

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/324480227>

Reconnaissance d'écriture

Research · April 2004

CITATIONS

0

READS

11

1 author:



[Azzeddine Menasria](#)

Université 8 mai 1945 - Guelma

2 PUBLICATIONS 0 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Apport des principales approches avancées dans la modélisation et la commande robuste des systèmes de production d'énergies renouvelables. [View project](#)

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET

DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DE CONSTANTINE

FACULTE DES SCIENCES ET DE L'INGENIEUR

DEPARTEMENT D'ELECTRONIQUE

THESE DE MAGISTER

Présenté par :

Mr. Menasria Azzeddine

Option

Traitement du signal

Thème:

**RECONNAISSANCE D'ECRITURE
MANUSCRITE
PAR TECHNIQUE
A BASE DE RESEAUX DE NEURONES**

Soutenu le 03/2005

Examiné par le jury

Président	Mr. M.barkat	Professeur	Université Constantine
Rapporteur	Mr. A.Bennia	Maitre de Conférence	Université Constantine
Examineur	Mr. A.Chref	Professeur	Université Constantine
Examineur	Mr. M.Khamadja	Professeur	Université Constantine

Année 2004/2005

REMERCIEMENTS

Je tiens à remercier Monsieur **A.H. Bennis** pour avoir proposé le sujet de cette thèse, pour ses conseils qu'il m'a souvent donné et de m'avoir facilité énormément la tâche lors la rédaction.

Je remercie le professeur Monsieur **H.Tebbikh** le directeur de Laboratoire de l'Automatique et l'Informatique Industrielle de Guelma (LAIG), de m'avoir autorisé à utiliser le matériel informatique.

Je remercie également le professeur **M.Barkat** pour l'honneur qu'il me fait en acceptant de présider le jury, ainsi Monssieurs **A. Charef**, professeur à l'Université de Constantine, et **M. Khamadja**, professeur à l'Université de Constantine, d'avoir accepté de juger le travail de cette thèse.

Je tiens à remercier également tous les membres du laboratoire du traitement du signal à Constantine, sans oublier les membres du laboratoire 'LAIG' de l'université de Guelma.

Que toutes les personnes (dont le nombre est très élevé pour les citer ici) qui ont contribué de près ou de loin, directement ou indirectement à ce travail, trouvent ici le témoignage de ma profonde reconnaissance. Enfin je ne pourrais pas terminer ces remerciements sans une pensée à l'ensemble de mes enseignants qui sont à l'origine de tout mon savoir.

ملخص

في هذه الأطروحة قمنا بدراسة مشكلة التعرف على الكتابة الخطية بطريقة أوتوماتيكية وقد خصصنا هذا العمل في معالجة الكتابة الحرفية لللكوك البريدية.

كما قمنا بعرض مختلف المراحل اللازمة، ابتداء من إدخال الصورة عن طريق فاحص الصور سكانير أو آلات التصوير الرقمية ثم مرحلة ما قبل المعالجة بهدف إزالة التشويشات الناتجة عن المرحلة الأولى، فإذا أصبحت الصورة مؤهلة للمعالجة تأتي مرحلة التجزئة حيث يتم استخراج السطور ثم تجزئتها إلى كلمات ليتم التعرف عليها بحيث يمكن تجزئة الكلمة ثم التعرف على كل حرف على حدى أو عملية التعرف على الكلمة جملة واحدة.

في كلتا الحالتين يتم استخلاص جملة من الخصائص المشتركة بين المفردات أو الحروف لتأتي مرحلة التصنيف عن طريق شبكة العصبونات بحيث تكون الخصائص المستخرجة من الحروف أو الكلمة هي مدخل الشبكة و أسماء الحروف أو الكلمات هي مخرج الشبكة العصبونية.

Résumé

Dans ce mémoire nous avons traité le problème de la reconnaissance de l'écriture manuscrite hors-ligne, dont l'intérêt industriel est la lecture automatique des montants littéraux de chèques postaux. Nous avons proposé une étude basée sur l'analyse optique du texte. Dans un premier temps une phase de pré-traitement de l'image permet de réduire la variabilité intrinsèque à l'écriture ainsi que les bruits possiblement introduits lors de l'acquisition. Une étape de segmentation permet d'isoler les lignes, et divise les mots en caractères. Les systèmes de reconnaissance sont liés à la nature de la modélisation mise en œuvre. L'approche globale consiste à modéliser les mots comme des entités globales et l'approche analytique fondamentalement opposée consiste à reconnaître un mot après avoir identifié les lettres le constituant. La dernière étape à être réalisée directement sur les données présentées à l'entrée du système est l'extraction des caractéristiques les plus pertinentes pour la reconnaissance. Finalement la classification est réalisée par simulation connexionniste à base de réseaux de neurones multicouches.

Abstract

In this work, we dealt with the problem of off-line recognition of the handwritten writing, one of its applications is the automatic reading of the literal amounts of postal cheques. We proposed a study based on the text optical analysis. Initially a phase of preprocessing of the image makes it possible to reduce meriting intrinsic variability as well as the noise possibly introduced at the time of acquisition.

A stage of segmentation allows the application of lines and divides the words in characters. Recognition systems are related to the nature of the modeling adapted. The global solution consists of modeling words like complete entities and the analytical approach basically opposed consists of recognizing a word after identifying their letters. The last step to be realized directly on the data presented at the input of the system is the extraction of the most relevant characteristics for the recognition. Finally the classification is carried out by connexionnist simulation based on multi-layer neural networks.

LISTE DES FIGURES

<u>FIGURE</u>	<u>TITRES</u>	<u>PAGES</u>
Figure I.1	Schéma général d'un système de reconnaissance de forme.....	05
Figure II.1	Etapes de reconnaissance d'écriture.....	17
Figure II.2	Histogramme de niveaux de gris.....	18
Figure II.3	Correction de l'inclinaison.....	20
Figure II.4	Exemple du mot « vingt » avant et après l'extraction de ontour.....	22
Figure II.5	Exemple du mot « vingt cinq» avant et après l'extractionde contours exterieur.....	22
FigureII.6a	Propagation de l'étiquetage des pixels de gauche à droite sur une ligne Horizontale.....	23
FigureII.6b	Propagation de l'étiquetage des pixels de haut en bas sur une colonne verticale.....	24
Figure II.6c	Conflit entre la propagation horizontale et verticale.....	24
Figure II.6d	Résolution du conflit correspondant au 1 ^{er} cas.....	24
Figure II.6e	Résolution du conflit correspondant au 2 ^e cas.....	25
Figure II.7	Histogramme de la segmentation lignes.....	26
Figure II.8	Histogramme de la segmentation... ..	27
Figure II.9	Extraction des caractères.....	28
Figure II.10	Les alignements dans le mot.....	29
Figure II.11	Exemples de Localisation des hampes, jambages et médiane du mot 'vingt'.....	32
Figure II.12	Extraction des boucles méthode d'étiquetage.....	33
Figure II.13	Extraction des boucles méthode basée sur les contours.....	33
Figure II.14	Division de l'image en des zones.....	34
Figure II.15	Représentation d'un contour de la lettre « a » à la moyen d'une grille.....	34
Figure II.16	Définition du code de chaîne.....	35
Figure II.17	Exemple des différentes projections des Profils pour la lettre 'a'.....	36
Figure II.18	Opérations géométrique.....	38
Figure II.19	Représentation complexe d'un contour.....	41
Figure II.20	Reconstruction de la forme originale du contour extérieur.....	42
Figure III.1	Vue d'artiste des connexions neuronales dans le cerveau.....	48
Figure III.2	Le neurone formel de McCulloch et Pitts.....	50
Figure III.3	Exemple d'un neurone formel.....	51
Figure III.4	Fonction de Heaviside, utilisée par MacCulloch et Pitts.....	51
Figure III.5	Modélisation d'un neurone.....	52
Figure III.6	Réseaux multicouches.....	

Figure III.7	Réseaux entièrement connectés.....	54
Figure III.8	Structure d'un perceptron.....	55
Figure III.9	Architecture de L'ADALINE.....	56
Figure III.10	Réseau de neurones multicouches.....	61
Figure IV.1	Aperçu sur la base de donnée.....	65
Figure IV.2	Extraction des paramètres de distribution.....	66
Figure IV.3	Le vecteur caractéristique Histogrammes de Direction.....	67
Figure IV.4	Les huit projections des Profils pour la lettre 'a'.....	68
Figure IV.5	Evolution de l'erreur en fonction de nombre d'itération.....	69
Figure V.1	Caractères arabe ont un même corps.....	72
Figure V.2	a- différentes composantes connexes (c.c), b- les diacritiques.....	73
Figure V.3	Architecture générale du système de classification neuronal.....	74
Figure V.4	Exemple d'un c.c contient un et deux boucles.....	77
Figure V.5	Détection de diacritique.....	77
Figure V.6	Caractéristique d'histogramme de direction.....	79
Figure V.7	Présentation du vecteur de primitive.....	80
Figure V.8	Evolution de l'erreur en fonction du nombre d'itérations.....	81

LISTE DES TABLEAUX

<u>TABLEAU</u>	<u>TITRES</u>	<u>PAGES</u>
Tableau II.1	Répartition des exemples dans la base de test cas général	47
Tableau IV.1	Taux de reconnaissance sur les ensembles d'apprentissage et de test.	69
Tableau V.1	Vocabulaires des montants littéraux de chèques.....	75
Tableau V.2	Représentation de points diacritique de chaque cc de la figure V.5.....	78
Tableau V.3	Décomposition du vocabulaire.....	78
Tableau V.4	Taux de reconnaissance sur les ensembles d'apprentissage et de test.	81

TABLE DES MATIERES	Page
LISTE DES FIGURES	I
LISTE DES TABLEAUX	III
INTRODUCTION GENERALE	01
CHAPITRE I RECONNAISSANCE DES FORMES	
1. Introduction.....	05
2. Processus de reconnaissance des formes.....	05
2.1. Le monde physique « réel ».....	06
2.2. Le codage.....	06
2.3. Le pré traitement.....	06
2.4. L'analyse.....	06
2.5. L'apprentissage.....	07
2.6. La décision.....	07
3. Méthodes et technique de classifications.....	08
3.1. La méthode bayesienne.....	08
3.2. Classificateur euclidien.....	09
3.3. Le classificateur quadratique.....	09
3.4. La méthode du plus proche voisin.....	10
3.5. Les méthodes connexionnistes.....	10
4. Domaines d'applications de la reconnaissance des formes.....	11
4.1. Reconnaissance des signaux.....	11
4.2. Reconnaissance des images.....	11
5. Conclusion.....	12
CHAPITRE II LA RECONNAISSANCE DE L'ECRITURE	
1. Introduction.....	13
2. Différents aspects de reconnaissance de l'écriture.....	13
2.1 Reconnaissance en-ligne ou hors-ligne.....	14
3. Méthodes de reconnaissance.....	15
3.1 Reconnaissance globale ou analytique	15
4. Stratégies de reconnaissance	16
5. Organisation générale d'un système de reconnaissance.....	16
5.1 Prétraitements.....	17
5.1.1 Binarisation.....	18
5.1.2 Lissage.....	19
5.1.3 La correction de l'inclinaison.....	20
5.1.4 Normalisation de la taille.....	20

5.1.5 Extraction des contours.....	21
5.1.6 Extraction des composantes connexes.....	22
5.2 Segmentation.....	25
5.2.1 Extraction des mots.....	27
5.2.2 Extraction des caractères.....	28
5.2.3 Les alignements dans le mots.....	28
5.3 Extraction des primitives (caractéristiques).....	30
5.3.1 Les caractéristiques adaptées a l'écriture.....	32
5.3.1.1 Détection des caractéristiques structurelles.....	32
5.3.1.2 Détection des caractéristiques statistiques.....	33
5.3.1.3 Détection des caractéristiques géométriques.....	34
5.4 Classification.....	42
5.4.1 Les approches de Classification.....	43
5.4.1.1 Les approches statistiques	43
5.4.1.2 Les approches structurelles	45
5.4.1.3 Approches hybride.....	45
5.5 La décision.....	45
5.5.1 Evaluation des performances d'un classifieur.....	46
5.5.2 Taux de reconnaissance et taux d'erreurs.....	46
Conclusion.....	47
CHAPITRE III RESEAUX DE NEURONES	
1. Généralités.....	48
2. Modélisation du neurone.....	49
2.1 Réseaux de neurones formels.....	49
2.1.1 Model simplifie (premier modèle de neurone).....	50
2.1.2 Modélisation générale.....	52
2.2 Différentes structures des réseaux de neurones.....	53
2.2.1 Les Réseaux à couches.....	53
2.2.2 Les Réseaux entièrement connectés.....	54
2.3 Les réseaux les plus célèbres.....	55
2.3.1 Le perceptron.....	55
2.3.2 Le modèle ADALINE.....	55
2.3.3 Le perceptron multicouches (PMC).....	56
2.4 L'apprentissage.....	56
2.4.1 Apprentissage supervisé.....	56
2.4.2 Apprentissage semi-supervisée.....	57
2.4.3 Apprentissage non-supervisée (auto-organisation).....	57

2.5 Règles d'apprentissages.....	57
2.5.1 La règle de Hebb	57
2.5.2 La règle de Widrow-Hoff.....	58
2.6 Le Perceptron Multicouches Comme un classifieur.....	59
2.7 Evolution du perceptron multicouches.....	63
Conclusion.....	64
CHAPITRE IV RECONNAISSANCE DE CARACTERES LATINS	
1 Introduction :.....	65
2 Application a la reconnaissance des caractères Latins.....	65
2.1 Base de données.....	65
2.2 Pré-traitements.....	65
2.3 Le choix des primitives.....	66
2.3.1 Les Paramètres de Distribution (PD).....	66
2.3.2 Le vecteur caractéristique Histogrammes de Direction (HD).....	67
2.3.3 La densités des profiles (DP):.....	68
2.3.4 Les sept moments invariants (MS).....	68
2.3.5 Les coefficients de Fourier (DF).....	69
2.4 Constitution du vecteur de caractéristiques.....	69
2.5 Classification et reconnaissance.....	70
2.5.1 Implémentation de réseaux.....	70
2.5.2 L'apprentissage.....	70
2.5.3 Reconnaissance.....	70
CHAPITRE V APPROCHES GLOBALES POUR LA RECONNAISSANCE DES MOTS MANUSCRITS	
1. Introduction.....	72
2. Caractéristiques de l'écriture arabe.....	72
3. Phases de conception du système de reconnaissance.....	73
3.1 Acquisition.....	75
3.2 Pré-traitement.....	75
3.3 Détection et choix des primitives	75
3.3.1 Caractéristiques structurelles.....	76
3.3.1.1 Les nombres des composantes convexes.....	76
3.3.1.2 Le nombre de boucles pour chaque c.c	76
3.3.1.3 Le nombre de hampe et jambage pour chaque c.c.....	77
3.3.1.4 Le nombre de pointe diacritique	77
3.3.2 Caractéristiques géométrique.....	79
3.4 Constitution du vecteur de caractéristiques.....	79

4. Classification.....	80
4.1 Implémentation de réseaux.....	80
4.2 Apprentissage.....	81
4.3 Reconnaissance.....	81
Conclusion générale.....	82

INTRODUCTION GENERALE

Dans le domaine de l'intelligence artificielle, l'un des objectifs les plus recherchés est de doter les ordinateurs de capacités qui sont naturelles à l'être humain. Un premier pas de ce sens consiste à faciliter la communication homme-machine en la rendant la moins contraignante possible.

L'écriture manuscrite est un moyen naturel de communication qui présente l'avantage d'être familier à la majorité des gens. De ce fait, elle constitue un moyen d'interaction facile avec l'ordinateur. Un système capable de lire le texte manuscrit devrait permettre aux ordinateurs de traiter une grande variété d'applications non encore accessibles. Cette tâche n'est pas triviale car les mots possèdent une infinité de représentations dues au fait que chaque personne possède une écriture qui lui est propre.

En outre depuis quelques années, les recherches dans le domaine de la reconnaissance de l'écriture se sont multipliées de manière spectaculaire et de nombreux nouvelles techniques ont été développées. Ces techniques sont loin d'atteindre des performances comparables à l'être humain. En absence de toute contrainte, la reconnaissance de l'écriture manuscrite s'avère un problème extrêmement complexe qui n'a pas de solution satisfaisante jusqu'à ce jour.

On distingue dans l'écriture deux types d'écriture : l'imprimé et le manuscrit. Dans le premier type les principales difficultés ont été surmontées ; la situation diffère complètement en ce qui concerne la reconnaissance des textes manuscrits. Les problèmes rencontrés sont liés d'une part au manque de consistance des caractères, et aux lettres liés à l'intérieur des mots et qui ont des formes différentes. Ils sont liés aussi aux nombres de scripteurs.

En revanche pour les systèmes de la reconnaissance de l'écriture, on distingue deux grandes catégories d'applications à vocabulaire limité où le nombre des mots à reconnaître constitue un lexique de taille restreinte (*approche globale*) ; et les applications à vocabulaire étendu nécessitent obligatoirement une phase de segmentation (*approche analytique*).

Des nombreuses études présentées dans la littérature, la plupart portent sur la reconnaissance de sous-unités isolées d'écriture, tels que les caractères, les mots ou

les chaînes de chiffres, parmi les travaux récents réalisés dans ce domaine on trouve :

Leroux et al [1] proposent une stratégie de contrôle pour la reconnaissance des montants littéraux des chèques postaux, c'est un système omni-scripteur, fondé sur la stratégie de coopération entre une méthode analytique et méthode globale, dans ce travail aucun résultat n'a été présenté, c'est uniquement une stratégie proposée qui peut être adoptée pour la réalisation des système de reconnaissance de montants littéraux de cheques,

Lemarie et al [2] effectuent une coopération neuro-Markovienne pour la reconnaissance des montants littéraux des cheques, le bilan de cette technique été positif, un taux de reconnaissance de l'ordre de 80.28% est atteint sur une base de teste de 2879 mots.

Miled [3] propose deux approches Markoviennes pour la reconnaissance hors-ligne omni-scripteur de l'écriture arabe, dans un vocabulaire relativement étendu (232 classes de mots différents correspondants à des noms de villes tunisiennes), Les modèles ont été entraînés sur une base de 4720 mots et les tests ont été effectués sur une autre base de 5900 mots. Des résultats modestes de 60% ont été enregistrés.

