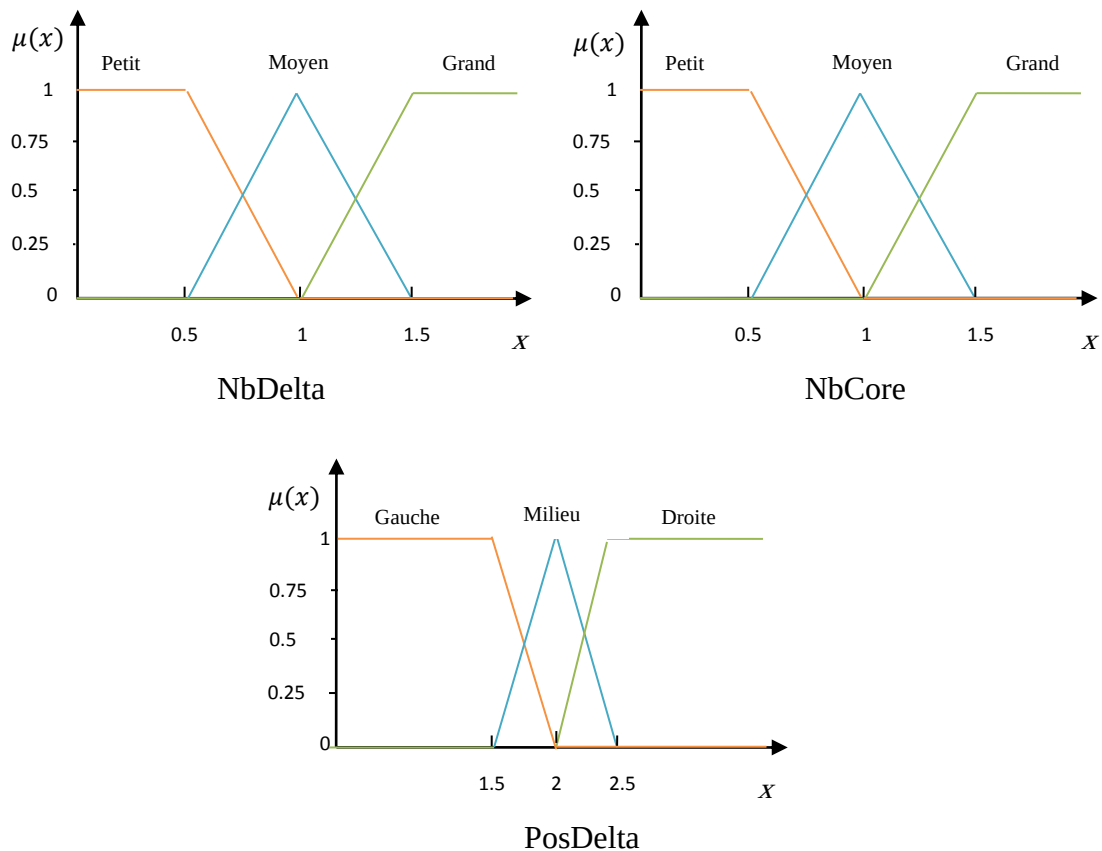


Fig. IV.7. Les fonctions d'appartenance des variables d'entrée du système.

Pour notre système on a utilisé les sept règles suivantes :

Si NbCore est *petit* et NbDelta est *petit* **Alors** Class est Arc.

Si NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *droite* **Alors** Class est Boucle à gauche.

Si NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *gauche* **Alors** Class est Boucle à droite.

Si NbCore est *moyen* et NbDelta est *moyen* et PosDelta est *milieu* **Alors** Class est Arc tendu.

Si NbCore est *grand* et NbDelta est *moyen* **Alors** Class est Spire.

Si NbCore est *grand* et NbDelta est *petit* **Alors** Class est Spire.

Si NCore est *grand* et NDelta est *grand* **Alors** Class est Spire.