Cheriet et al [4] proposent un système de reconnaissance de chaînes de chiffres indiens dédiée à la lecture automatique des montants numériques de chèques arabes, fondé sur la stratégie de coopération neuro-flou, le taux de reconnaissance est de l'ordre de 95.80%,

Lessandrol [5] propose un système omniscritteur de reconnaissance hors-ligne d'écriture manuscrite sans contrainte, capable de traiter un lexique de très grande dimension. Le taux de réussite est d'environ 78%,

Morita [6] , présente un système hybride HMM-MLP pour la segmentation et la reconnaissance de l'écriture manuscrite et plus particulièrement des dates et les montants littéraux présente sur les chèques brésiliens. Le bilan de cette technique a été positif, un taux de reconnaissance de 82.3% pour les dates et 96.8%pour les mots.

Nazif [7] propose un algorithme de segmentation et de reconnaissance d'écriture manuscrite hors-ligne. La phase de reconnaissance réalisé par l'approches

markoviennes, un taux de reconnaissance de 89.3% pour une base de test de 30000 mots et d'environ 88.8% pour une base de teste de 40000 mot.

Alceu et al [8] proposent un approches markoviennes pour la reconnaissance hors-ligne de chiffres arabe. Les tests ont été effectués sur une base de 12802 chiffres et donne un taux de reconnaissance 89.6%.

Bazziz [9] et al présente un systeme de reconnaissance

Notre travail s'insère dans le cadre de la reconnaissance hors-ligne de l'écriture manuscrite à vocabulaire limité, l'application visée derrière ce travail est la reconnaissance des montants littéraux des chèques postaux, en utilisant un classifieur neuronal.

Le but de ce mémoire est l'application de deux approches pour la reconnaissance de l'écriture : Approche analytique pour la reconnaissance de caractères latins et approche globale pour la reconnaissance de mots arabes.

Dans le premier chapitre nous donnons un aperçu général sur la reconnaissance des formes, et nous décrivons le schéma général d'un système de reconnaissance ainsi que les méthodes de classification et leurs domaines d'utilisation.

Le second chapitre traite des systèmes de reconnaissance de l'écriture manuscrite où nous présentons les aspects généraux de la reconnaissance de l'écriture et nous détaillons les étapes nécessaires au développement d'un système de reconnaissance de caractères.

Le chapitre 3 est consacré à l'état de l'art des réseaux de neurones, entre autre le MLP adopté dans notre projet. Une description de la méthode d'apprentissage utilisée est aussi présentée.

Le quatrième chapitre est consacré à la conception et à l'implémentation du modèle appliqué à la reconnaissance de caractères latins. Dans ce chapitre, nous abordons les différentes étapes de reconnaissance et les aspects pratiques liés aux choix des différents paramètres. Nous terminons ce chapitre par une présentation des résultats de notre application.

Le cinquième chapitre est spécifique à l'application à la reconnaissance globale de l'écriture arabe. La première section est un rappel des caractéristiques de l'écriture arabe. La seconde section présente la conception et l'implémentation de

notre système, et nous terminons ce chapitre par une représentation aux résultats commentés obtenus à partir de notre base de donnée.

CHAPITRE I

RECONNAISSANCE DES FORMES

RECONNAISSANCE DES FORMES

1. Introduction

La reconnaissance des formes est historiquement un chapitre de l'intelligence artificielle, elle a pour but d'essayer de recopier les sens de perception de l'être humain et d'essayer dans une certaine mesure de doter la machine par ces sens.

Les recherches effectuées sur la psychologie de perception ont montré qu'un sujet soumis à des excitations sensorielles reconnaît une forme s'il est capable d'effectuer les opérations suivantes :

- assimiler ces excitations à des excitations reçues antérieurement, et présentant certaines analogies.
- les distinguer des autres.

Ainsi interviennent les notions de recueil des données, d'analyse des données et de classification [11].

2. Processus de reconnaissance des formes

La reconnaissance des formes est la première étape d'un long processus de compréhension de notre univers dans le cadre général de la communication homme-machine.

Un système de reconnaissance de formes est représenté suivant le schéma général de la *figure I.1* [12].

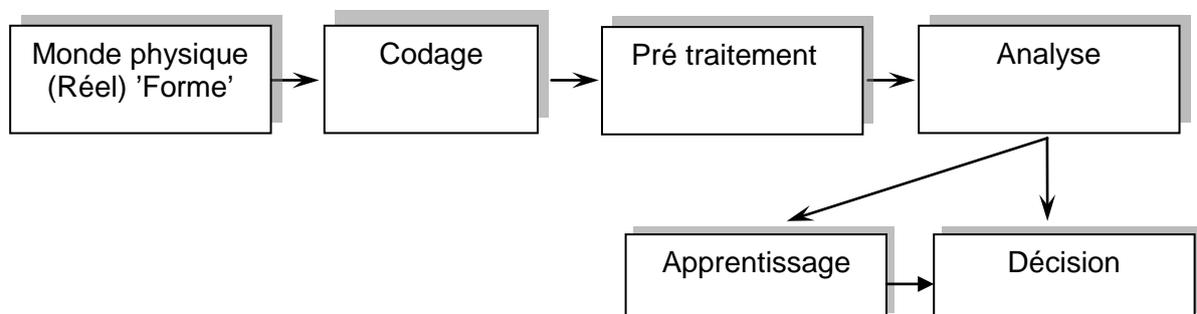


Figure I.1. Schéma général d'un système de reconnaissance de formes [2].

Rappelons, en premier lieu, le rôle de chaque étape dans l'élaboration du processus complet de reconnaissance :

2.1. Le monde physique « réel »

La chaîne part du monde physique qui est un espace analogique de dimension infini appelé espace des formes. Dans ce vaste espace, il est difficile de tenir compte de chaque objet seul lors du traitement d'un problème de reconnaissance de formes. La loi de passage au monde discret nécessite forcément une sélection et par conséquent une certaine simplification.

2.2. Le codage

Cette étape consiste à faire une conversion du monde physique continu vers un monde numérique de manière à pouvoir traiter, à transformer un ensemble de données analogiques en un ensemble de données numériques de manière à pouvoir les traiter par ordinateur. Cette transformation doit se faire de la manière la plus fidèle possible, c'est à dire sans perte d'information pertinente et en conservant les propriétés essentielles de l'objet physique. Cet espace, appelé aussi espace de représentation, a une dimension encore trop importante même si elle est finie. La dimension de cet espace est choisie volontairement grande de manière à pouvoir disposer d'un maximum d'information sur la forme.

2.3. Le pré traitement

Les étapes de prétraitement consistent à sélectionner dans l'espace de représentation l'information nécessaire à l'application. Cette sélection passe souvent par l'élimination du bruit dû aux conditions d'acquisition, par la normalisation des données, aussi bien que par la suppression de la redondance.

Donc, le rôle du prétraitement est de préparer les données reçues du capteur à la phase suivante d'analyse consacrée à l'extraction des paramètres.

2.4. L'analyse

Lors de l'étape d'analyse, les techniques de RdF calculent un certain nombre de caractéristiques ou paramètres. Ces paramètres correspondent à des mesures de natures géométrique, topologique ou statistique et servent comme seules données représentant la forme. Ils sont généralement limités en nombre.

Donc, le but de l'analyse est d'extraire les propriétés caractéristiques de l'objet et de les exprimer sous une forme numérique ultérieure d'apprentissage et de reconnaissance.

2.5. L'apprentissage

L'apprentissage ou modélisation est une étape clé dans la chaîne de reconnaissance. Son rôle est d'éclairer la décision à l'aide de reconnaissance a priori sur les formes.

A partir de critères spécifiques aux formes, l'apprentissage tente de définir des modèles de référence ou de caractériser des classes de décision. Il permet ainsi de dicter au système l'algorithme de décision le plus adéquat vis-à-vis des règles de modélisation choisies.

L'apprentissage peut être paramétrique (modélisation de paramètres), ou non paramétrique (la décision ne nécessite pas la spécification de types de paramètres et se traduit, par exemple, par une minimisation du risque d'erreur).

2.6. La décision

La décision ou classement est l'étape de reconnaissance proprement dite, son rôle est d'identifier la forme test à partir de l'apprentissage réalisé.

La méthode de décision est souvent celle utilisée par l'apprentissage, ce qui veut dire que les critères utilisés pour la comparaison sont les mêmes que ceux utilisés pour l'apprentissage. En effet, il est évident que le choix de critères différents pour la décision, n'assurant pas l'uniformité de la description, ne peut pas conduire avec certitude à un résultat cohérent.

Parmi les techniques utilisées, certaines sont fondées sur la notion de proximité et nécessitent de calculer une distance ou une probabilité de ressemblance avec les modèles définis. D'autres sont fondées sur l'analyse de la structure de la forme et essayent plutôt de vérifier une certaine cohérence dans les relations entre les différentes sous structures.

La réponse de la décision peut être, selon le cas, le nom de la forme en cas de bonne reconnaissance, plusieurs noms en cas d'ambiguïté ou bien le rejet de la forme en cas d'incompatibilité de description avec les formes de références. Dans les deux premiers cas, la réponse peut être accompagnée d'un taux ou score de confiance.

3. Méthodes et technique de classifications

Il existe plusieurs méthodes de classification des données, le choix de la dernière dépend de plusieurs critères tel que le type de caractéristiques de la forme mais le point commun à toutes ces techniques est l'extraction d'un petit nombre de paramètres caractéristiques.

3.1. La méthode bayésienne

L'approche bayésienne consiste à choisir parmi un ensemble des formes, celle pour laquelle la suite de primitives (observations) extraite a la plus grande probabilité a posteriori par rapport aux formes préalablement apprises.

Dans cette approche, on suppose que les mesures x_i (suite d'observations) faites sur une forme peuvent s'exprimer sous la forme d'un vecteur $X=(x_1, x_2, \dots, x_n)$, de l'espace R_n . On dispose d'un ensemble V d'apprentissage, c'est-à-dire d'un jeu de tels vecteurs dont on connaît en plus la classe d'appartenance. Le problème peut alors se résumer sommairement de la façon suivante : étant donné un vecteur inconnu obtenu par mesure sur une forme, trouver la classe à laquelle on doit l'affecter. La probabilité que la vraie forme « c », représentée par le vecteur $C=(c_1, c_2, \dots, c_n)$; ait pu générer X est donnée par la formule de Bayes

$$P\{C|X\} = \frac{P(C)P(X|C)}{P(X)} \quad \text{I.1}$$

Cette formule met en évidence les termes fondamentaux de l'approche statistique, qu'il faudrait déterminer pour réduire la classe optimale :

$P(X|C)$: est la probabilité d'observer X sachant que la vraie suite d'observation est C , (représente le vrai caractère « c »). Ce terme est lié à la variation des observations correspondant à la classe de C .

La valeur de C qui maximise la probabilité conditionnelle $P(C|X)$ est appelée *solution de maximum de vraisemblance* (MV), celle qui maximise la probabilité conditionnelle $P(C|X)$ est appelée *solution au sens du maximum a posteriori* (MAP), $P(C)$ est la probabilité a priori que C soit la suite d'observations choisie aléatoirement parmi l'ensemble V des vecteurs d'observations et $P(X)$ est la probabilité de la suite d'observations X .

Puisque $P(X)$ est indépendant de C , le vrai caractère (en occurrence c) qui fournit la

meilleure interprétation de la suite d'observation, est celui qui maximise la probabilité conjointe $P(X, C)$

$$P(X/C) = \text{MAX}_{C \in V} P(X/C) * P(C). \quad \text{I.2}$$

Pour estimer $P(C)$, il suffit de totaliser $P(X/C)$ pour toutes les suite observations (représentant, des caractères) $C \in V$

$$P(X) = \sum_{C \in V} P(X/C) * P(C) \quad \text{I.3}$$

Or ce n'est pas raisonnable de stocker toutes les combinaisons de $P(X/C)$.

Nous pouvons simplifier la formule en supposant l'indépendance conditionnelle des caractéristiques, ce qui nous amène à :

$$P(X/C) = \prod_{i=1}^n P(x_i / c_i) \quad \text{I.4}$$

La probabilité $P(x_i / c_i)$ est simplement la probabilité de confusion entre x_i et c_i , c'est-à-dire trouver la primitive x_i à la place de la primitive c_i .

3.2. Classificateur euclidien

Il s'agit de l'un des plus simples classificateurs qui puissent être conçus. La classe dont le vecteur de caractéristiques moyen est le plus proche, au sens de la distance Euclidienne, du vecteur de caractéristiques de l'objet à classifier est assignée à ce dernier. Les fonctions discriminantes utilisées sont donc de la forme suivante :

$$\Phi_i(X) = -\frac{1}{2}(X - M_i)^T (X - M_i) \quad \text{I.5}$$

3.3. Le classificateur quadratique

Comme le nom l'indique, les frontières de décision fournies par ce modèle de classificateur sont quadratiques. Les fonctions discriminantes s'expriment par :

$$\Phi_i(X) = -\frac{1}{2}(X - M_i)^T (X - M_i) \quad \text{I.6}$$

Où $M_i = E(X|C_i)$ est le vecteur de caractéristiques moyen des éléments qui appartiennent à la classe C_i , $E\{\}$ désignant l'opérateur d'espérance mathématique, et $\{\}^T$ celui de transposition.

Le terme quadratique $X X^T$ est indépendant de la classe de l'objet, et les fonctions discriminantes peuvent également s'écrire :

$$\Phi_i(X) = M_i^T X - \frac{1}{2} M_i^T M_i \quad \text{I.7}$$

Les frontières qui séparent les classes dans l'espace R^d sont ici linéaires.

3.4. La méthode du plus proche voisin

L'algorithme KNN (K Nearest Neighbors) associe une forme inconnue à la classe de son plus proche voisin en la comparant aux formes stockées dans une classe de références nommées prototypes. Il reste les K formes les plus proches de la forme à reconnaître suivant un critère de similarité. Une stratégie de décision permet d'affecter des valeurs de confiance à chacune des classes en compétition et d'attribuer la classe la plus vraisemblable (au sens de la métrique choisie) à la forme inconnue, le critère de similarité entre deux formes est basé sur la distance euclidienne.

3.5. Les méthodes connexionnistes

Ces méthodes relèvent du champ général des "réseaux de neurones formels". Étant donné que le but de la reconnaissance est de projeter l'ensemble des formes décrites par un ensemble de caractéristiques dans l'espace des classes, en vue de leur identification. On cherche alors à imposer au réseau de neurones choisi une certaine fonctionnalité en lui présentant en entrée la forme dont on connaît la classe d'appartenance, et en comparant la sortie qu'il produit avec celle qui code la classe idéale.

En effet, l'utilisation d'un réseau de neurones en classificateur dépend du codage des sorties. Dans la méthode généralement employée chaque unité de la couche de sortie représente une classe possible pour les formes en entrée. L'introduction sur la couche d'une forme inconnue induit après propagation des résultats de calculs élémentaires vers la couche de sortie des valeurs sur les différentes sorties. L'élément ayant la valeur la plus grande permet alors de choisir à quelle classe affecter la forme en entrée.

4. Domaines d'applications de la reconnaissance des formes

Nous citons tout d'abord quelques applications actuelles de la reconnaissance des formes et quelques unes des performances atteintes.

4.1. Reconnaissance des signaux

Le traitement du signal fournit des paramètres très utilisables pour pousser plus loin l'analyse, et décider quel est le signal émis, en fonction d'un répertoire de signaux possibles. Un exemple très spectaculaire, promis à un grand développement, est celui de la reconnaissance de la parole. Pour l'instant, les machines commercialisées reconnaissent des mots (ou suite de quelques mots), pris dans un vocabulaire limité à quelques dizaines d'éléments, des prototypes de laboratoires sont capables de tenir compte de la syntaxe et de maîtriser un vocabulaire raisonnablement étendu (quelques milliers de mots). On estime que d'ici une dizaine d'années, les machines seront capables de comprendre la parole humaine et auront de performances suffisantes pour être utilisables dans des tâches complexes (intégration de bases des données, acquisition de connaissance spécialisées,...etc.).

Un autre domaine extrêmement utile est celui des signaux biomédicaux ; la reconnaissance des formes permettrait d'automatiser ou de simplifier des tâches à la fois très complexes et très répétitives, comme un dépouillement électroencéphalogramme, ou encore assurer une surveillance automatique sur des mesures en temps réel. Un grand nombre d'études sont en cours dans ce vaste domaine, et déjà certains systèmes sont implantés en laboratoires hospitaliers.

Si l'on quitte les signaux physiologiques, la reconnaissance des formes s'intéresse aussi aux mesures des signaux d'origines artificielles : surveillance des machines, interprétation des échos,...etc. Un bon exemple est la détection d'objets sur signal radar.

4.2. Reconnaissance des images

L'autre domaine prépondérant est celui de l'analyse et l'interprétation des images. Depuis les dessins les plus simples, comme les chiffres ; jusqu'aux images multi spectrales complexes issues de satellites, le champ des applications est immense.

Dans le domaine des images médicales, on trouve les problèmes de comptage de cellules ou de chromosomes, de sélection de radiographies, d'interprétation des résultats de tous les systèmes d'imagerie.

Un grand nombre d'images proviennent du domaine de la robotique, en particulier industrielle: reconnaissance de pièces pour saisie, par exemple. L'analyse des paysages est également très utile (photos aériennes, guidage en temps réel d'engins, etc.).

Classification d'empreintes digitales : les empreintes digitales sont intéressantes pour identifier les personnes.

La lecture automatique de texte manuscrit, un des vieux et difficiles problèmes de la RdF. Seuls des cas relativement simples sont résolus avec un taux d'erreur tolérable, par exemple la lecture des codes postaux pendant le tri du courrier et les montants littéraux de chèques.

5. Conclusion

Dans ce chapitre nous avons présenté un aperçu général sur le processus de la reconnaissance des formes et les méthodes de classification ainsi que leurs domaines d'utilisation.

CHAPITRE III

RESEAUX DE NEURONES

LA RECONNAISSANCE DE L'ECRITURE

1. Introduction

De nos jours, l'écriture est toujours le moyen de communication visuelle le plus utilisé par l'homme. Il n'est donc pas surprenant de voir que de nombreux travaux scientifiques portent sur sa reconnaissance automatique. L'écriture est en fait la réalisation d'un message à transmettre, c'est-à-dire la représentation physique d'un contenu sémantique. Le média ou support généralement utilisé est le papier. Le but de la reconnaissance de l'écriture est de prendre une décision quant au contenu sémantique du message transmis à partir de sa représentation physique. Les applications de systèmes capables de remplir cette tâche sont nombreuses ; nous pouvons citer entre autres la lecture automatique de bons de commande, le traitement automatique des chèques, la vérification de signatures ou encore le tri automatique du courrier.

La reconnaissance de l'écriture est rattachée au vaste domaine de la reconnaissance de formes. Sa spécificité vient bien sûr des données à analyser et de leurs diverses sources de variation. En effet, un même mot écrit par plusieurs personnes peut avoir des formes assez différentes. Pour cette raison, les caractéristiques extraites de l'écriture sont très importantes pour la suite du processus de reconnaissance. Ce dernier quant à lui, peut être mis en oeuvre à l'aide de presque toutes les techniques développées en reconnaissance de formes.

2. Différents aspects de reconnaissance de l'écriture

La reconnaissance optique des caractères est un processus qui permet de convertir un texte écrit sur papier en texte sous forme d'un fichier informatique en représentation symbolique. Pour l'écriture latine, le codage typiques est opéré par le code ASCII (American Standard Code for Information Interchange), tandis que pour l'arabe on utilise généralement l'ASMO (Arabic Standard Metrology Organisation).

Par ailleurs, il n'existe pas de système universel d'OCR (Optical Character Recognition) qui peut traiter tous les cas d'écriture, mais plutôt des voies d'approche dépendant du type de données traitées et bien évidemment de l'application visée [12]. Il existe différents modes de classification des systèmes OCR, parmi les principaux, nous citons :

Les systèmes qualifiés de « **en-ligne** » ou « **hors-ligne** » relatifs au mode *d'acquisition* pour cela il existe un grand nombre de techniques classées généralement en deux grandes catégories :

- Les approches *globales* et *analytiques* selon que l'analyse s'opère sur la totalité du mot ou par segmentation en caractères (ou en parties de caractères) et analyse individuelle de ceux-ci.
- Les approches *statistiques*, *structurelles* et *stochastiques* relation qui traitent les caractéristiques extraites des formes considérées relatives au mode *de classification*.

2.1 Reconnaissance en-ligne ou hors-ligne

Ce sont deux modes différents d'OCR ayant chacun ses outils propres d'acquisition et ses algorithmes correspondants de reconnaissance.

- **La reconnaissance en-ligne (on line)**, dite encore dynamique s'opère en temps réel pendant l'écriture : les symboles sont reconnus au fur et à mesure qu'ils sont écrits à la main.

Ce mode, réservé généralement à la reconnaissance de textes manuscrits, présente plusieurs avantages. C'est une approche « signal » où la reconnaissance effectuée sur des données ou une seule dimension, l'écriture présente par une séquence de points dont les coordonnées sont fonction du temps ; ainsi l'ordre de l'écriture, l'information temporelle, les levers du stylo sont exploités reconnaissance. De plus, la réponse en contenu du système permet à l'utilisateur de corriger et de modifier son écriture de manière interactive. L'acquisition du tracé est assurée généralement par une tablette graphique munie par d'un stylo électronique, les coordonnées du point cursif. une résolution typique de *200 dpi (dot per inche)* et une fréquence d'échantillonnage de *100 points par seconde*, format le plus caractéristique les plus usitées [13].

- **La reconnaissance hors-ligne (off line)** ou en différé dite encore *statique*, démarre par l'acquisition. Elle convient pour les documents imprimés et les manuscrits déjà rédigés. Le mode hors-ligne peut être considéré comme le cas le plus général de la reconnaissance d'une écriture. Il se rapproche du mode de reconnaissance visuelle : l'interprétation de l'information est indépendante de source de génération, ce qui laisse les champs ouverts à différents types d'applications

La reconnaissance hors-ligne relève d'une approche de traitement généralement « image-2D », ce qui constitue une différence fondamentale avec la reconnaissance en-ligne classique. La quantité d'informations manipulées dans les deux cas est nettement différente.

A titre d'exemple, un mot cursif écrit en mode en-ligne (échantillonné à 100 points par seconde) nécessite environ 230 octets, alors que le mode hors-ligne, il occupe 80 K octets (numérisation à une résolution 300dpi) [15], les taux réalisées en mode en ligne sont généralement plus élevés que dans le cas hors ligne pour des conditions de test similaires en se basent sur les divers degrés de complexité des problèmes à traiter.

3. Méthodes de reconnaissance

3.1 Reconnaissance globale ou analytique

- **L'approche globale** : dit encore holistique considère le mot comme une seule entité et le décrit indépendamment des caractères qui le constituent. Cette approche présente l'avantage notamment de garder le caractère dans son contexte avoisinant qui permet une modélisation plus efficace des variations de l'écriture et des dégradations qui peuvent l'entacher. Cependant cette méthode est pénalisée par la taille mémoire, le temps de calcul et la complexité du traitement croît linéairement par la taille du lexique considéré, d'où une limitation du vocabulaire.
- **L'approche analytique** : contrairement à l'approche globale, le mot est segmenté en caractères ou en fragments morphologiques significatifs inférieurs aux caractères appelés *graphèmes*. La reconnaissance consiste alors à reconnaître les entités segmentées puis tendre vers une reconnaissance du mot, ce qui constitue une tâche assez délicate pouvant générer différents types d'erreurs [12].

4. Stratégies de reconnaissance

Les principales applications de la reconnaissance de l'écriture manuscrite sont la lecture des montants littéraux de chèques et la lecture des enveloppes postales. Dans les deux cas le système doit reconnaître un ensemble de mots. À ce moment deux stratégies peuvent être envisagées : reconnaître les mots séparément ou alors essayer de reconnaître le groupe de mots.

Dans la plupart des systèmes existants la reconnaissance des mots est tout d'abord mise en oeuvre, puis la combinaison des différents mots permet d'obtenir un résultat pour le groupe de mots. Dans le cas de la reconnaissance des montants littéraux de chèques cette dernière étape permet d'effectuer une vérification grammaticale et ainsi de proposer seulement des montants grammaticalement corrects. Dans le cas de l'application à la reconnaissance des adresses le fonctionnement est similaire : la reconnaissance se fait au niveau du mot et non pas de la phrase entière.

5. Organisation générale d'un système de reconnaissance

L'objectif de la reconnaissance de l'écriture manuscrite est de développer un système qui se rapproche le plus de l'être humain dans sa capacité de lire. Cependant, cette reconnaissance de l'écriture consiste à extraire d'une forme inconnue (mot, lettres, chiffres) une description plus simple et à établir sur celle-ci une décision. Cette décision est effectuée généralement en mesurant la ressemblance d'une forme inconnue avec un ensemble de références stockées en mémoire et décrites dans une représentation analogue. Les références sont obtenues lors d'une phase antérieure qualifiée d'apprentissage. Cette phase est très importante dans tout système de reconnaissance de l'écriture. Autrement dit c'est un passage de l'espace observable vers un espace de décision d'appartenance à une classe.

La construction d'un système de reconnaissance de l'écriture comprend plusieurs étapes distinctes représentées par la *Figure II.1*.

L'entrée du système acquière soit une forme ou un signal par un capteur (appareil photo, scanner, etc.). Cette entrée peut être une image scannée ou un photol qui sera stocké dans un fichier. Par la suite plusieurs traitements sont opérés sur ces images et ces fichiers. Le but de ces pré-traitements est d'éliminer les phénomènes qui

provoquent une dégradation des performances du système, de réduire les bruits de quantification (binarisation) et de préserver la connexité des composantes connexes dans l'image. Le résultat de cette phase permettra d'extraire ou de mettre en évidence des particularités locales ou globales. Cette étape permet d'engendrer pour chaque image, un vecteur de primitives qui sert comme entrée au module responsable de la classification.

Dans la section suivante nous abordons l'état de l'art de chaque phase de la reconnaissance à savoir : les pré-traitements, l'extraction des primitives et la classification.

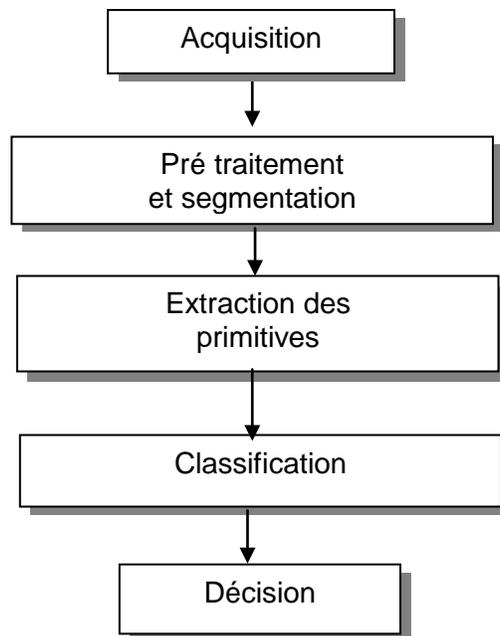


Figure II.1 : Étapes de reconnaissance d'écriture.

5.1 Prétraitements

L'objectif des prétraitements est de faciliter la caractérisation de la forme (caractère, chiffre, mot) ou de l'entité à reconnaître soit en nettoyant l'image représentant la forme ou en réduisant la quantité d'information à traiter pour ne garder que les informations les plus significatives. Le nettoyage de l'image consiste essentiellement à éliminer les bruits résiduels issus de la binarisation. La réduction de la quantité d'information à traiter peut être obtenue à partir des opérations visant à

ramener l'épaisseur du trait à un seul pixel (soit par squelettisation, soit par suivi de trait) [ou à partir d'extracteurs de contours supérieurs, inférieurs et/ou intérieurs.

Les différentes étapes des pré-traitements sont :

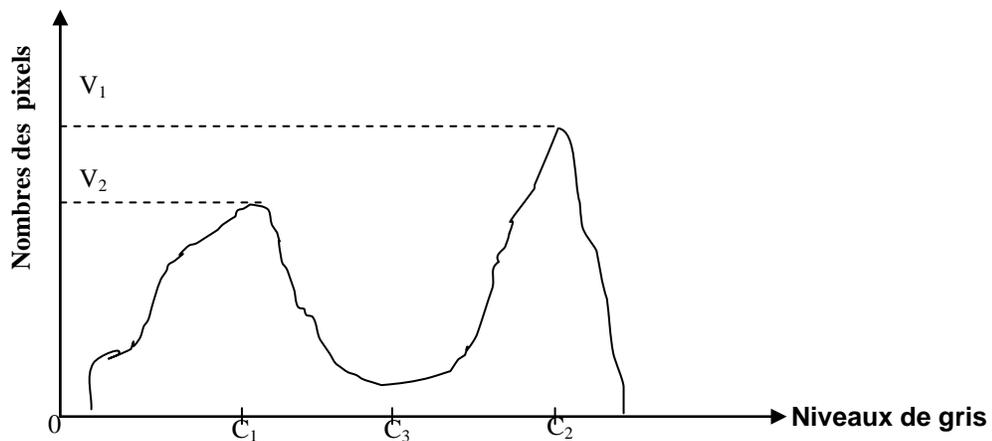
5.1.1 Binarisation :

Pour les images acquises en niveaux de gris, la binarisation devient nécessaire avant d'attaquer la phase du traitement. La binarisation permet de mieux distinguer les caractères du fond, elle consiste à attribuer à chaque pixel de l'image une valeur de 0 ou 1 : '0' qui représente le noir (le texte), et le '1' représente le blanc (le fond de la page). Pour cela, elle applique en premier lieu l'opération de seuillage [16].

Seuillage : Il consiste à déterminer la valeur du seuil à partir duquel tous les pixels ayant un niveau de gris inférieur à cette valeur sont représenté par un zéros "0" le noir, et tout les pixels de niveau de gris supérieur auront la valeur un "1" (le blanc).

La valeur du seuil est déterminée à partir de l'histogramme de niveau de gris de tous les pixels de l'image, elle sera égale à la valeur du niveau de gris se trouvant dans la vallée entre les deux pics de l'histogramme *Figure II.2*.

Le seuil doit être calculé d'une manière adéquate, car les composantes du texte liées des traits fins peuvent se déconnecter, ce qui modifie la forme originale du texte.



C_1 et C_2 : sont les niveaux de gris des pics, C_3 : le niveau de gris de seuil

Figure II.2 : Histogramme de niveaux de gris.

5.1.2 Lissage :

L'image de caractère peut être entachée de bruit dont l'origine est la qualité du document ou le mode d'acquisition utilisé, ce qui conduit soit à des absences de points (trous), soit des points en plus (des empâtements). Les techniques de lissage permettent de résoudre de résoudre ces problèmes par des opérations locales appelées *Nettoyage* et *bouchage*.

Nous utilisons les notions suivantes pour la description des masques de transformation de voisinage : '1' pour un point de la forme et '0' pour un point du fond et X pour un point quelconque.

L'opération de *nettoyage* conduit à supprimer les petites tâches et les excroissances de la forme. Elle est réalisée de différentes manières suivant le type de bruit à enlever [12] :

- Par élimination des points de la forme isolés ou situés à l'extrémité des contours, en appliquant sur l'image le masque suivant dont les huit directions

0	0	0
X	1	0
0	0	0

- Par élimination des points formant des angles droites ou des ex croisement du contour, en appliquant le masque suivant dans le huit directions :

X	0	0
1	1	0
0	0	0

Pour le *bouchage*, il s'agit d'égaliser les contours et de boucher les trous internes à la forme des caractères en lui ajoutant des points noirs on distingue :

- Le bouchage de trous isolés. Si le voisinage d'un point du fond correspond au masque suivant, alors ce point est mis à 1.

1	1	1
1	0	1
1	1	1

5.1.3 La correction de l'inclinaison :

Cette opération consiste à corriger la pente d'un mot ou à redresser l'inclinaison des lettres dans un mot afin de faciliter la *segmentation*. L'opération est effectuée à l'aide d'une transformation ligne par ligne où chaque pixel noir de coordonnées (x,y) est remplacé par les coordonnées (x',y') données par :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & -\tan \theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

II.1

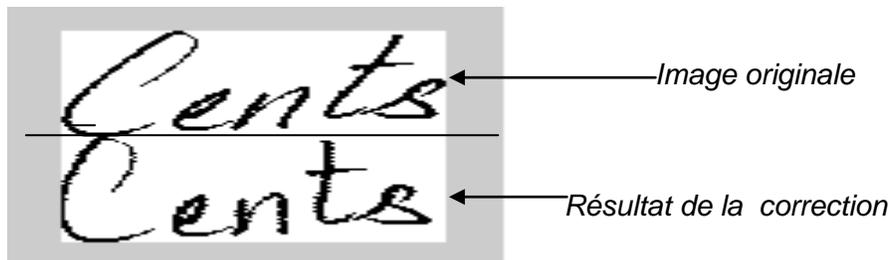


Figure II.3 : Correction de l'inclinaison.

5.1.4 Normalisation de la taille :

La taille des caractères peut varier d'une fonte à l'autre, et même au sein d'une même fonte après agrandissement ou réduction, ce qui peut causer une instabilité des paramètres [12].

Une technique de prétraitement consiste à ramener les caractères à la même taille, tel que l'algorithme de **Sihari**. [15]. Cet algorithme opère en deux étapes. La première normalise le caractère en hauteur et la seconde, en largeur. L'ordre de normalisation ainsi choisi évite que les caractères fins ne se déforment pas.

Il s'agit de transformer l'image du caractère de dimensions h_i, l_i en une image de dimensions h, l . Une étape intermédiaire consiste à produire une image de dimension h', l' , avec :

$$p = \frac{h}{h_i} \text{ et } l' = p.l$$

II.2

La normalisation en hauteur est exécutée en transformant chaque pixel (x, y) noir de l'image du caractère en $(p.x, p.y)$. La normalisation en largeur de l'image ainsi obtenue se fait par l'examen de deux cas :

Si $l' < l$, alors l'image normalisée en hauteur est centrée dans une surface de dimension h, l .

Si $l' > l$, chaque pixel noir de l'image (x, y) subit la transformation en $((x/l'), l, y)$.

5.1.5 Extraction des contours

On appelle le point de contour, un pixel qui se situe sur la frontière séparant deux régions voisines. Dans les images à niveaux de gris, il est intéressant d'extraire le contour à l'aide d'un calcul de *gradient*. Ce contour est alors d'autant plus marqué que le niveau des pixels résultant du gradient est élevé. En revanche, dans les images binaires, il est plus avantageux d'utiliser un algorithme de *suivi de contour* car il fournit directement une liste ordonnée de points.

-a) Le gradient :

Le contour se manifeste dans l'image par des variations locales importantes des valeurs de niveaux de gris mis en évidence par des élévations de la dérivée première de la fonction image.

Le gradient d'une fonction $f(x, y)$ est défini par

$$\nabla f(x, y) = \frac{\partial f}{\partial x} \vec{i} + \frac{\partial f}{\partial y} \vec{j} \quad \text{II.3}$$

\vec{i} et \vec{j} sont les vecteurs unitaires sur x et y .

Dans le domaine discret les dérivées suivant x et y peuvent être exprimées par les approximations suivantes :

$$A_x = f(x+1, y) - f(x, y) \quad \text{suivant } x$$

$$A_y = f(x, y+1) - f(x, y) \quad \text{suivant } y$$

L'amplitude du gradient est donc

$$A = \sqrt{A_x^2 + A_y^2} \quad \text{II.4}$$

La *Figure II.4* montre l'exemple d'un mot avant et après l'extraction de ses contours.

-b) Procédure de suivi de contour

L'algorithme de traçage de contour utilise le concept de 8-voisinage, ou 4-voisinage. Il peut être décrit comme étant un observateur qui traverse les pixels de l'image du texte (la composante connexe) d'un pixel à un autre, et sélectionne le plus à droite.



Figure II.4 : Exemple du mot « vingt » avant et après l'extraction de contour.

Le point de départ peut être détecté en parcourant la composante de droite à gauche haut en bas. L'algorithme s'achève quand le pixel courant coïncide avec celui de départ. (L'algorithme est détaillé dans [16]. Ce type d'algorithme retrouve seulement le contour extérieur de la composante connexe *Figure II.5*.)

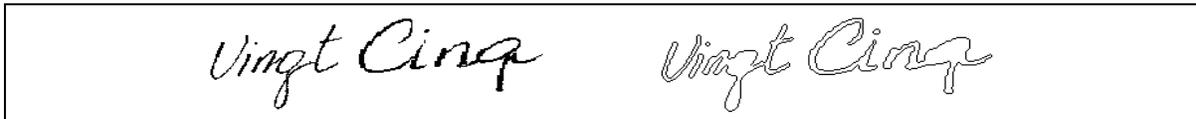


figure II.5 : Exemple du mot « vingt cinq » avant et après l'extraction de contours extérieurs.