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

• Le réseau fusionné compact :

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

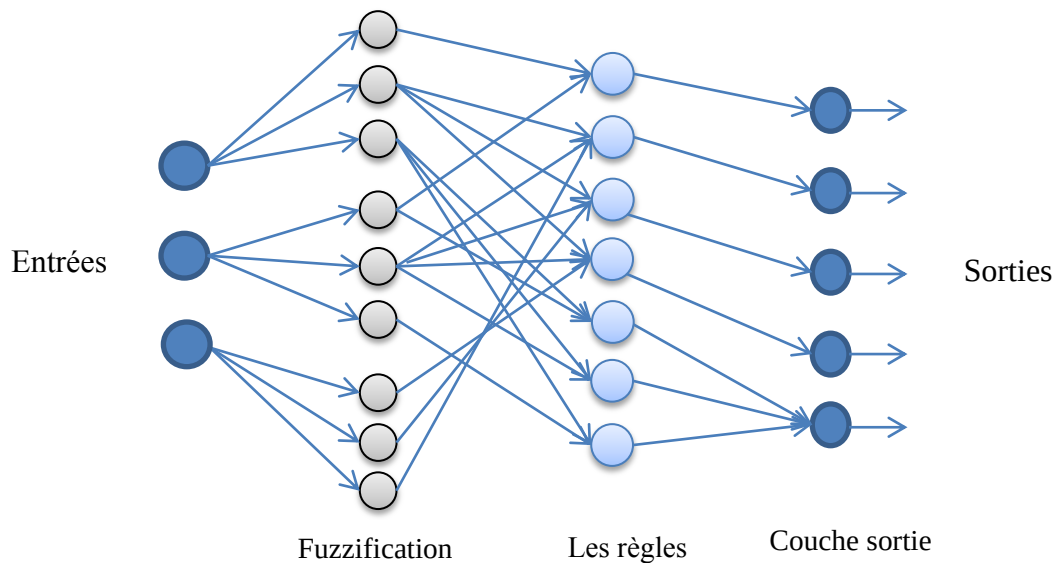


Fig.IV.8. Réseau fusionné compact.

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 7 neurones qui correspondent aux 7 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.

• 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.IV.9).

Par exemple les poids reliant les 3 neurones de couche d'entrées avec le premier neurone de la couche des règles sont respectivement les valeurs des fonctions d'appartenance : $\mu_{NbCore}(Petit)$, $\mu_{NbDelta}(Petit)$, 0, ce qui nous donne la première partie de la première règle d'inférence.

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.

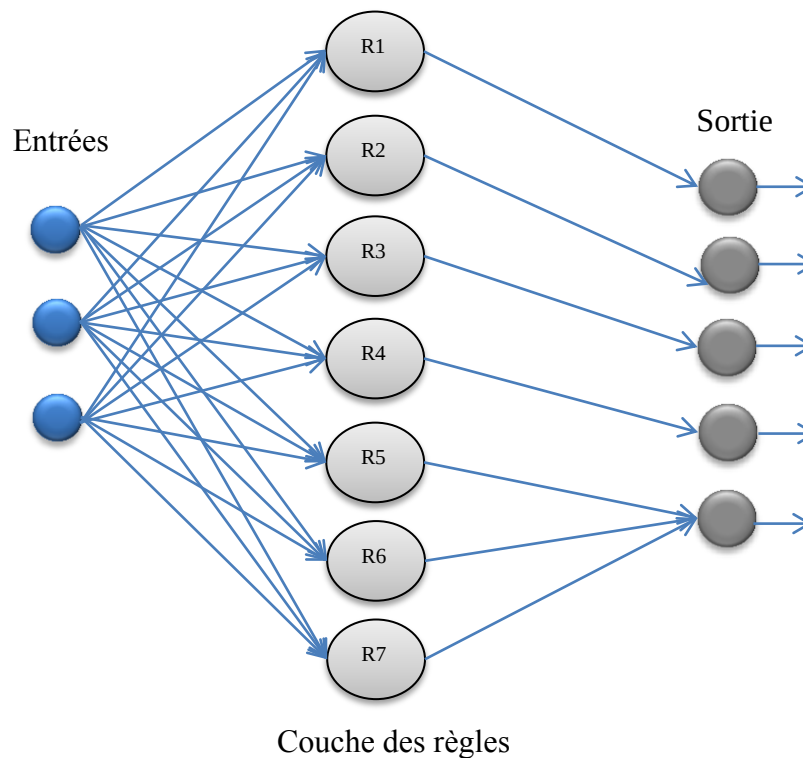


Fig. IV.9. Réseau NEFClass.

• **Le réseau de Pedrycz :**

C'est un réseau de neurones flous de trois couches (voir Fig.IV.10), la première couche contient les neurones qui correspondent aux entrées du système, la deuxième couche est la couche de fuzzification, elle possède 9 neurones qui correspondent aux sous-ensembles flous des entrées.