Le contour est également utilisé comme étape préalable à un changement de représentation de l'information, en tant qu'empreinte des formes contenant une quantité réduite de données.

5.1.6 Extraction des composantes connexes :

L'extraction des composantes connexes, procédure également appelée capture des connexités ou étiquetage des pixels, est largement utilisée en Reconnaissance des Formes (RdF) pour segmenter les images binaires. La technique consiste à regrouper les pixels voisins dans un ensemble appelé composante connexe. Chaque ensemble

est disjoint des autres et peut ensuite être aisément isolé. La 4-connexité est distinguée de la 8-connexité suivant que le critère de voisinage comprend les 4 ou les 8 voisins d'un pixel.

Il existe deux principaux algorithmes pour accomplir cette tâche :

- le premier est basé sur une procédure de suivi de contour: en parcourant le contour d'un objet et en revenant au point de départ, une composante connexe est délimitée, à l'exclusion cependant des contours intérieurs correspondant aux éventuels trous.
- le second algorithme procède par une propagation d'un étiquetage des pixels lorsque l'on effectue un balayage des lignes et des colonnes de l'image.

Nous avons élaboré un algorithme de ce type fonctionnant en une seule passe, suivant le critère de 4-connexité. La propagation de l'étiquette des pixels suivant les colonnes (verticalement) est prioritaire sur la propagation suivant les lignes (horizontalement). On procède de la manière suivante: En parcourant une ligne horizontale de gauche à droite, on associe un numéro (une étiquette) à chaque pixel de telle sorte que tous les pixels voisins portent le même numéro (le numéro zéro est réservé pour un pixel "vide"). Lorsque sur cette ligne, le voisinage est interrompu, puis reprend plus loin, le numéro est incrémenté de 1. Les étiquettes sont représentées par les lettres A, B et C sur la *Figure II.6a*.

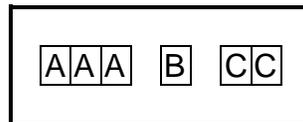


Figure II.6a : Propagation de l'étiquetage des pixels de gauche à droite sur une ligne horizontale.

Lorsqu'une nouvelle ligne est commencée, on propage naturellement l'étiquetage de haut en bas en recopiant le numéro du pixel qui se trouve au-dessus du premier pixel de la nouvelle ligne. S'il n'y a pas de pixel au-dessus, un nouveau numéro est utilisé, *Figure II.6b*.



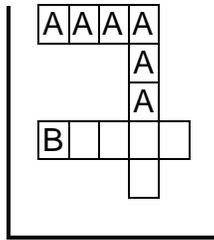


Figure II.6b : Propagation de l'étiquetage des pixels de haut en bas sur une colonne verticale.

Lorsqu'un conflit se présente entre la propagation horizontale et la propagation verticale des étiquettes, deux cas se présentent alors *Figure II.6c* :

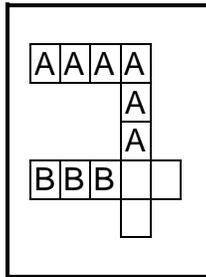


Figure II.6c : Conflit entre la propagation horizontale et verticale.

1^{er} cas : si le numéro des pixels horizontaux correspond à une nouvelle étiquette, alors il est facile de résoudre le conflit en remplaçant la nouvelle étiquette de tous les pixels à gauche du point de conflit par le numéro prioritaire du pixel de la ligne précédente *Figure II.6d*. La nouvelle étiquette est alors annulée ;

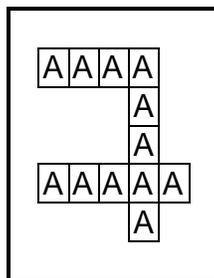


Figure II.6d : Résolution du conflit correspondant au 1^{er} cas.

2^e cas : Si le numéro des pixels horizontaux a déjà été propagé à partir de la ligne précédente, il serait trop long de remplacer tous les pixels correspondants. Aussi, on note dans un tableau que les deux étiquettes en conflit désignent une unique composante connexe *Figure II.6e*.

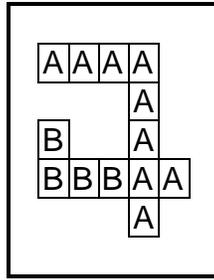


Figure II.6e : Résolution du conflit correspondant au 2^e cas.

5.2 Segmentation

Après la phase de prétraitement, plusieurs systèmes de reconnaissance divisent la page du texte en lignes, chaque ligne en mot, et certains systèmes divisent le mot en caractères ou en segments de caractères dans le but d'aborder les phases ultérieures, ce découpage est appelé segmentation.

Elle est définie comme étant un processus qui consiste à diviser l'image en régions à nuance rapprochées, l'idée de base est de regrouper dans la même région les pixels possédants la même valeur de niveau de gris.

Pour la reconnaissance de l'écriture la segmentation se fait séquentiellement en deux phases : décomposition des pages en lignes pour *extraire les mots* et *l'extraction des caractères*. Pour cela nous avons appliqué des méthodes de segmentation basée sur

- L'analyse de l'histogramme horizontal et vertical.
- Les minima des contours extérieurs.
- L'extraction des composantes connexes

a)- Extraction des mots : elle regroupe deux étapes :

- *Séparation des lignes* : localisation des lignes de texte.
- *Séparation des mots* : séparation des lignes en éléments constitutifs.

B)- Extraction des caractères :

- *La segmentation caractères* : séparation des parties connexes du mot en caractères.

5.2.1 Extraction des mots :

Après l'étape d'acquisition et prétraitement, l'image obtenue est une matrice binaire. L'extraction des mots se fait en deux étapes distinctes : une séparation de

lignes, suivie par une séparation de mots. La séparation de lignes se fait par projection horizontale qui n'est rien d'autre qu'une simple somme du nombre de points allumés par ligne, La séparation de mots se fait par projection verticale, une simple somme du nombre de points allumés par colonne.

Les projections horizontale et verticale sont définies par les équations suivantes :

$$h(i) = \sum_j g(i, j) \quad : \text{ Pour la projection horizontale.}$$

$$v(i) = \sum_i g(i, j) \quad : \text{ Pour la projection verticale.}$$

Où $g(i, j)$ vaut 0 ou 1.

-a) Séparation des lignes

La détection des lignes de texte commence une fois que la projection horizontale de toutes les lignes d'image est faite.

On peut dire le début ou la fin d'une ligne de texte est détectée, si la valeur de projection horizontale (P.H) *Figure II.7*, est inférieure à un seuil.

Le seuil est obtenu au minimum de l'histogramme (la fonction h). La confirmation de cette détection se fait si le nombre de lignes d'image détectées adjacentes est lui même supérieur à un autre seuil. C'est de cette façon qu'on confirme l'existence d'une ligne de texte.

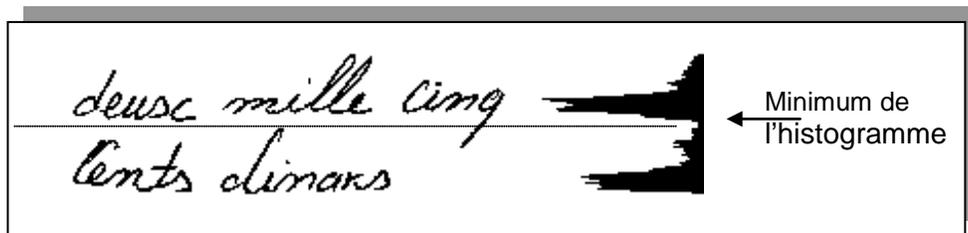


Figure II.7 : Histogramme de la segmentation lignes.

- b) La séparation des mots

La majorité des études consacrées à la segmentation des mots dans la phrase reposent sur l'hypothèse d'existence d'un espace inter-mots toujours supérieur à l'espace inter-lettres ou à la largeur moyenne d'une lettre.

La segmentation des mots permet de localiser les bornes de chaque mot de la ligne isolée, les bornes des mots peuvent être déterminées en effectuant une projection verticale et en cherchant l'espace noir qui est représenté par les pixels de valeur

supérieure à zéro, le début et la fin de la suite de ces espaces noirs entre deux espaces blancs représente les bornes, *Figure II.8*.



Figure II.8 : Histogramme de la segmentation mots

5.2.2 Extraction des caractères :

Les études dans le domaine de la segmentation du mot en caractère sont principalement basées sur l'analyse du contour. Les critères de segmentation que nous avons répertoriés sont liés à la détection des minimums du contour supérieur du mot *figure II.9 (c)*. Les hypothèses de base sont les suivantes :

- les caractères composant le mot ne sont reliés que par un seul trait ;
- la segmentation peut être effectuée par une coupure verticale *figure II.9 (d)* située au minimum local du trait de connexion suivi par une procédure de capture des composantes connexes.

La détection de ce minimum local est obtenue par le calcul de la dérivée du contour, c'est-à-dire, dans l'espace discrétisé, de la différence des ordonnées de deux points d'abscisses consécutives. Puis le minimum est validé en fonction des critères d'unicité du trait de connexion et de son épaisseur, afin de ne pas couper un caractère comportant un minimum local à l'intérieur de son tracé.

Afin de renforcer les hypothèses de base, plusieurs autres critères ont été proposés dans différentes études incluant la segmentation des mots manuscrits: On peut aussi appliquer la technique des composantes connexes, procédure également appelée capture des connexités ou étiquetage des pixels.

La technique consiste à regrouper les pixels voisins dans un ensemble appelé composante connexe. Chaque ensemble est disjoint des autres et peut ensuite être aisément isolé *Figure II.9 (e)*.

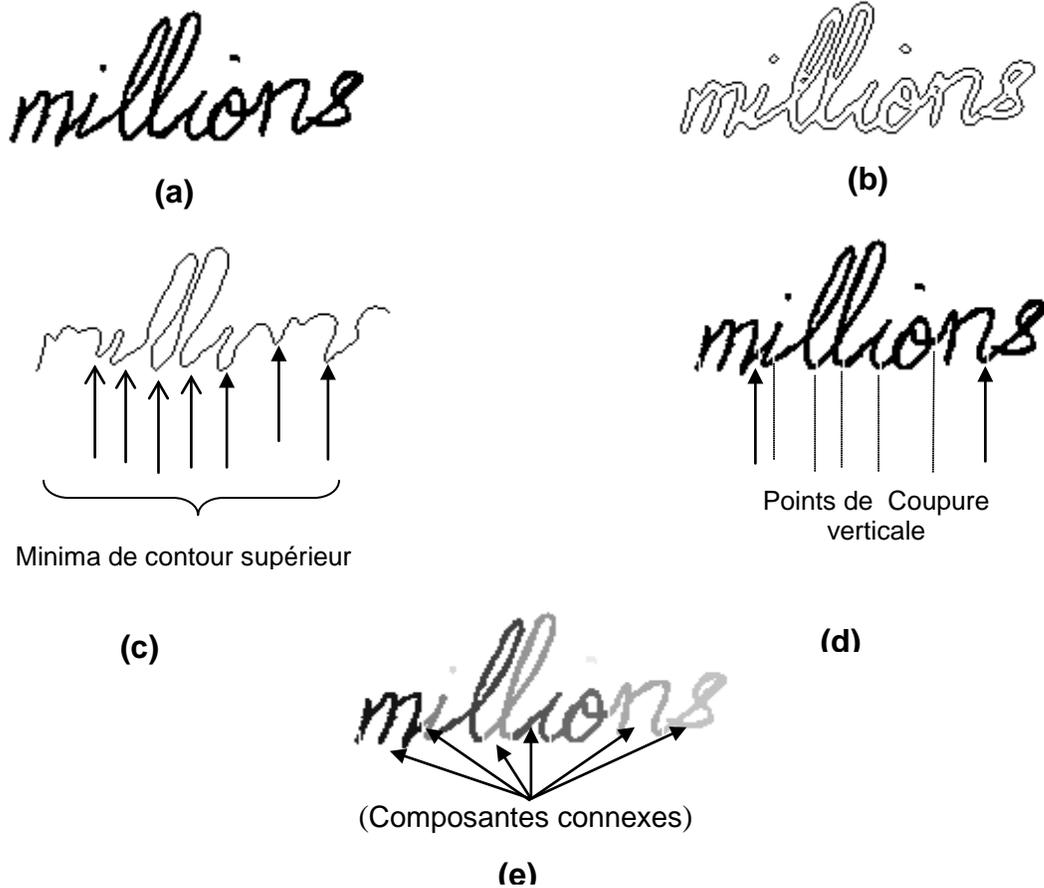


Figure II.9 Extraction des caractères - a)- Mot ' millions'

- b) Le contour du mot
- c) contour supérieur
- d) Détection des points de coupure
- e) Extraction des caractères

5.2.3 Les alignements dans le mot

5.2.3.1 Détermination de la zone médiane du mot

Dans les méthodes comportant une segmentation explicite du mot en lettres (méthodes de reconnaissance analytique), la première étape est généralement la détermination de la zone médiane du mot, c'est-à-dire, par convention dans la suite du texte, la zone qui permet de distinguer les lettres à hampes ou à jambages, de celles qui en sont dépourvues. La méthode la plus simple utilisée est basée sur l'analyse de l'histogramme horizontal de l'image.

L'histogramme permet de mettre aisément en évidence la zone médiane du mot car la contribution des minuscules sans hampe ni jambage y est déterminante.

On cherche dans un premier temps la ligne de base où la ligne de l'écriture. On calcule la somme $S(i)$ suivante :

$$S(i) = \sum h(j) \quad \text{II.5}$$

L'indice i correspondant à la ligne de base où la somme $S(i)$ est maximum, est noté M . Dans la plupart des cas, cette ligne d'indice i se trouve à l'intérieur de la zone médiane du mot, même si elle est parfois plus près d'un bord de la zone que de l'autre.

Dans un deuxième temps, on recherche dans la partie supérieure à la ligne d'indice M ainsi que dans la partie inférieure, les indices des minima de l'histogramme respectivement m_h et m_b . Dans le cas idéal, ces deux minima délimitent la zone médiane. Mais, en pratique, l'histogramme est souvent étalé et dessine une lente décroissance autour de la zone médiane. On recherche donc plutôt les maxima de la dérivée de la fonction h afin d'obtenir l'indice pour lequel la variation est la plus importante.

La ligne de séparation entre les hampes et la zone médiane est obtenue au maximum de la dérivée de la fonction h_i compris entre m_h et M , tandis que la ligne de séparation entre la zone médiane et les jambages est obtenue au maximum de la valeur absolue de la dérivée de h_i compris entre M et m_b .

La *Figure II.10* illustre la détection de ces paramètres sur le mot "quatre".

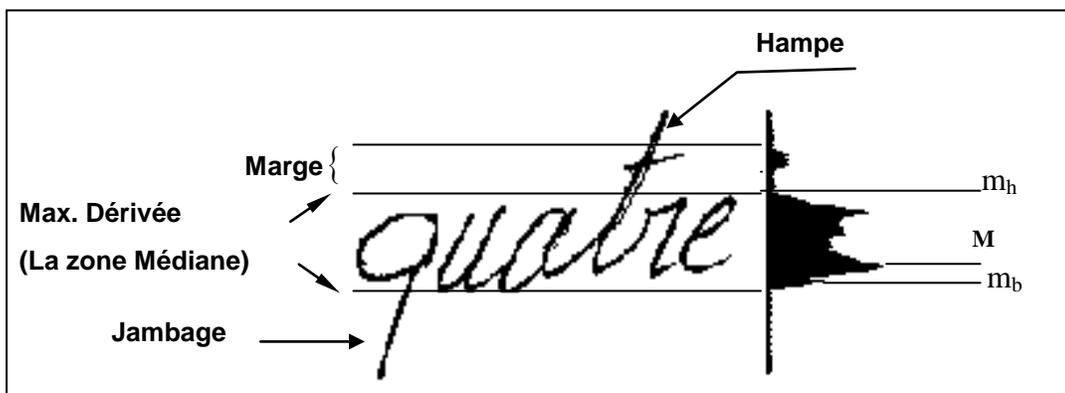


Figure II.10 : Les alignements dans le mot.

Lorsque la zone médiane du mot est obtenue, les lettres à hampe ou à jambage sont distinguées des lettres médianes quand elles dépassent les limites de la zone médiane d'une grandeur supérieure à une marge fixée.

5.3 Extraction des primitives (caractéristiques)

Un des problèmes fondamentaux de la reconnaissance d'écritures est de déterminer quelles caractéristiques employer pour obtenir un bon résultat de classification.

L'objectif de l'extraction et de la sélection des caractéristiques est d'identifier les caractéristiques qui sont importantes dans la discrimination de classes de formes. Cela signifie de trouver des primitives les plus robustes possibles, les plus pertinentes (afin de diminuer la taille du vecteur de primitives) et les moins coûteuse en calcul.

Le résultat de cette phase d'extraction de primitives est une séquence de symboles représente l'image binaire. Cette séquence est représentée sous forme d'un vecteur qui constituera les données d'entrée de la phase de reconnaissance.

En fonction de l'objectif fixé et de la méthode d'extraction choisie, l'approche de l'extraction des primitives peut être systématique ou heuristique.

La modélisation et le codage conduisent à une approche systématique dans la mesure où l'objectif fixé est la détermination d'une représentation complète de la forme, même de façon approximative. Dans la modélisation, les primitives sont obtenues a posteriori, par le résultat de l'approximation, tandis que, en ce qui concerne le codage, les catégories de primitives sont définies a priori. Un test, qui est par exemple réalisé à l'aide d'une sonde, permet de valider la présence de chacune des primitives sur l'ensemble de la forme.

Le paramétrage conduit plutôt à une approche heuristique. Dans ce cas, on ne cherche pas nécessairement une représentation complète mais seulement des indices significatifs. De même que dans le cas du codage, ces indices sont des primitives définies a priori.

Au-delà de cette classification un peu formelle, la différence entre les approches systématique et heuristique comme entre le caractère a priori ou a posteriori, s'avère plus nuancée dans la pratique.

Les primitives sont classées en trois catégories qui sont *géométrique, structurelle, et statistique*.

-a) Caractéristiques géométriques :

Les mesures géométriques ou physiques sont généralement des mesures de distance entre points appartenant ou non au mot et qui permettent de caractériser ses propriétés métriques.

Parmi les caractéristiques géométriques les plus utilisées on retrouve :

- Les moments géométriques, invariants en changement d'échelles, rotation et translation.
- Descripteur de Fourier ces caractéristiques sont basées sur des informations extraites à partir de l'image du contour
- Description locale des contours (Codage des contours) : est une technique exprimant les frontières discrètes de l'image par une séquence de code comme la chaîne de Freeman spécifiant la direction de déplacement d'un vertex à un autre.
- Description des profils des caractères

-b) Caractéristiques structurelles :

Les caractéristiques structurelles décrivent le mot dans sa globalité, parmi les plus utilisées dans la reconnaissance globale de mots on trouve :

- Les hampes et les jambages qui sont généralement des primitives recherchées dans la perception humaine.
- Les boucles et les cavités qui permettent aussi de distinguer entre les mots qui possèdent des primitives presque similaires.
- Les points diacritiques, car ils permettent de distinguer les caractères ayant le même corps principal.

-c) Caractéristiques Statistiques :

Les mesures de densité de pixels noirs ou blancs sont généralement utilisées pour décrire l'influence de certaines parties du mot et donc la présence éventuelle de traits caractéristiques. La plus part des méthodes utilisant ce type de mesures

procèdent à un découpage préalable de l'image en zones et ce sont les densités de pixels dans chaque zone qui forment les composantes du vecteur de caractéristiques.

5.3.1 Les caractéristiques adaptées à l'écriture

La détection des caractéristiques est l'une des étapes les plus délicates dans la construction d'un système de reconnaissance de caractères manuscrits.

Nous présentons dans cette partie les principales caractéristiques et leurs détections

5.3.1.1 Détection des caractéristiques structurelles

5.3.1.1.1 Localisation des hampes et jambages :

Grâce à la connaissance de la zone médiane du mot, chaque caractère est identifié à une classe générique parmi cinq.

Les cinq classes génériques sont notées 'H', 'J', 'f', 'M', et '?' respectivement Hampes, Jambages, hampes et jambages, Médiants et ambigus *Figure II.11*.

Une hampe est détectée si l'ordonnée supérieure du caractère est située dans la zone supérieure à la zone médiane du mot à une distance supérieure à la marge (position absolue).

Un jambage est détecté si l'ordonnée inférieure du caractère est située dans la zone inférieure à la zone médiane du mot à une distance supérieure à la marge (position absolue).

Le caractère est classé Médian si sa hauteur est inférieure à celle de la zone médiane du mot (hauteur relative), à condition toutefois que le graphème ne dépasse pas les zones définies par les marges (position absolue).

Sinon, si le caractère ne vérifie aucune de ces conditions, il est classé ambigu.

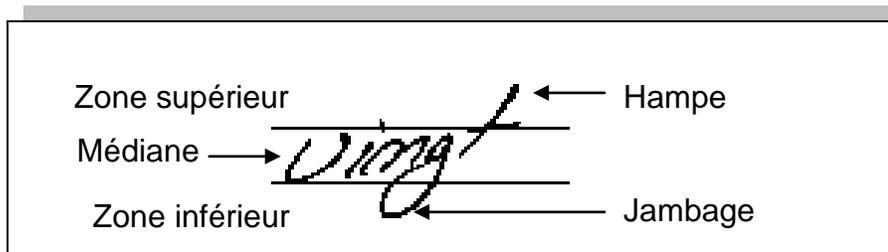


Figure II.11 : Exemples de Localisation des hampes, jambages et médiane du mot 'vingt'.

5.3.1.1.2 Les boucles :

Il y a deux méthodes permettant d'extraire les boucles : celles qui procèdent par l'étiquetage des zones blanches de l'image et celle qui opèrent sur le contour de la forme.

L'analyse des boucles par étiquetage (la connexité) des zones blanches de l'image consiste à associer à chaque zone blanche une étiquette, ensuite une procédure de remplissage par des zéros (noir) de toutes les zones ayant en contact avec les limites du cadre de l'image *Figure II.12*.



Figure II.12 : Extraction des boucles méthode d'étiquetage.

La méthode de détection qui opère sur les contours du caractère nécessite une procédure de suivi de contour, en parcourant le contour d'un objet et en revenant au point de départ, une composante connexe est délimitée, à l'exclusion cependant des contours intérieurs correspondant aux éventuels trous *Figure II.13*.



Figure II.13 : Extraction des boucles méthode basée sur les contours.

5.3.1.2 Détection des caractéristiques statistiques

5.3.1.2.1 Caractéristiques issues de la technique de zonage « ZONING »

Une information globale caractérisant la répartition des points constituant le caractère. Dans la fenêtre minimale est également exploitée. La fenêtre englobant le

caractère ou le mot étudié est divisée en régions de façons différentes *Figure II.14*. Pour chaque découpage, on détermine la répartition des points de l'image entre les régions constituant ce découpage.

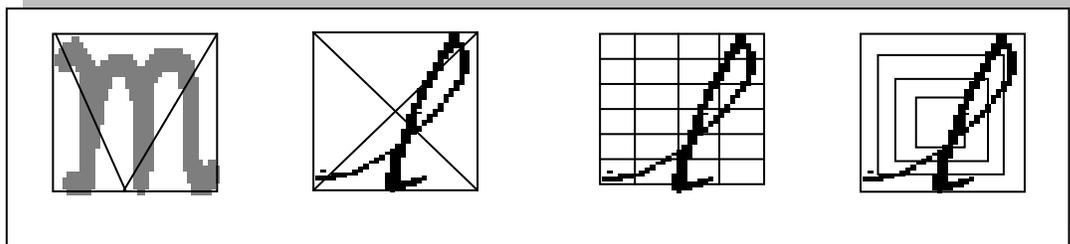


Figure II.14 : Division de l'image en des zones de quatre manières.

5.3.1.3 Détection des caractéristiques géométriques

5.3.1.3.1 Codage des contours :

Le codage des contours est une technique exprimant les frontières discrètes de l'image du caractère par une séquence de code ou une suite de symboles. Un premier type de symboles donne la position absolue d'un point appartenant au contour appelé *point de départ*. Un deuxième type décrit la position relative de tous les autres points des contours *Figure II.15* pris séquentiellement dans un sens défini arbitrairement [20].

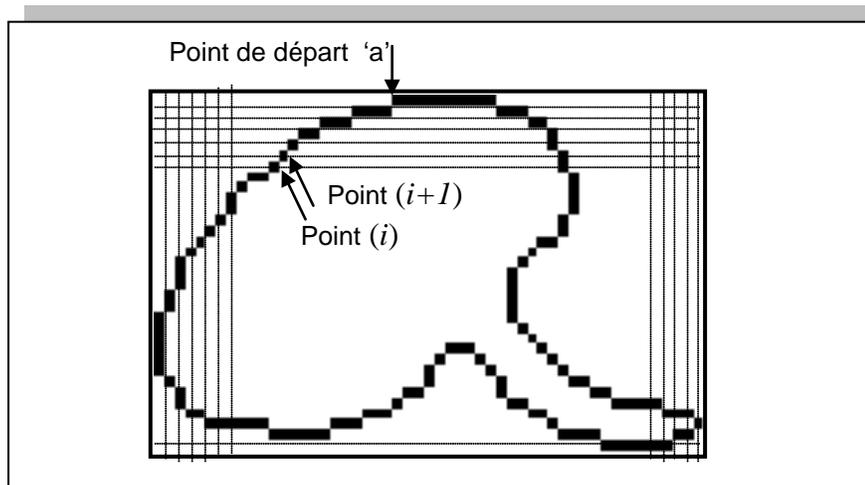


Figure II.15 : Représentation d'un contour de la lettre « a » au moyen d'une grille.

Il existe une technique qui permet de définir la position relative d'un ensemble des points à savoir le code de chaîne ou codage de Freeman.

Dans ce code la position d'un nouveau point voisin $i + 1$ par rapport au point i , connexe de connectivité 4 ou 8, s'exprime au moyen d'un mot binaire à 2 ou à 3 bits.

Le nombre spécifique indiqué par le mot binaire correspond à l'une des directions qui relie les deux points comme indiqué à la *figure II.16*.

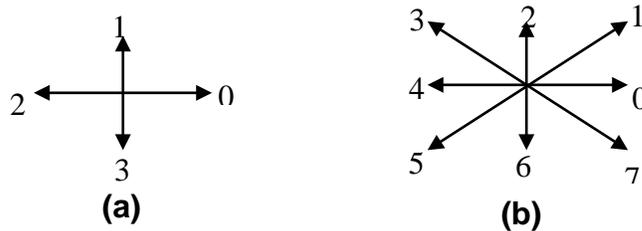


Figure II.16: Définition du code de chaîne : (a) Code de chaîne de connectivité 4.
(b) code de chaîne de connectivité 8.

A titre d'exemple, le code de chaîne du contour illustré à la *Figure II.15* de connectivité 8 en prenant le point 'a' comme point de départ et le sens des aiguilles d'une montre comme sens de parcours est :

$C_8=00000070077767666655545665545454454453444444434433322222221226666677770707$
 $00070000765444544344434433433334455000000121111111111010100010000007007776766$
 $66555456600000012111111111101010001.$

5.3.1.3.2 Extraction des profils

L'utilisation des profils pour caractériser les caractères manuscrits a été suggérée par divers auteurs ([17] par exemple). Le caractère est encadré. Dans une fenêtre minimale. Le profil compte le nombre de pixels (distance) entre la boîte de bondissement de l'image de caractère et le bord du caractère. Le profil d'un caractère peut être pris à n'importe quelle position, gauche, droite, supérieur, inférieur et orienté *Figure II.17*.

Les profils décrivent bien les formes externes des caractères et laissent distinguer entre un grand nombre de lettres.

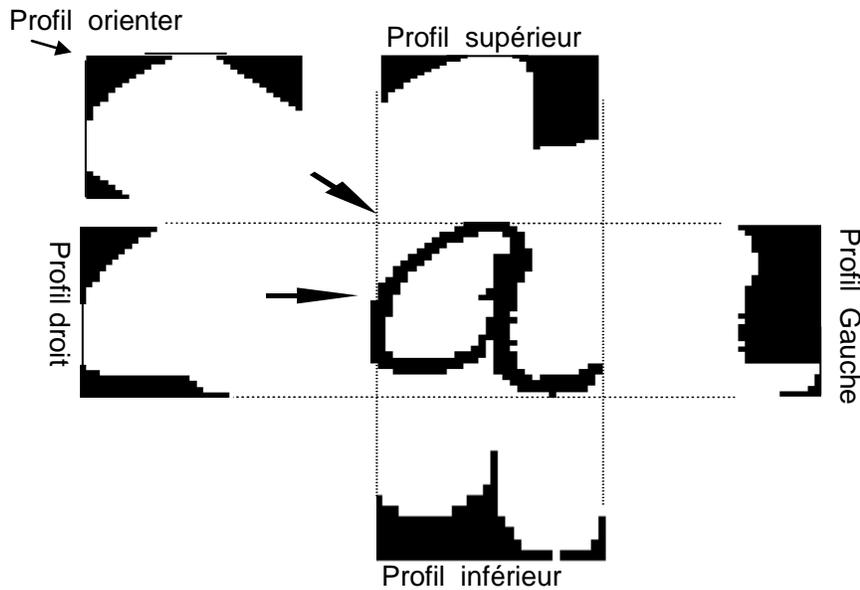


Figure II.17 : Exemple des différentes projections des Profils pour la lettre 'a'.

5.3.1.3.4 Description de formes au moyen de moments invariants

La méthode des moments géométriques, comme toutes les méthodes statistiques, permet d'extraire des paramètres propres à la forme à reconnaître. Ces derniers permettent de la distinguer de toutes les autres formes qui lui sont Peu semblables. Les moments invariants ont le mérite de répondre à trois critères [18] qui sont.

- invariance par rapport à la translation,
- invariance par rapport au changement d'échelle,
- invariance par rapport à la rotation.

5.3.1.3.4.1 Méthode des moments

Les moments sont souvent utilisés en physique pour décrire la répartition des masses d'un corps en chaque point on peut prendre le même formalisme pour décrire la répartition des niveaux de gris dans un objet, dans le cas d'un objet binaire où $f(x, y)$ prend la valeur «1 »à l'intérieur et « 0 »à l'extérieur, les différents moments fournissent des informations importantes concernant l'arrangement spatial des points de l'objet.

L'équation générale des moments est la suivante :

$$m_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \quad \text{II.6}$$

Avec p et $q \in \mathbb{N}$ et $f(x, y)$ est le niveau de gris de l'image.

Le **centre de gravité** d'une image « \bar{x}, \bar{y} » est défini par les relations suivantes :

$$\bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

Où m_{00} est la surface de la forme dans le cas d'une image binaire.

Afin de rendre les moments m_{pq} invariants par rapport à translation, on les définit en choisissant « \bar{x}, \bar{y} » comme origine. On parle alors de moments centrés μ_{pq} dont la définition formelle est la suivante :

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x-\bar{x})^p (y-\bar{y})^q \cdot f(x, y) dx dy \quad \text{II.7}$$

Pour une image discrète les relations (II.5) et (II.6) deviennent :

$$m_{pq} = \sum_x \sum_y x^p y^q \cdot f(x, y) \quad \text{II.8}$$

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q \cdot f(x, y) \quad \text{II.9}$$

Ce qui donne pour p et $q=0,1$:

$$\mu_{00} = m_{00}$$

$$\mu_{10} = \mu_{01} = 0$$

$$\mu_{11} = m_{11} - (m_{00} \cdot \bar{x} \cdot \bar{y})$$

Les moments centrés normalisés sont définis par :

$$n_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad \text{avec } \gamma = \frac{p+q}{2} + 1 \quad \text{II.10}$$

Afin de rendre les moments φ_i invariant par rotation et grossissement (changement d'échelle), les moments suivants en été introduits :

$$\varphi_1 = n_{20} - n_{02}$$

$$\varphi_2 = (n_{20} - n_{02})^2 - 4 \cdot n_{11}^2$$

$$\varphi_3 = (n_{30} - 3 \cdot n_{12})^2 + (3 \cdot n_{11}^2 - n_{03})^2$$

$$\varphi_4 = (n_{30} - 3 \cdot n_{12})^2 + (3 \cdot n_{21}^2 - n_{03})^2$$

$$\begin{aligned}\varphi_5 &= (n_{30} - 3 \cdot n_{12})^2 \cdot (n_{30} - n_{12}) \left[(n_{30} - 3 \cdot n_{12})^2 - (3 \cdot n_{21} - n_{03})^2 \right] \\ &\quad + (n_{21} - n_{03}) \cdot (3 \cdot n_{11} - n_{03}) \left[3 \cdot (n_{30} + n_{12})^2 - (n_{21} + n_{03})^2 \right] \\ \varphi_6 &= (n_{20} - n_{02})^2 \left[(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2 \right] + 4 \cdot n_{11} \cdot (n_{30} + n_{12}) \cdot (n_{21} + n_{03}) \\ \varphi_7 &= (3 \cdot n_{21} - n_{03}) \cdot (n_{30} + n_{12}) \left[(n_{30} - 3 \cdot n_{12})^2 - (3 \cdot n_{21} - n_{03})^2 \right] \\ &\quad + 3 \cdot (n_{12} - n_{30}) \cdot (n_{21} + n_{03}) \left[3 \cdot (n_{30} + n_{12})^2 - (n_{21} + n_{03})^2 \right]\end{aligned}$$

On obtient finalement les moment invariants par translation, rotation et changement d'échelle appelé φ_i .

La *Figure II.18* montre une image de caractère 'z' déformés par rotation, translation et par changement d'échelle ainsi les moments φ_i ,

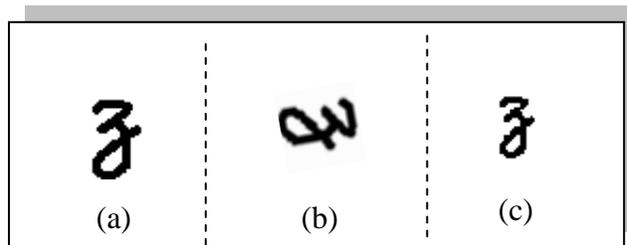


Figure II.18 : Opérations géométrique :

(a) Image originale.

(b) Image tournée de 80° .

(c) Image dont la taille est réduite par 50%.

Tableau II.1 Moments φ_i , $i=1, \dots, 7$ calculés sur les trois images décrites à la *figure II.16*

	Figure (a)	Figure (b)	Figure (c)
φ_1	0.371211	0.371211	0.385173
φ_2	0.053906	0.053906	0.063728
φ_3	0.003607	0.003607	0.003976
φ_4	0.000612	0.000612	0.000542
φ_5	-0.000000	-0.000000	-0.000000
φ_6	-0.000077	-0.000077	0.000000
φ_7	0.000000	0.000000	-0.000119

En observant les moments décrits dans le Tableau II.1, on constate une assez bonne propriété d'invariance.

5.3.1.3.5 Description de formes par descripteur de Fourier

L'analyse harmonique est l'instrument majeur de la théorie des signaux. La transformation de Fourier, généralisée par l'emploi des distributions, permet d'obtenir une représentation spectrale des signaux déterministes. Celle-ci exprime la répartition fréquentielle de l'amplitude, de la phase, de l'énergie ou de la puissance des signaux considérés.

L'analyse de Fourier consiste à décomposer un signal en une somme de signaux élémentaires particuliers, qui ont la propriété d'être faciles à mettre en œuvre et à observer.

Le descripteur de Fourier correspond à une suite de nombres dont le but est de caractériser au mieux la forme à décrire. Il existe principalement deux méthodes (angulaire et complexe) qui permettent de les définir.

5.3.1.3.5.1 Aspects théoriques de la TF

5.3.1.3.5.1.1 TF d'un signal réel

Considérons un signal monodimensionnel de type temporel. La TF de ce signal est définie de la façon suivante :

$$X(f) = \text{TF}[x(t)]$$

$$= \int_{-\infty}^{+\infty} x(t) \exp(-2\pi ft) dt$$

II.11

Les signaux traités dans cette étude sont codés sur leurs parties réelles seulement, leurs parties imaginaires étant fixées à zéro.

$$X(f) = \text{TF}[x(t)], \text{ avec } x(t) \text{ réel}$$

$$\begin{aligned} &= \int_{-\infty}^{+\infty} x(t) \cos(2\pi ft) dt - i \int_{-\infty}^{+\infty} x(t) \sin(2\pi ft) dt \\ &= \text{Re}\{X(f)\} + i \text{Im}\{X(f)\} \\ &= |X(f)| \exp(i V_x(f)). \end{aligned}$$

II.12

Le module, $|X(f)| = \sqrt{(\operatorname{Re}\{X(f)\})^2 + (\operatorname{Im}\{X(f)\})^2}$ appelé spectre d'amplitude, est une fonction paire. Le spectre de phase $V_x(f) = \arg(X(f))$, qui est une fonction impaire, sensible aux translations, ne sera pas pris en considération et nous nous limiterons donc à l'étude de $|X(f)|$ pour les seules fréquences positives.