Les poids initiaux de la première couche ont le même raisonnement que le réseau fusionné compacte, et ceux de la deuxième couche sont des poids aléatoires, la troisième couche comporte cinq neurones.

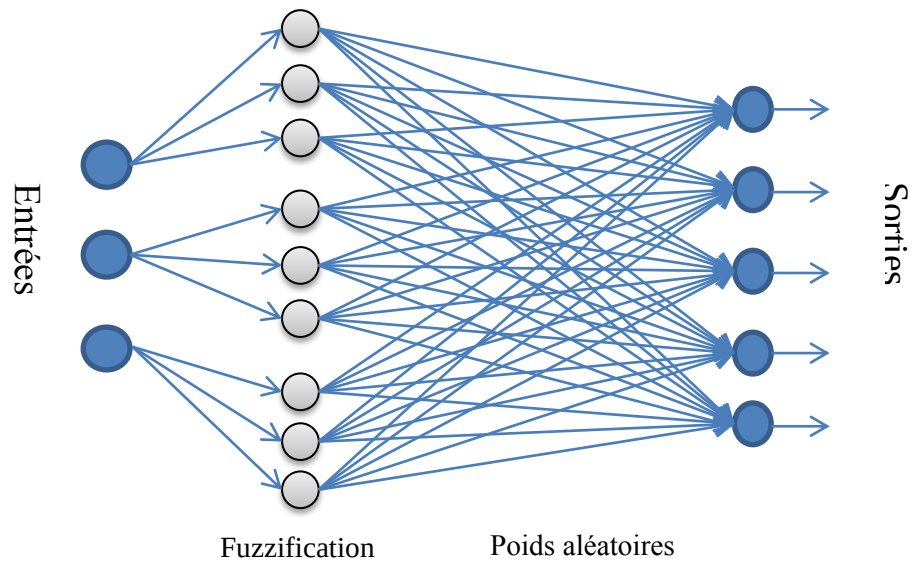


Fig. IV.10. Réseau de Pedrycz.

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

Le système	Nombre de couche	Nombre de neurones dans chaque couche
RNA	3	3
		4
		5
NEFClass	3	3
		7
		5
Compact	4	3
		9
		7
		5
Pedrycz	3	3
		9
		5

Tab.IV.3. Le nombre de couche et le nombre de neurones de chaque système.

IV.4. 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 implémenté 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.

Pour déterminer la capacité de généralisation et de prise de décision, six exemples 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 en même droiture.
- **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.

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

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.IV.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.IV.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.IV.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.IV.7. Les sorties de réseau de Pedrycz.

Pour les exemples 1, 3, 4 et 5 les décisions prise par les systèmes utilisées 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 prise par les systèmes : NEFClass, Pedrycz et fusionnée 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.

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

Le réseau de neurones	Réseau de Neurones Artificiels	Le réseau fusionné compact	Le réseau de Pedrycz	Le système NEFClass
Taux de classification	96%	100%	100%	100%

Tab.IV.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. IV.11).

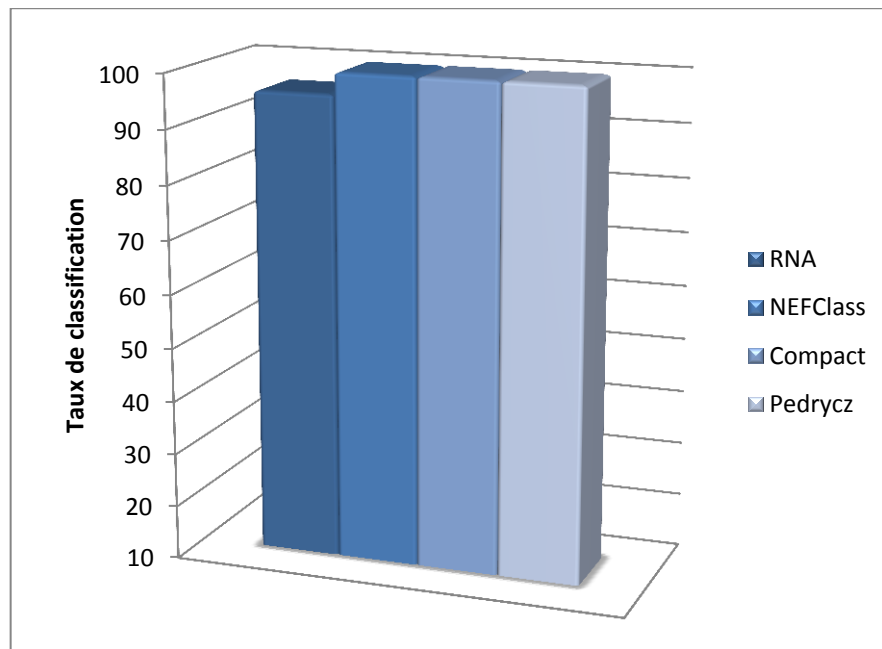


Fig. IV.11. Représentation graphique de taux de classification des différents réseaux de neurones utilisés.

Nous remarquons 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 utilisés pour la classification des empreintes digitales et que la performance des réseaux de neurones flous dépasse celle des réseaux de neurones artificiels. De plus, on peut inférer que le réseau de Pedrycz est le meilleur dans la prise de décision que les deux autres systèmes flous : NEFClass et réseau fusionné compact.

Nous constatons aussi que, 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.

IV.5. Quelques interfaces de l'application

L'interface principale de notre logiciel est présentée par la figure suivante (voir Fig. IV.12).

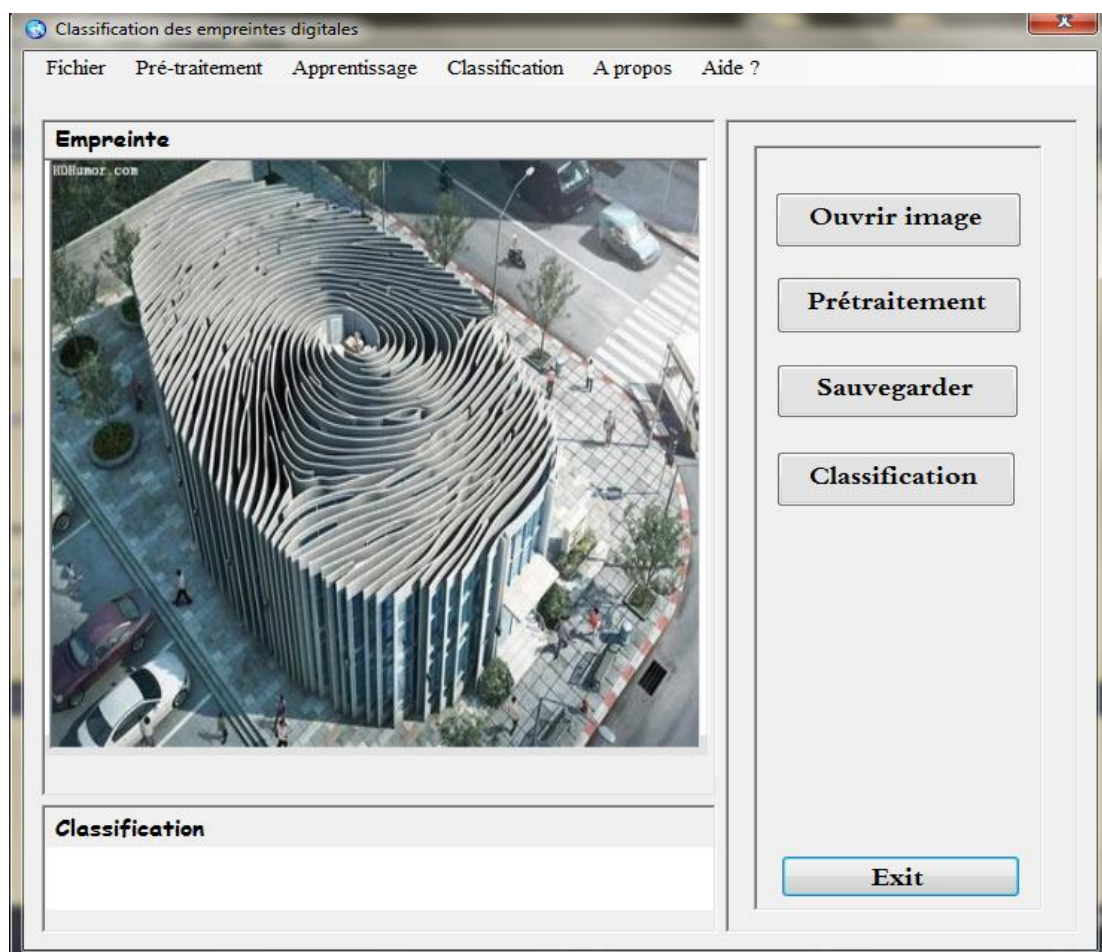


Fig. IV.12. Interface principale.