5.3.1.3.5.1.2 TF discrète (TFD)

Soit T_e la cadence à laquelle on prélève N échantillons pendant la durée T d'un signal x ; en introduisant les simplifications de notation suivantes :

$$x(k) = x(kT_e/T)$$

$$X(n) = X(n/T * f_e)$$

$$W_N = \exp(2\pi i/N) \quad \text{avec } N \text{ pair.}$$

La Transformation de Fourier Discrète (TFD) est définie par :

$$X(n) = \sum_{k=N/2}^{N/2-1} x(k) W_N^{-nk}$$

II.13

Et la transformation inverse par :

$$x(k) = \sum_{n=N/2}^{N/2-1} X(n) W_N^{nk} \quad \text{II.14}$$

Où $x(k)$ est le vecteur du signal discret monodimensionnel et $X(n)$ sa TFD.

5.3.1.3.5.1.3 Définition complexe de descripteurs de Fourier

Après une opération de *suivi du contour* extérieur d'un objet, on obtient la suite ordonnée de ses N points représentée sous la forme complexe *figure II.19*.

Un point s'y déplaçant génère un signal monodimensionnel complexe $z(n)$ où n représente l'abscisse curviligne du contour[22].

$$z(n) = x(n) + jy(n) \quad \text{II.15}$$

x : représente la ligne,

y : représente le colonne.

Pour n variant de 0 à $N-1$.

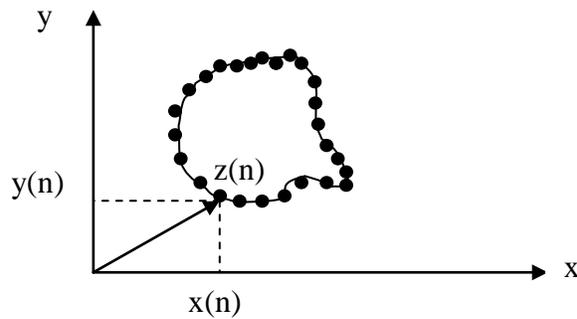


Figure II.19 : Représentation complexe d'un contour.

On remarque que cette fonction est périodique de période T , ce qui formalise par la relation suivante :

$$z(n+kT) = z(n) \quad : \text{avec } k \text{ entier}$$

Compte tenu de la nature périodique (de période N) de cette suite, on peut la représenter en utilisant la transformation de Fourier discrète (DFT) de l'équation II.13

$$z(n) = (1/N) \cdot \sum_{k=0}^{N-1} Z(k) e^{2j\pi kn/N}, \text{ pour } 0 \leq n \leq N-1. \quad \text{II.16}$$

$$Z(k) = \sum_{n=0}^{N-1} z(n) e^{-2j\pi kn/N}, \text{ pour } 0 \leq k \leq N-1. \quad \text{II.17}$$

Les coefficients $Z(k)$ ($k = 0, 1, \dots, N-1$) désignent les **descripteurs de Fourier** du contour. Après application de la FFT, le contour est décrit par les coefficients (descripteurs) de Fourier que l'on visualise (en module) sous forme de "raies".

La reconstruction du contour peut être réalisée par l'application de la transformée de Fourier inverse (IFFT) équation (II.17) aux coefficients.

On peut également effectuer une opération de "filtrage", par exemple en supprimant certains coefficients. Après transformée inverse, on obtient un contour fermé qui approxime plus ou moins bien le contour initial *Figure II.20*.

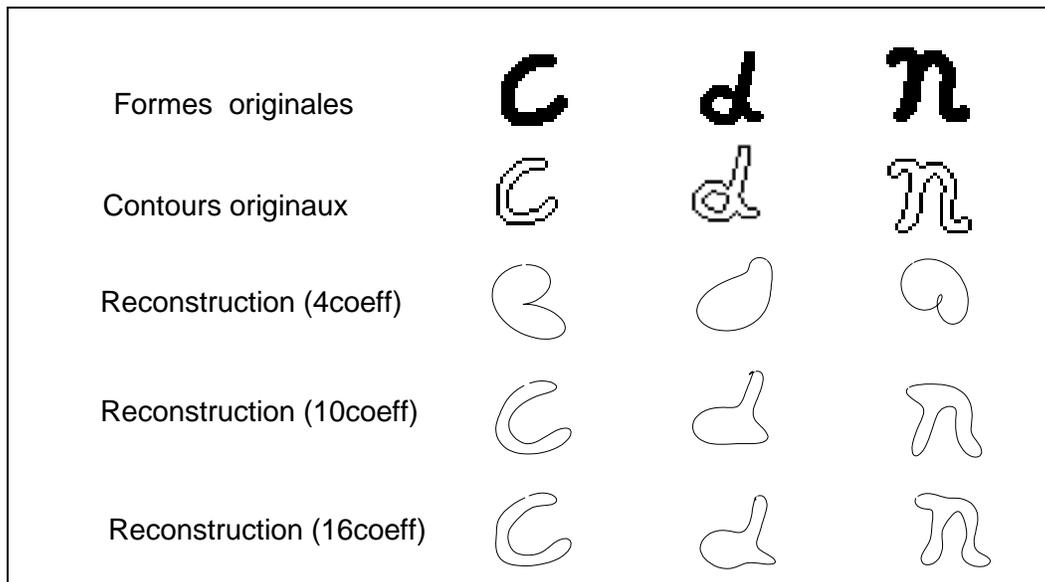


Figure II.20 : Reconstruction de la forme originale du contour extérieur.

5.4 Classification

La classification est l'élaboration d'une règle de décision qui transforme les attributs caractérisant les formes en appartenance à une classe (passage de l'espace de codage vers l'espace de décision). Avant qu'un modèle de décision ne soit intégré dans un système de reconnaissance de l'écriture, il faut avoir procédé auparavant à deux étapes : l'étape d'*apprentissage* et l'étape de *test*.

L'étape d'apprentissage consiste à caractériser les classes de formes de manière à bien distinguer les familles homogènes de formes. C'est une étape clé dans le système de reconnaissance. On distingue deux types d'apprentissage : apprentissage *supervisé* et apprentissage *non supervisé* [17].

Dans le cas de l'apprentissage supervisé, un échantillon représentatif de l'ensemble des formes à reconnaître est fourni au module d'apprentissage. Chaque forme est étiquetée par un opérateur appelé professeur, cette étiquette permet d'indiquer au module d'apprentissage la classe dans laquelle le professeur souhaite que la forme soit rangée. Cette phase d'apprentissage consiste à analyser les ressemblances entre les éléments d'une même classe et les dissemblances entre les éléments de classes différentes pour en déduire la meilleure partition de l'espace des représentations. Les paramètres

décrivant cette partition sont stockés dans une table d'apprentissage à laquelle le module de décision se référera ensuite pour classer les formes qui lui sont présentées.

Dans le cas de l'apprentissage non supervisé, on fournit au système de reconnaissance un grand nombre de formes non étiquetées. L'étape de la classification va se charger d'identifier automatiquement les formes appartenant à une même classe.

Dans le domaine de la reconnaissance de l'écriture, ce sont les méthodes basées sur un apprentissage supervisé qui sont le plus souvent utilisées ; et plus particulièrement pour les caractères manuscrits isolés car les classes naturelles sont connues et en nombre limité.

L'étape de test permet d'évaluer la performance du classifieur pour un apprentissage donné. C'est une étape importante car elle peut mettre en cause le choix des primitives ou le choix de la méthode d'apprentissage. En effet, il est difficile de trouver *a priori* les primitives pertinentes et la méthode d'apprentissage la plus adaptée au problème posé d'où l'utilité de procéder par itérations successives. Ces itérations consistent à extraire des primitives jugées utiles au problème de reconnaissance à résoudre et de tester la performance du système avec cet ensemble de primitives. Au fur et à mesure que les performances du système souhaitées ne sont pas atteintes alors il suffit de trouver à nouveau une nouvelle famille de primitives ou de combiner les primitives extraites avec de nouvelles primitives.

Ces phases d'apprentissage et de test sont réalisées préalablement à l'intégration du module de décision dans le système de reconnaissance. Dans tous les cas, on peut permettre au système de reconnaissance d'itérer les phases d'apprentissage et de test tant qu'on n'a pas atteint les performances désirées. Le calcul de cette performance est le résultat du classifieur utilisé. Pour construire un classifieur, il existe trois approches : structurelle, statistique et hybride.

L'approche réseau de neurones est une implémentation dérivée de l'approche statistique et de l'approche structurelle.

Dans notre cas, nous abordons les deux approches : structurelle et statistique en considérant que les réseaux de neurones font partie des approches statistiques.

5.4.1 Les approches de classification

5.4.1.1 Les approches statistiques

Cette approche consiste à déterminer des caractéristiques extraites d'une forme pour les caractériser d'une manière statistique. Elle a besoin d'un nombre élevé d'exemples afin de réaliser un apprentissage correct des lois de probabilité des différentes classes. Autrement dit, cette approche bénéficie des méthodes d'apprentissage automatique qui s'appuient sur des bases théoriques connues telles que la théorie de la décision bayésienne, les méthodes de classification non supervisées et l'analyse en composantes principales. Les deux principales familles de méthodes utilisées sont les méthodes *paramétriques* et les méthodes *non paramétriques*.

- Les méthodes paramétriques opèrent sous l'hypothèse que les classes étudiées suivent une distribution de probabilité d'une certaine forme connue *a priori*. La prise de décision consiste à déterminer la classe pour laquelle la forme inconnue présente la probabilité d'appartenance maximale. Elles exigent des bases d'apprentissage assez importantes pour une estimation correcte des paramètres de la distribution supposée. L'approche statistique englobe : la règle de Bayes [12], les méthodes neuronales et les chaînes de Markov [19].

- Les méthodes non paramétriques, les lois de probabilité sont inconnues pour une des classes. Le problème revient à établir des frontières de décision entre les classes. Les techniques les plus utilisées en reconnaissance de formes sont : la méthode du plus proche voisin [12], et la méthode d'appariement de graphes [12]. Pour de plus amples informations, Kunt [20] décrit un ensemble de méthodes statistiques en reconnaissance de formes.

Malgré leur nature différente, les approches statistiques et structurelles peuvent être combinées aux mêmes domaines d'application. Le choix d'une approche peut être lié à des contraintes matérielles telle que la taille de la base d'apprentissage disponible, le temps de calcul requis et la taille mémoire nécessaire. L'utilisation conjointe des deux approches peut être une solution optimale pour le problème de reconnaissance de l'écriture.

Dans le cadre de notre projet, la méthode de classification du système de reconnaissance d'écriture repose sur la théorie des réseaux de neurones, qui est présentée dans le chapitre III.

5.4.1.1 Les approches structurelles

Ces approches consistent à mettre en relation la structure des formes analysées et la syntaxe d'un langage formel. La description des formes est réalisée par l'intermédiaire de phrases et le problème de classification est ramené à un problème d'analyse de grammaire (*parsing*). De manière générale, les approches syntaxiques ou structurelles permettent la description de formes complexes à partir de formes élémentaires. Ces dernières, encore appelées caractéristiques, sont extraites directement des données présentes en entrée du système. La différence principale entre ces méthodes et les méthodes statistiques est que ces caractéristiques sont des formes élémentaires et non pas des mesures. Une autre différence est qu'elles introduisent la notion d'ordre dans la description d'une forme. Les méthodes les plus répandues utilisent le calcul de distance d'édition entre deux chaînes et la programmation dynamique.

5.4.1.2 Approche hybride

Une dernière approche est envisageable : l'approche hybride. La reconnaissance par modèles de Markov cachés en est une. En effet ils utilisent une approche statistique tout en ayant la possibilité d'utiliser des descriptions structurelles. Leur application dans le domaine de la reconnaissance de l'écriture est de plus en plus présente. [19] [21], après leur succès en reconnaissance de la parole. Une description complète de l'application de ces modèles à la reconnaissance de mots manuscrits est présentée dans [13]. Il existe également des approches couplant différentes techniques : modèle de Markov et réseaux de neurones , modèles de Markov cachés et k- plus proches voisins.

5.5 La décision

Les méthodes de reconnaissance vues précédemment sont utilisées pour décrire les règles de prise de décision, cette dernière peut être :

-**Caractère ou mot reconnu** : Si le système arrive à associer un et un seul prototype au caractère ou mot à reconnaître, il prend une décision unique.

-**Caractère ou mot ambigu** : Si le système associe plusieurs prototypes au caractère ou mot à reconnaître, il propose ainsi plusieurs choix avec une confiance de même ordre.

-**caractère ou mot rejeté** : Si le système n'arrive pas à associer un prototype au caractère ou mot à reconnaître : il ne prend aucune décision de classification. On définit ainsi plusieurs facteurs de performance de système tel que :

-*le taux de reconnaissance* : qui représente le pourcentage de caractères ou mot reconnus parmi ceux présentés au système.

-*le taux de substitution* : qui présente le pourcentage de caractères ou mots acceptés par le système mais classés de façon incorrecte (à cause d'un problème d'ambiguïté).

-*le taux de rejet* : qui représente le pourcentage de caractère ou mots rejetés (non reconnus) parmi ceux représentés à l'entrée du système.

5.5.1 Evaluation des performances d'un classifieur

Nous décrivons dans ce paragraphe les critères les plus souvent employés pour évaluer la performance des systèmes de reconnaissance en phase de généralisation. L'objectif est d'obtenir une estimation la plus fidèle possible du comportement du système dans des conditions réelles d'utilisation. Pour cela des critères classiques comme les taux de reconnaissance et taux d'erreurs sont presque systématiquement utilisés [23]. Mais d'autres critères comme la fiabilité ou la complexité apportent aussi des informations utiles.

5.5.2 Taux de reconnaissance et taux d'erreurs

Les taux de reconnaissance et d'erreurs permettent d'évaluer la qualité du classifieur f_M par rapport au problème pour lequel il a été conçu. Ces taux sont valus grâce une base de test qui contient des formes d'écritures dans le même espace de représentation E que celles utilisent pour l'apprentissage. Elles sont aussi étiquetées par leur classe réelle d'appartenance afin de pouvoir vérifier les réponses du classifieur. Pour que l'estimation du taux de reconnaissance soit la plus fiable possible il est

primordial que le reconnaisseur n'ait jamais utiliser les échantillons de cette base pour faire son apprentissage (la base de test ne doit avoir aucun individu en commun avec la base d'apprentissage et les éventuelles bases de validation). De plus cette base de test doit être suffisamment représentative du problème de classification.

En général quand les échantillons étiquetés à disposition sont suffisamment nombreux ils sont séparés en deux parties disjointes et en respectant les proportions par classes de la base initiale. Une partie sert pour former la base d'apprentissage et l'autre pour former la base de test. Le découpage le plus courant est de 2/3 pour l'apprentissage et le 1/3 restant pour la base de test. Les performances en terme de taux de reconnaissance sont alors déterminées en présentant au classifieur chacun des exemples e_j de la base de test et en comparant la classe donnée en résultat $f_M(e_j) = s$ à la vraie classe de e_j . En considérant que la base de test contient N individus et que sur ceux-ci $N_{corrects}$ sont bien classés par le système le taux de reconnaissance τ_{reco} est simplement défini par :

$$\tau_{reco} = \frac{N_{corrects} \times 100}{N} \quad \text{II.18}$$

Le taux d'erreur τ_{err} est défini à partir du nombre d'individus N_{err} mal classés.

$$\tau_{err} = \frac{N_{err} \times 100}{N} \quad \text{II.19}$$

La répartition des exemples bien et mal classés est illustré sur le *Tableau II.1*.

Base de test : N exemples	
$N_{corrects}$	N_{err}

Tableau II.1 : Répartition des exemples dans la base de test cas général

Conclusion

Dans ce chapitre nous avons présenté les différentes étapes nécessaires à la construction d'un système de reconnaissance hors-ligne de l'écriture manuscrite. Pour chacune de ces étapes à savoir : les pré-traitements, segmentation, l'extraction des primitives et la classification.

Notre effort s'est principalement dirigé vers l'extraction des meilleures primitives pertinentes et les différents modèles des classifieurs utilisant les réseaux de neurones.

CHAPITRE II LA RECONNAISSANCE DE L'ECRITURE

1. Introduction.....	13
2. Différents aspects de reconnaissance de l'écriture	13
2.1 Reconnaissance en-ligne ou hors-ligne.....	14
3. Méthodes de reconnaissance.....	15
3.1 Reconnaissance globale ou analytique	15
4. Stratégies de reconnaissance	16
5. Organisation générale d'un système de reconnaissance.....	16
5.1 Prétraitements	17
5.1.1 Binarisation :.....	18
5.1.2 Lissage :.....	19
5.1.3 La correction de l'inclinaison :.....	20
5.1.4 Normalisation de la taille :	20
5.1.5 Extraction des contours	21
5.1.6 Extraction des composantes connexes :.....	22
5.2 Segmentation	25
5.2.1 Extraction des mots :.....	25
5.2.2 Extraction des caractères :.....	27
5.2.3 Les alignements dans le mot.....	28
5.2.3.1 Détermination de la zone médiane du mot	28
5.3 Extraction des primitives (caractéristiques).....	30
5.3.1 Les caractéristiques adaptées à l'écriture	32
5.3.1.1 Détection des caractéristiques structurelles.....	32
5.3.1.2 Détection des caractéristiques statistiques	33
5.3.1.3 Détection des caractéristiques géométriques.....	34
5.4 Classification.....	42
5.4.1 Les approches de classification	43
5.4.1.1 Les approches statistiques.....	43

5.4.1.1 Les approches structurelles	45
5.4.1.2 Approche hybride	45
5.5 La décision	45
5.5.1 Evaluation des performances d'un classifieur	46
5.5.2 Taux de reconnaissance et taux d'erreurs	46
Conclusion	47

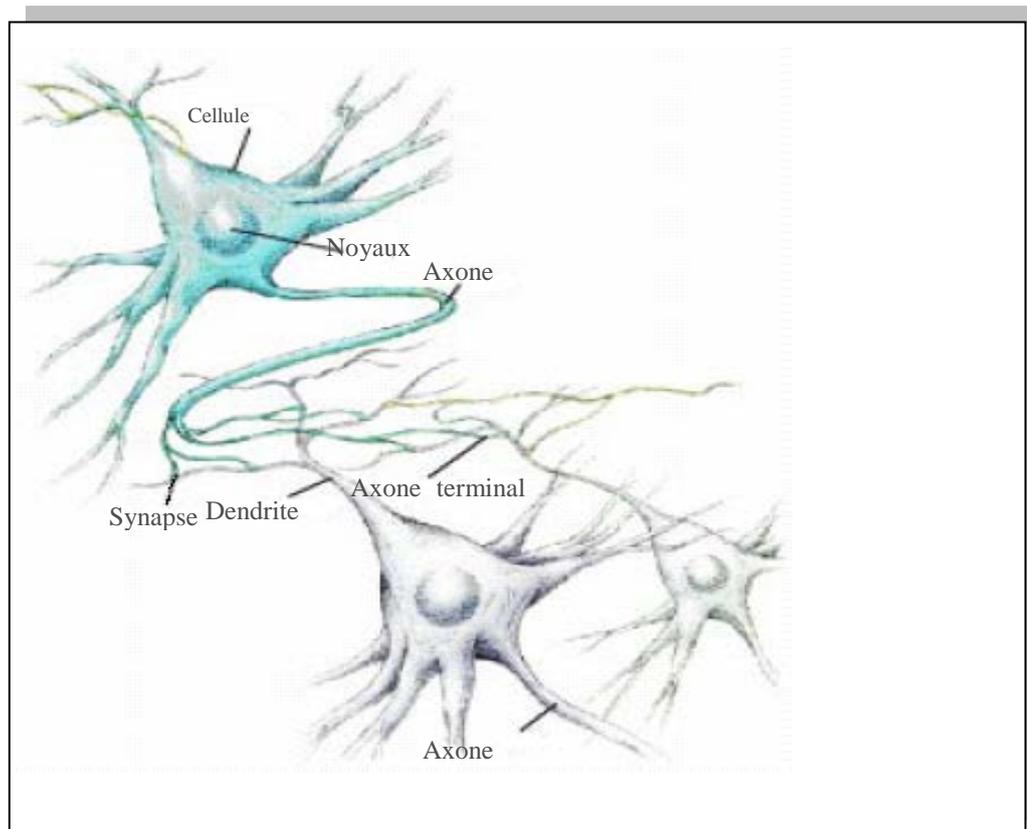
CHAPITRE III

RESEAUX DE NEURONES

RESEAUX DE NEURONES

1. Généralités

Nous abordons les réseaux de neurones dans la seule perspective de faire de la reconnaissance des formes et de la classification. Notre objectif n'est donc pas celui des « neurosciences » consistant à chercher à utiliser la puissance des ordinateurs pour simuler l'intelligence humaine ou animale. Rappelons quelques généralités sur ces cellules particulières que sont les neurones biologiques. Il s'agit donc de cellules comportant un noyau comme toutes les cellules vivantes mais comportant des branches appelées axons. Ces axons peuvent être reliés pour « connecter » des neurones.



**Figure III.1 : Vue d'artiste des connexions neuronales dans le cerveau.
Cellule émettrice en haut et cellules réceptrices en bas.**

Ces connexions se comptant par milliards et produisant ce qu'on appelle un réseau de neurones. Chacun de ces neurones peut être activé ou non par ce qu'on appelle la « force synaptique » et transmettre son « activation » par l'intermédiaire des

synapses aux autres neurones. Chaque neurone pouvant être considéré comme comportant des entrées et des sorties. Ce réseau de neurones peut recevoir des activations par l'intermédiaire des sens notamment et l'état d'excitation des neurones définit alors un état de la perception.

2. Modélisation du neurone

2.1 Réseaux de neurones formels :

Les réseaux de neurones formels sont à l'origine une tentative de modélisation mathématique du cerveau humain. Les premiers travaux datent de 1943 et sont l'œuvre de McCulloch et Pitts. Ils présentent un modèle assez simple pour les neurones et explorent les possibilités de ce modèle. L'idée principale des réseaux de neurones "modernes" est la suivante : On se donne une unité simple, un neurone, qui est capable de réaliser quelques calculs élémentaires. On relie ensuite entre elles un nombre important de ces unités et on essaye de déterminer la puissance de calcul du réseau ainsi obtenu. Il est important de noter que ces neurones manipulent des données numériques et non pas symboliques.

Citons quelques étapes dans la formalisation des réseaux de neurones :

- Première définition d'un neurone formel par McCulloch et Pitts en 1943
- Les percepts ou concepts sont physiquement représentés dans le cerveau par l'entrée en activité (simultanée) d'une assemblée de neurones (Hebb, 1949).
- Deux neurones entrant en activité simultanément vont être associés (c'est-à-dire que leurs contacts synaptiques vont être renforcés. On parle de loi de Hebb et d'associationnisme
- Le perceptron de Frank Rosenblatt (1958) : le premier modèle pour lequel un processus d'apprentissage a pu être défini. De cette période, date également les travaux de Widrow et Hoff.
- Le livre de Minski et Papert "Perceptrons" (1969). Cet ouvrage contient une étude critique très complète des perceptrons. On lui reproche parfois violemment d'avoir sonné le glas des recherches sur les réseaux neuronaux dans les années 70, ce que nient leurs auteurs. Ce livre a été réédité en 1980, avec des ajouts et corrections manuscrites dans les marges, sans doute pour qu'on ne puisse pas les accuser de camoufler la première version du texte.
- L'algorithme de rétro propagation du gradient dans les réseaux multicouches découvert au début des années 80 par Rumelhart et McClelland, Parker, Hinton,

Le Cun. Les « inventeurs » sont nombreux car l'idée de descente de gradient est naturelle. La plupart de ces travaux étaient associés à des études empiriques montrant la puissance du modèle.

- Le modèle de Hopfield (1982) qui utilise des réseaux totalement connectés basés sur la règle de Hebb qui ont permis de définir la notion d'attracteurs et de mémoire associative.

2.1.1 Model simplifié (premier modèle de neurone):

Le modèle de neurone formel présenté ici, dû à McCulloch et Pitts, *Figure III.2* est un modèle mathématique très simple dérivé d'une analyse (elle aussi assez simple) de la réalité biologique. On constate tout d'abord que le modèle biologique fait intervenir une notion temporelle qui est difficile à intégrer dans un modèle simple. On oublie donc cette notion et de ce fait on remplace l'intégration temporelle par une simple sommation des signaux arrivant au neurone (ces signaux sont communément appelés les entrées du neurone). On compare ensuite la somme obtenue à un seuil et on déduit de la comparaison la sortie du neurone. Cette sortie sera par exemple *Figure III.3* égale à 1 si la somme est supérieure au seuil et à 0 dans le cas contraire. Plus formellement encore, il suffit pour obtenir ce comportement de soustraire le seuil considéré à la somme des entrées, et de faire passer le résultat par la fonction de transfert du neurone qui est ici la fonction de Heaviside *figure III.4*. Le résultat après transfert est alors la sortie du neurone.

Cet enchaînement "sommation" puis "non-linéarité" représente finalement les propriétés "physiques" du neurone.

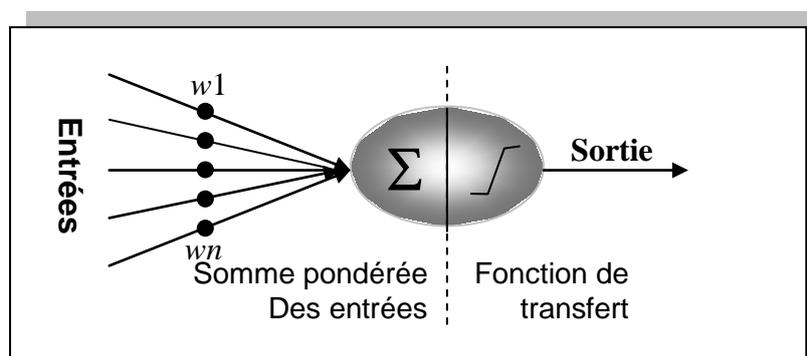


Figure III.2 : Le neurone formel de McCulloch et Pitts

La modélisation des synapses est assez simpliste en première approche. On se contente en effet d'introduire la notion de synapse excitatrice et de synapse

inhibitrice en multipliant la sortie d'un neurone par 1 ou -1 avant de la transmettre aux autres neurones. Afin de donner plus de puissance au modèle, on généralise ce modèle en introduisant ensuite une connexion synaptique qui est une valeur réelle. La sortie d'un neurone est alors multipliée par la valeur de la connexion synaptique avant d'être transmise à un autre neurone. On modélise ainsi la connexion entre deux neurones.

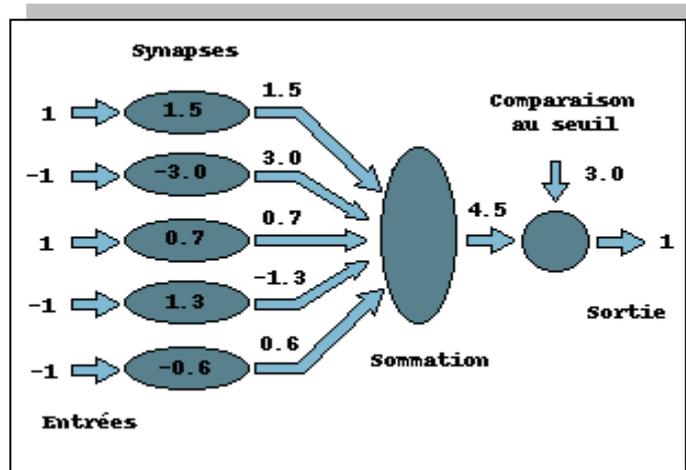


Figure III.3 : Exemple d'un neurone formel.

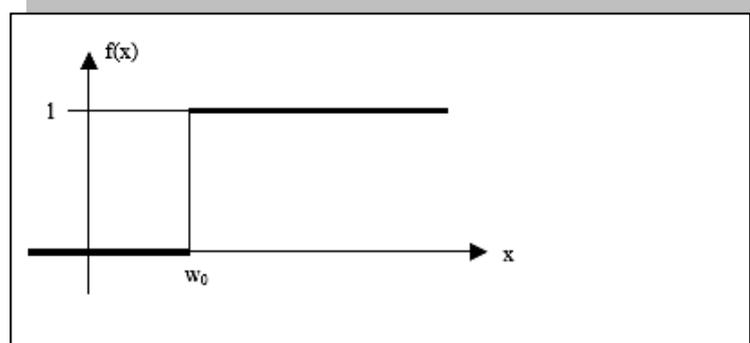


Figure III.4 : Fonction de Heaviside, utilisée par MacCulloch et Pitts.

En résumé, un neurone formel réalise simplement une somme pondérée de ses entrées, ajoute un seuil à cette somme et fait passer le résultat par une fonction de transfert pour obtenir sa sortie.

Le fonctionnement du neurone formel est donné par l'équation suivante :

$$S = F\left(\sum_{i=1}^n w_i \times .e_i\right) \quad \text{III.1}$$

Tel que :

$.e_i$: $i=1 \dots n$: les entrées du neurone formel.

S : sa sortie.

w_i : les paramètres de pondération.

2.1.2 Modélisation générale

D'une façon plus générale, on peut définir un neurone formel par les quatre éléments suivants :

- La nature des entrées.
- La fonction de sortie qui calcule la sortie du neurone.
- La nature de la sortie du neurone.

Nous adapterons par la suite les notations suivantes :

- $(e_i)=1 \dots n$: seront les entrées.
- h sera la fonction d'entrée totale.
- F sera la fonction de sortie.
- D'autre part, nous utiliserons aussi

$E=h(e_1 \dots e_n)$ comme entrée totale.

$S=F(E)$ comme sortie.

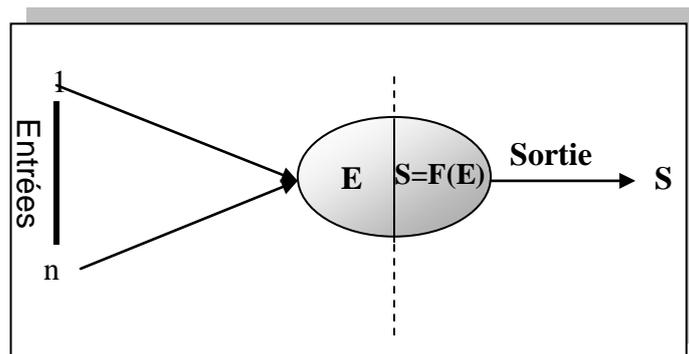


Figure III.5 : Modélisation d'un neurone.

La nature des entrées et de la sortie peuvent être :

- Binaire : $(-1, +1)$ ou $(0, 1)$.
- Réelles.

La fonction d'entrée totale h peut être :

- Booléenne
- Linéaire : $h(e_1 \dots e_n) = \sum w_i \cdot e_i$

La fonction de sortie 'F' peut être :

- Une fonction binaire à seuil ; dans ce cas, on s'arrange pour que la forme de h soit telle qu'on puisse utiliser la fonction de Heaviside ou la fonction Signe.

- Une fonction as seuil où multi-seuils
- Une fonction non-linéaire (la fonction sigmoïde)
- Une fonction stochastique.

2.2 Différentes structures des réseaux de neurones

Les structures qui peuvent être utilisées sont très variées. Si l'on se réfère aux études biologiques du cerveau, on constate que le nombre de connexions est énorme. Par exemple, des chercheurs ont montré que le cortex était divisé en différentes couches, à l'intérieur d'une même couche les interactions entre neurones sont très grandes mais les neurones d'une couche sont reliés aux neurones des autres couches, le tout formant un système gigantesque.

D'une manière générale, l'architecture des réseaux de neurones formels peut aller d'une connectivité totale à une connectivité locale où les neurones ne sont reliés qu'à leurs plus proches voisins. Il est courant d'utiliser des réseaux à structure régulière pour faciliter leur utilisation.

Les réseaux de neurones peuvent se répartir en deux grandes catégories, selon qu'ils interdisent ou qu'ils permettent les liens récurrents entre leurs neurones. Les premiers dites *réseaux à couches*, voient l'activation de leurs neurones les traverser en une vague unique n'ont donc pas véritablement de comportement dans le temps.

En revanche, les réseaux appartiennent à la seconde catégorie, nommés *réseaux entièrement connectés*, sont caractérisés par des comportements dynamiques qui peuvent être plus ou moins complexes selon les contraintes d'architecture qu'ils comportent.

2.2.1 Les Réseaux à couches

Il est possible, à l'étude du système visuel, de concevoir une organisation par couches successives du système nerveux, les signaux arrivant sur les cellules sensibles, sont transmis à un système de neurone hyper connectés, avec entre autres des mécanismes de bouclage puis aboutissent aux cellules motrices qui déclenchent un comportement.

Ce système portant simplifié est déjà très complexe à modéliser si l'on tient à conserver la structure de connectivité quasi-totale dans les couches intermédiaires des traitements du signal. C'est pourquoi on utilise plutôt une

structure de réseaux à couches telle que les neurones appartenant à une même couches ne soient pas connectés entre eux, *figure III.6*, chacune des couches recevant des signaux de la couche précédente, transmet le résultat des ces traitements à la couche suivante.

Les deux couches extrêmes correspondent à la couche qui reçoit ses entrées du milieu extérieur à la couche qui fournit le résultat des traitements effectués. Les couches intermédiaires sont appelées couches cachées, leur nombre est variable.

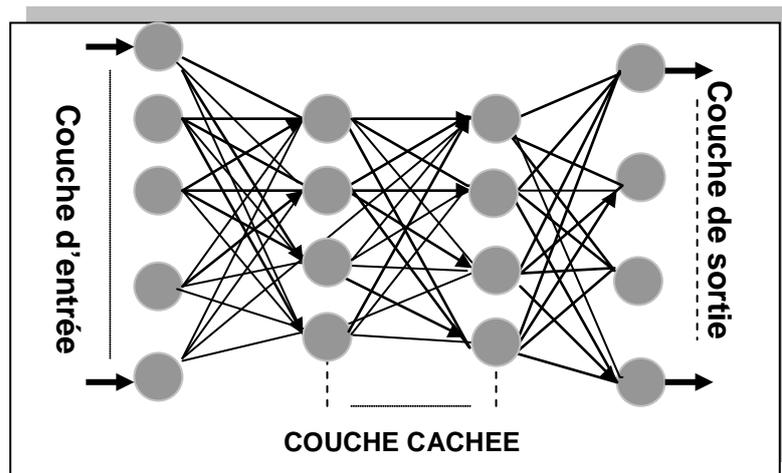


Figure III.6 : Réseaux multicouches.

2.2.2 Les Réseaux entièrement connectés

Dans ces réseaux, chaque cellule est reliée à toute les autres et possède un retour sur elle-même, il n'y a en général qu'une seule couche vue et au mieux deux; une d'entrée et une de sortie. Un neurone reçoit des impulsions de tous ses compagnons et même de lui-même.

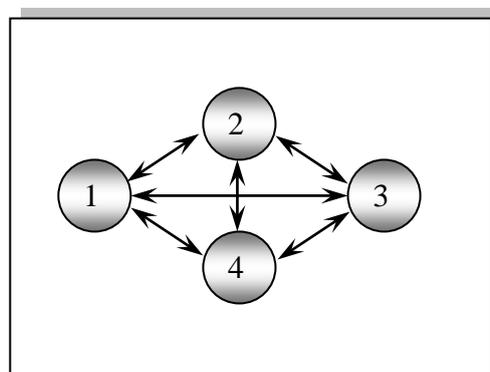


Figure III.7. Réseaux entièrement connectés

2.3 Les réseaux les plus célèbres

Il y a de très nombreuses sortes de réseaux de neurones actuellement. Personne ne sait exactement combien. De nouveaux réseaux (ou du moins des variations de réseaux plus anciens) sont inventés chaque semaine. On en présente ici de très classiques.