Dans le menu fichier on peut charger, sauvegarder une image d'empreinte digitale, on peut également crée une base données Access à partir d'une base d'images d'empreintes digitales (voir Fig. IV.13).

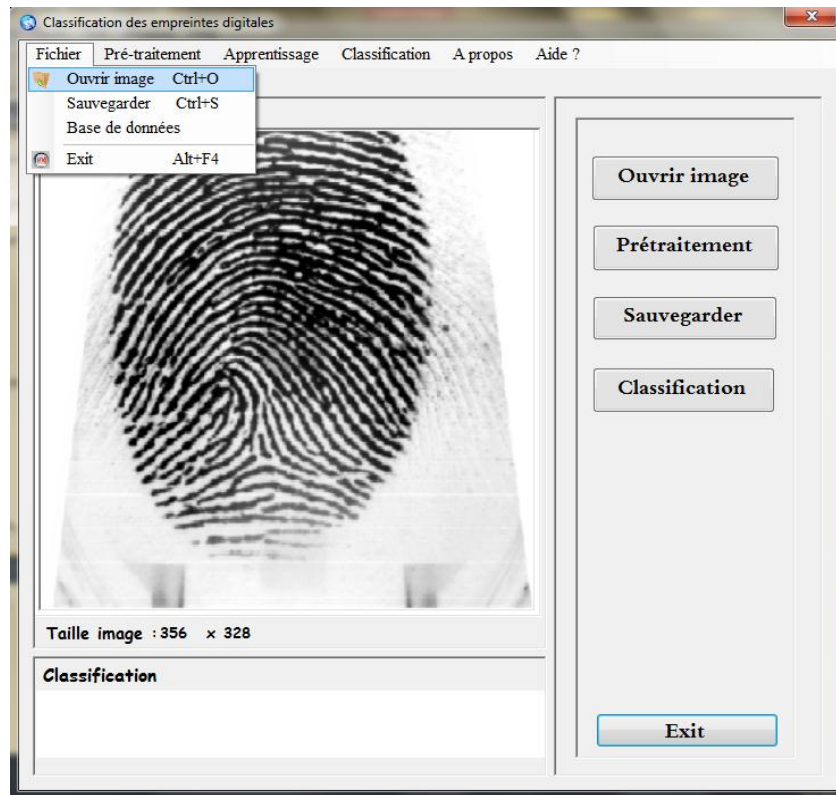


Fig. IV.13. Le menu Fichier.

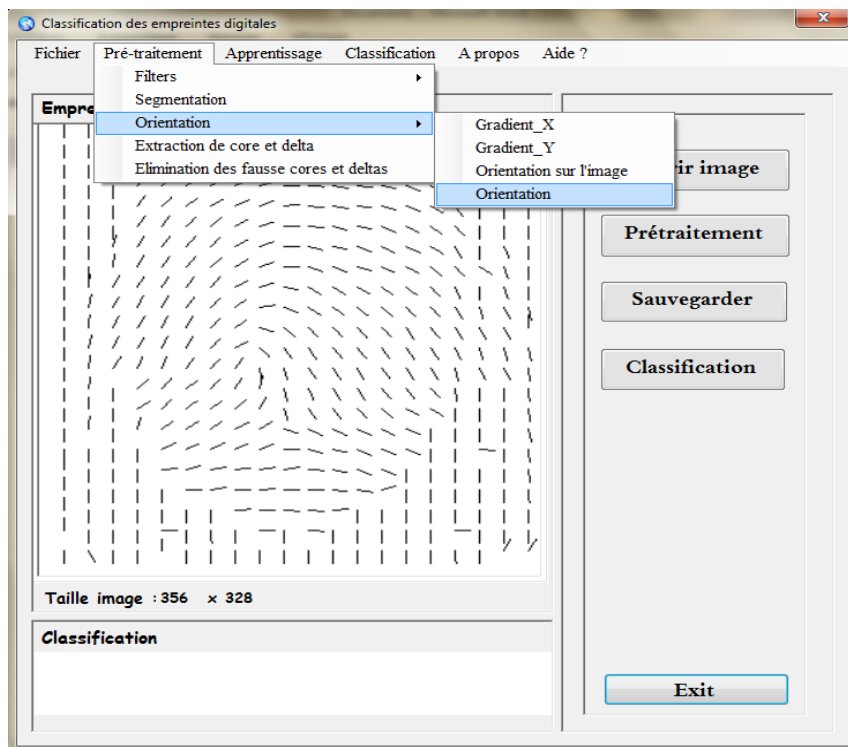


Fig. IV.14. Le menu Prétraitement.

Dans le menu Prétraitement (voir Fig. IV.14) on peut appliquer les traitements suivants sur l'image de l'empreinte digitale :

- Améliorer la qualité d'image par l'un des filtres médian ou moyenne ;
- Segmenter l'image de l'empreinte digitale ;
- Extraire l'image d'orientation et les points singuliers globaux de l'empreinte digitale ;
- Eliminer les faux cores et deltas.

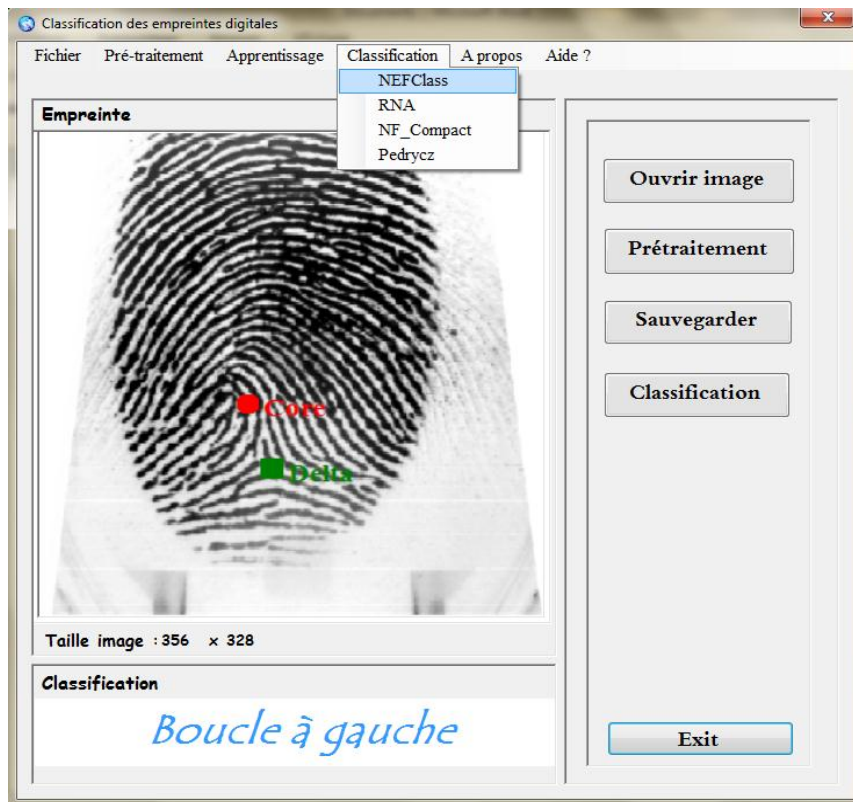


Fig. IV.15. *Le menu Classification.*

Dans le menu classification (voir Fig. IV.15) on peut sélectionner l'un des systèmes (NEFClass, Pedrycz, Réseau de neurones artificiel et Réseau fusionné compact) pour classifier l'empreinte digitale. On peut également effectuer l'apprentissage de ses systèmes dans le menu Apprentissage.

IV.6. Conclusion

Dans ce travail, nous avons 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 Henry. Nous avons 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 floue nous donne des très bons résultats.

Conclusion

Générale

Conclusion générale et perspectives

Dans ce mémoire nous avons étudiés le problème de classification des empreintes digitales qui est un problème très complexe. Celui-ci est survenu à plusieurs reprises dans l'identification des personnes par les empreintes digitales. Pour la résolution de ce problème on a utilisé les réseaux de neurones flous comme un classificateur qui a comme entrées les caractéristiques globales de l'empreinte digitale qui sont le core et le delta.

Le travail réalisé consiste en deux étapes fondamentales : en premier lieu il y a l'extraction des points singuliers globaux (core et delta) de l'image de l'empreinte digitale en second lieu l'étape de classification.

Pour l'extraction des points singuliers globaux, l'image de l'empreinte digitale subit les traitements suivants :

- L'extraction de l'image d'orientation correspondante;
- L'utilisation de l'algorithme de l'index de Poincaré pour la détection des points singuliers globaux ;
- Le post-traitement (segmentation) est effectué pour éliminer des faux cores et deltas
- La construction de vecteur caractéristique qui est constitué de (nombre de core, nombre de delta et la position de delta par rapport au core).

Les résultats obtenus pour cette première étape, sont très satisfaisants et surtout pour les images de bonne qualité.

Et, concernant la classification on a adopté quatre approches différentes :

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

Après l'implémentation de ces approches et l'analyse des résultats obtenus, on a pu constater que toutes les approches 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.

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 de notre système pourraient être nettement améliorées si on ajoute d'autres règles d'inférences à notre base des règles. Nous préconisons aussi d'ajouter d'autres informations au vecteur caractéristique.

Références

Bibliographiques

Bibliographie

- [1] M.ANDRE, *‘Introduction aux techniques de traitement d’images’*, Eyrolles 1987.
- [2] <http://www.map.toulouse.archi.fr/works/panoformation/imagenum/imagenum.htm>.
- [3] <http://www.wikipedia.org/>
- [4] M.BOUTERFAS, M.SOUIDI, *‘Développement d'un système de reconnaissance des cellules sanguines’*, 2011.
- [5] R. BELGUECHI."Contribution à la reconnaissance d'empreintes digitales par Approche Hybride" Mémoire de Magister, Institut National de formation en Informatique (I.N.I).2006.
- [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] N.GALY, *‘ Etude de système complet de reconnaissance d’empreintes digitales pour un capteur microsysteme à balayage’*, Mémoire de Doctorat, Institut National Polytechnique de GRENOBLE, 2005.
- [9] H.DANG, *‘ Biométrie pour l’Identification’*, Projet de fin d’étude Institut de la Francophonie pour l’Informatique, 07 – 2005.
- [10] R.Cappelli, D.Maio, *‘ The State of the Art in Fingerprint Classification’*, Automatic Fingerprint Recognition Systems, p183-193, 2004.
- [11] N.JEAN, *‘Analyse et classification des empreintes digitales’*, projet de fin d’étude, Université de Caen Département d'informatique, 2009.
- [12] B.LE DUC, *‘Authentification des empreintes digitales dans un système BioPKI’*, L'Institut de la Francophonie pour l'Informatique, 20 janvier 2007.
- [13] T.RAYMOND, *‘Fingerprint image enhancement and minutiae extraction’*, The University of Western Australia, 2003.
- [14] M. MOHAMED, H. NYONGESA, *‘Automatic Fingerprint Classification System Using Fuzzy Neural Techniques’* School of Computing and Management Sciences, Sheffield Hallam University, Sheffield, IEEE, 2002.

- [15] D.MALTONI, D.MAIO, K.JAIN, S.PARABHAKAR, '*Handbook of Fingerprint Recognition*', Spring. Library of Congress Cataloging-in-Publication Data, pp 174-184, 2003.
- [16] X.GUOL, Y.YIN, Z. SHI, '*Personalized Fingerprint Segmentation*', Springer-Verlag Berlin Heidelberg, pp 806, 2009.
- [17] M.KAWAGOE, A.TOKO, '*Fingerprint Pattern Classification*' Pattern Recognition, vol.17, pp. 295-303, 1984.
- [18] 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.
- [19] 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.
- [20] M.KAMIJO, '*Classifying fingerprint images using neural network: Deriving the classification state*', *Neural Network*, pp. 1932-1937, 1993.
- [21] T. DUSENGE, '*La Reconnaissance des Empreintes Digitales*', BA3-INFO Université Libre de Bruxelles, 25 mai 2009.
- [22] S.CHEKROUN, '*Commande Neuro- floue sans capteur de vitesse d'une machine asynchrone triphase*', Mémoire de Magister, Ecole Normale Supérieure d'Enseignement Technologique d'Oran, Département de Génie Electrique, 2009.
- [23] 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.
- [24] F.LAHOUAZI. '*Mise en œuvre d'une stratégie de commande neuro-floue : Application à un pendule inversé*', Mémoire de Magister, Université de TIZI-OUZOU, Faculté de génie électrique et d'informatique, 2011.
- [25] 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.

- [26] R.JEAN-PHILLIPE, « *Réseaux neuronaux, Une introduction accompagnée d'un modèle Java* ».Vuibert, Paris, pp 56-57, 2010.
- [27] M.LARABI, B.ROUIDI, '*Diagnostic Neuro-Flou : Application à la machine asynchrone*', Mémoire de Magister, Institut d'électronique, Université de SKIKDA. 2005.
- [28] R.MAHADOUI, '*Diagnostic industriel par Neuro-Flou-Application a un système de production*'. Mémoire de Magister, Université El-hadj Lakhdar –BATNA Faculté des Sciences de l'Ingénieur, Département de Génie industriel, 2007.
- [29] 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, 14 décembre 2007.
- [30] 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.
- [31] G.BALMISSE, '*La logique floue*', septembre 2002.
- [32] 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*', Laboratoire d'Automatique et Productique (LAP) Université de Batna, Institut d'Informatique, Centre Universitaire de Khenchela, Algérie, 9-10-2010.
- [33] N.AOUCHAR, C.BEKHALED, '*Application des systèmes hybrides neuro-flous au diagnostic des huiles de transformateurs*', Mémoire de fin d'études, Ecole Polytechnique d'Alger, Département d'Electrotechnique, 2005.

Résumé:

Le travail suivant présente un système de classification des empreintes digitales et ses performances. Ce système est basé sur l'extraction des points singuliers globaux des empreintes digitales (cores et deltas) et l'utilisation des réseaux de neurones flous. L'image de l'empreinte digitale subit les traitements suivant : l'extraction de l'image d'orientation correspondante, détection des points singuliers globaux par l'algorithme de l'index de Poincaré et la construction de vecteur caractéristique. Le but de ce système est de classer les empreintes digitales en cinq classes selon la classification de Henry et ceci est mis en œuvre en utilisant deux approches : les réseaux de neurones artificiels et les réseaux de neurones flous, les résultats obtenus par ces deux approches sont très satisfaisantes. Le taux de classification pour le réseau de neurones artificiels est 96%, et ceux des réseaux de neurones flous sont 100%.

Mots clés : L'empreinte digitale, traitement d'images, classification, les points singuliers globaux, les réseaux de neurones artificiels, la logique floue, les réseaux de neurones flous.

Abstract:

This work presents a fingerprints classification system and its performances. This system is based on the extraction of the global features of the fingerprints (cores and deltas) and the use of fuzzy neural networks. The fingerprint image undergoes the following treatments: the extraction of the corresponding orientation image, detection of the global singulars points by the Poincaré index algorithm and the construction of the characteristic vector. The objective of this system is to classify fingerprints into five classes according to the classification of Henry and this is implemented using two approaches: artificial neural networks and fuzzy neural networks, the results obtained by these two approaches are very satisfactory. The classification rate of the artificial neural network is 96%, and those fuzzy neural networks are 100%.

Keywords: The fingerprint image, images processing, classification, global singulars points, the artificial neural networks, fuzzy logic, fuzzy neural networks.

ملخص:

العمل التالي يعرض نظاما لتصنيف البصمات وأهميته. يستند هذا النظام على استخراج النقاط الفريدة العامة من البصمات (النوى و دلتا)، واستخدام الشبكات العصبية الغامضة. تخضع صورة البصمة للعمليات التالية: استخراج صورة التوجيه الموافقة للبصمة ، اكتشاف النقاط الفريدة العامة بواسطة خوارزمية مؤشر بوانكاريه و تكوين شعاع الخصائص. الغرض من هذا النظام هو تصنيف البصمات إلى خمس فئات وفقا لتصنيف هنري ويتم هذا باستخدام نهجين: الشبكات العصبية الاصطناعية والشبكات العصبية الغامضة، والنتائج التي تم الحصول عليها من قبل هذين النهجين كانت مرضية للغاية، حيث أن معدل تصنيف الشبكة العصبية الاصطناعية هو 96%، و معدل تصنيف الشبكات العصبية الغامضة هو 100%.

الكلمات المفتاح: بصمة الأصبع، معالجة الصور، التصنيف، النقاط الفريدة العامة ، الشبكات العصبية الاصطناعية، المنطق الغامض، الشبكات العصبية الغامضة.