2.3.1 Le perceptron :

C'est un des premiers réseaux de neurones, conçu en 1958 par **Roseblatt**. Il est linéaire et monocouche. Il est inspiré du système visuel. Essentiellement, le perceptron est composé de deux couches *figure III.8*. La première (d'entrée) appelée à l'origine la rétine du perceptron, quant à la deuxième couche, elle donne la réponse du perceptron correspondant à la stimulation présentée en entrée.

Les sorties des neurones ne peuvent prendre que deux états (-1 et 1 ou 0 et 1) seuls les poids des liaisons entre la couche d'association et la couche finale peuvent être modifiés.

2.3.2 Le modèle ADALINE

L'ADALINE (Adaptatif Linéaire Élément) conçu par B.Widrow dans les années 1960, est un perceptron sans couche cachée donc à un seul neurone *figure III.9*, qui reçoit le stimulus arrivant de la couche d'entrée et donne la réponse correspondante.

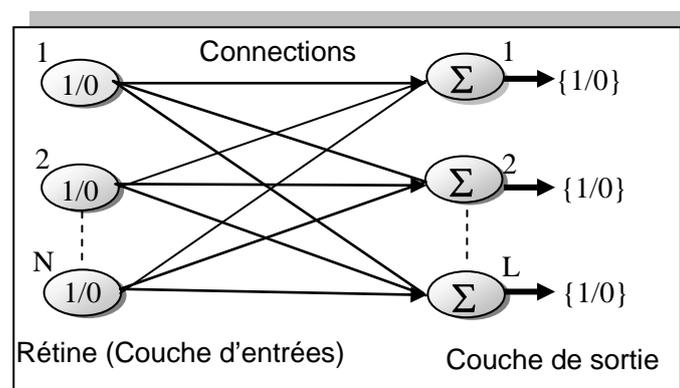


Figure III.8 : Structure d'un perceptron.

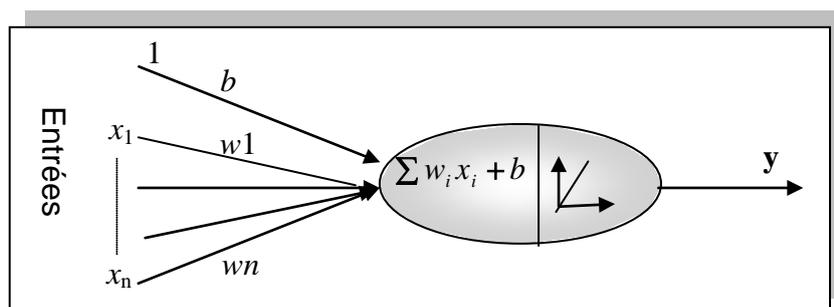


Figure III.9 : Architecture de L'ADALINE

2.3.3 Le perceptron multicouches (PMC)

C'est une amélioration du perceptron comprenant une ou plusieurs couches intermédiaires dites couche cachée figure III.6. Ils utilisent, pour modifier leur poids, un algorithme de rétropropagation du gradient, qui est une généralisation de la règle de WIDROW-HOFF, les PMC agissent comme un séparateur non linéaire et peuvent être utilisés pour la classification, le traitement d'image, l'aide de la décision ou la commande d'un processus

Vu l'importance de ce modèle dans notre étude, il fera l'objet d'une étude détaillée dans ce qui suit.

2.4 L'apprentissage

Le principal problème pour les réseaux de neurones est d'arriver à trouver un ensemble de valeurs des synapses (poids), qui sont les porteurs de l'information, tel que les configurations d'entrée se traduisent par les réponses voulues, et cela en partant d'une valeur particulière des poids des connections, le réseau améliore ses réponses en ajustant ses coefficients selon un algorithme ou une règle d'apprentissage

Il existe trois classes d'apprentissage : l'apprentissage supervisé, semi supervisé et non-supervisé.

2.4.1 Apprentissage supervisé

Les apprentissages supervisés demandent que l'on donne au réseau neuronal des exemples résolus, c'est à dire des couples de vecteurs entrée/sortie. Cet apprentissage exploite le plus souvent plusieurs idées simples. L'idée principale est la minimisation itérative d'un critère de l'erreur en sortie du réseau. On initialise les

matrices de connexion au hasard, puis l'on fait évoluer ces matrices de manière à ce qu'elles autorisent l'association souhaitée, c'est à dire jusqu'à ce qu'un critère de l'erreur (entre les sorties réellement obtenues et les sorties souhaitées) soit quasi nul.

2.4.2 Apprentissage semi-supervisé

Dans ce type d'apprentissage l'utilisateur possède seulement des indications imprécises sur le comportement final du réseau, mais en revanche, il est possible d'obtenir des indications qualitatives (correctes/ incorrectes).

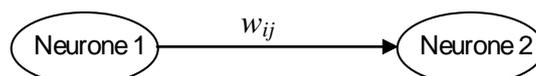
2.4.3 Apprentissage non-supervisé (auto-organisation)

Dans ce type d'apprentissage (sans professeur) les poids synaptiques du réseau sont modifiés selon des critères internes comme la co-activation des neurones. Le comportement de ce type d'apprentissage est comparable à des techniques d'analyse de données. Un exemple des réseaux à apprentissage non-supervisé : les cartes topologiques de Kohonen. Enfin, certains réseaux associent les deux types d'apprentissage supervisé et non-supervisé, c'est le cas par exemple du réseau de Boltzmann.

2.5 Règles d'apprentissages

2.5.1 La règle de Hebb :

Dans le domaine de la recherche sur le fonctionnement des neurones biologiques et sur les mécanismes d'apprentissage de l'intelligence humaine, Hebb a proposé un type de réseau de neurones totalement interconnecté (c'est à dire où les neurones sont reliés par des connexions de type synapse fonctionnant à la fois en « entrée » et en « sortie ». Ces connexions sont affectées de poids qui évoluent au cours du temps et en fonction de l'activation de chacun des deux neurones extrémités de cette connexion. Il a ainsi défini une règle d'apprentissage dite de Hebb :



Cette règle considère alors que toute connexion entre deux neurones se renforce si ces deux neurones sont actifs au même moment. Si on note A_1 et A_2 l'activation des neurones 1 et 2 et si on suppose qu'un neurone actif à son activation qui vaut 1

et qu'un neurone inactif a son activation qui vaut 0 alors l'expression de la règle de Hebb est la suivante :

$$w_{ij}(t + \delta t) = w_{ij}(t) + \mu \times A_i \times A_j \text{ avec } \mu \neq 0 \quad \text{III.2}$$

Au départ on a $w_{ij}(0) = 0$ pour tout i, j .

Ce type de réseaux a surtout une vocation pour la modélisation des neurones biologiques et reste peu utilisé en reconnaissance des formes. Un des inconvénients majeurs de ce modèle vient du fait que les w_{ij} ne peuvent qu'augmenter au cours du temps

2.5.2 La règle de Widrow-Hoff

La règle de Widrow-Hoff ou règle delta proposée en 1960, consiste à modifier chaque pas, les poids et les biais afin de minimiser la somme des carrés des erreurs en sortie en utilisant la règle suivante :

$$w(k+1) = w(k) + \eta(t_k - y_k)x_k^T \quad \text{III.3}$$

A chaque pas d'apprentissage k , l'erreur en sortie est calculée comme la différence entre la cible recherchée t et la sortie y du réseaux

$$E_k = e_k^T e_k = (t_k - y_k)^T (t - y) = \frac{1}{2}(t_k^T t_k + y_k^T y_k - 2y_k^T t_k) \quad \text{III.4}$$

Le gradient de se calcule comme suite :

$$\nabla E_{kw} = \frac{1}{2} \nabla [y_k^T y_k - 2y_k^T t_k]_w \quad \text{III.5}$$

le gradient se calcule comme suite :

$$\nabla E_{kw} = \frac{\partial E_k}{\partial w} = \frac{\partial E_k}{\partial y_k} \frac{\partial y_k}{\partial w} \quad \text{III.6}$$

D'après l'expression de E_k et avec $y_k = wx_k + b$ les dérivées partielles sont

$$\frac{\partial E_k}{\partial w} = (y_k - t_k)$$

$$\frac{\partial (wx_k + b)}{\partial w} = x_k$$

La mise à jour des poids se fait par l'équation :

$$w(k+1) = w(k) + \eta \nabla E_{kw} \quad \text{III.7}$$

Avec η : le gain d'apprentissage, $0 < \eta < 1$

De même on obtient l'expression de la modification du biais :

$$b(k+1) = b(k) + \eta(t_k - y_k) = b(k) + \eta \nabla E_{kb} \quad \text{III.8}$$

2.5.3 Apprentissage du perceptron multicouches

L'apprentissage du perceptron multicouches est supervisé et consiste à adapter les poids des neurones de manière à ce que le réseau soit capable de réaliser une transformation données, représentée par un ensemble d'exemples constitué d'une suite de N vecteurs d'entrées.

Donc l'apprentissage fonctionne sur le même principe que la règle de Widrow-hoff ; on dispose d'un ensemble d'exemples qui sont des couples (entrées-sorties désirées). A chaque étape un exemple est présenté en entrée du réseau. Une sortie réelle est calculée. Ce calcul est effectué de proche en proche de la couche d'entrée à la couche de sortie. Cette phase est appelée propagation avant ou relaxation du réseau. Ensuite l'erreur (somme quadratique des erreurs sur chaque sortie) est calculée. Cette erreur est ensuite rétro propagée dans le réseau, donnant lieu à une modification de chaque poids.

L'apprentissage du perceptron multi-couches consiste à minimiser l'erreur quadratique commise sur l'ensemble des exemples, Ce problème de minimisation de l'erreur a été résolu par des méthodes de *rétropropagation du gradient d'erreur*.

2.6 Le Perceptron multicouches comme un classifieur

L'introduction de couches intermédiaires dans le réseau MLP permet de résoudre des problèmes plus complexes que la simple séparation linéaire. Lorsqu'il existe au moins une couche cachée, les états internes du réseau ne peuvent plus être donnés directement par les exemples et les sorties désirées puisque les sorties des neurones appartenant aux couches intermédiaires sont inconnues. La *figure III.10* représente un réseau de neurones multicouches avec comme entrée le vecteur de primitives et en sortie les classes où seront classées les formes. Pour déterminer le nombre de couches cachées dans un réseau cela dépendra du problème à résoudre (comment les classes des formes sont-elles séparées?).

Plusieurs types de réseaux de neurones multicouches ont été développés. Nous focalisons notre étude sur le type MLP. Dans le processus de la construction de ce type de classifieur, l'apprentissage a comme objectif d'adapter les poids des connexions entre les neurones de sorte que le réseau donne en sortie la classe

d'appartenance des formes qui lui sont proposées en entrée. Ce qui revient à minimiser l'erreur commise par le réseau sur l'ensemble de formes de la base d'apprentissage. Pour une forme de la classe C_i , l'état du neurone O_i de la couche de sortie du MLP doit être proche de la valeur maximale et tous les autres proches de la valeur minimale. Ce problème de minimisation de l'erreur a été résolu par des méthodes de *rétropropagation du gradient d'erreur*. Toutefois, ces méthodes ne garantissent pas l'obtention du minimum global de la fonction à optimiser (blocage dans un minimum local). Le terme «*rétropropagation*» est utilisé pour décrire l'apprentissage du réseau de neurones de type MLP utilisant la descente du gradient appliquée à la fonction de la somme des erreurs quadratiques. Le réseau de type MLP à une couche cachée peut estimer n'importe quelle fonction dans R^n avec une précision arbitraire. Ainsi, le MLP est capable d'estimer des hyper-surfaces discriminantes très complexes. Les fonctions estimées sont soit linéaires ou non linéaires. Elles sont appelées fonctions d'activation. La fonction non linéaire que nous utiliserons est la fonction *sigmoïde*.

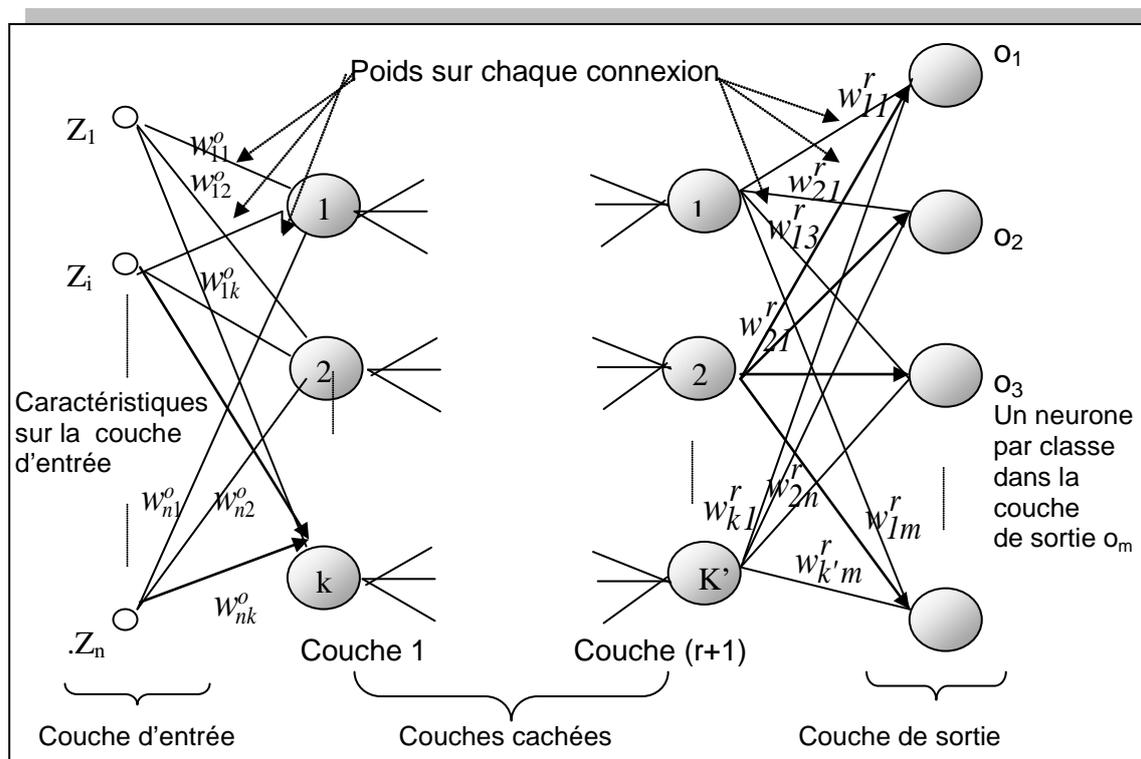


Figure III.10 : Réseau de neurones multicouches.

Par contre la difficulté majeure rencontrée lors de l'utilisation de ce type de classifieur consiste à déterminer le nombre de couches cachées, le nombre de neurones dans

chacune des couches et les poids des connexions entre les différentes couches (taux d'apprentissage...). De ce fait, pour une application donnée, la construction du classifieur de type MLP utilise des règles empiriques et nécessite un certain nombre d'essais afin d'obtenir des performances de généralisation intéressantes.

Pour déterminer les poids de toutes les connexions du réseau, l'utilisation des algorithmes d'apprentissage est obligatoire. L'objectif des algorithmes d'apprentissage est de minimiser l'erreur de décision effectuée par le RNA en ajustant les poids à chaque présentation d'un vecteur d'entraînement. Pour ce qui est de l'ajustement des poids à une étape donnée de la phase d'apprentissage, l'erreur à minimiser est habituellement celle produite lors de l'application d'un vecteur de l'ensemble d'apprentissage à l'entrée du réseau. Pour ce qui est de l'évaluation de la qualité d'apprentissage du réseau, l'erreur cumulée par tous les vecteurs de l'ensemble d'entraînement est évaluée. Cette erreur cumulée est calculée pour tous les cycles de la phase d'entraînement et est définie à partir de l'erreur quadratique. Cette mesure de l'erreur illustre la précision obtenue après P cycles d'apprentissage. En résumé, nous utiliserons pour l'apprentissage du réseau l'algorithme de rétro-propagation avec minimisation du gradient d'erreur qui est défini par les étapes suivantes :

1. Initialiser les poids à des petites valeurs et les seuils du réseau.
2. Insérer à l'entrée du réseau une observation (exemple) de la base de données en forme de vecteur de caractéristiques, puis calculer sa valeur d'activation et sa fonction d'activation en utilisant les formules (III.9) et (III.10).
3. Évaluer le signal d'erreur des sorties du réseau en utilisant la formule (III.11).
4. Ajuster les poids en utilisant la formule (III.9).
5. Évaluer le signal d'erreur pour chaque couche cachée en utilisant la formule (III.12).
6. Ajuster les poids de la couche cachée en utilisant la formule (III.10).
7. Répéter les étapes 2 à 6 pour l'ensemble des observations de la base d'apprentissage tant que le critère d'arrêt n'a pas été atteint.

Il existe plusieurs critères d'arrêts. Ces critères peuvent être combinés entre eux.

Le premier critère est basé sur l'amplitude du gradient de la fonction d'activation, puisque par définition le gradient sera à zéro au minimum. L'apprentissage du réseau du type MLP utilise la technique de recherche du gradient pour déterminer les poids du réseau.

Le second critère d'arrêt est de fixer un seuil que l'erreur quadratique ne doit pas dépasser. Toutefois ceci exige une connaissance préalable de la valeur minimale de l'erreur qui n'est pas toujours disponible. Dans le domaine de la reconnaissance de formes, il suffit de s'arrêter lorsque tous les objets sont correctement classés

Les deux premiers critères sont sensibles aux choix des paramètres (par exemple : le nombre de noeuds dans la couche cachée, le seuil d'erreur, ...) et si le choix n'est pas bon alors les résultats obtenus seront mauvais ou le temps de calcul de la performance du système de reconnaissance sera plus lent (par exemple définir un grand nombre de noeuds dans la couche cachée). Cependant, la méthode du crossvalidation n'a pas ce genre de problème.

En général, les formules utilisées par cet algorithme sont :

- pour l'ajustement de poids entre le noeud j (sortie ou cachée) et le noeud i (cachée ou entrée)

$$\Delta w_{ij} = \eta \delta_j o_i \quad \text{III.9}$$

Où η est la valeur de la constante d'apprentissage.

En général, $0.1 < \eta < 0.9$

et o_i est la valeur d'activation du neurone i tel que $o_i = f(\text{net}_i)$

$$\text{net}_i = \sum_j w_{ij} o_j \quad \text{III.10}$$

Avec $f(\text{net}_i)$ est la fonction d'activation. La fonction utilisée dans le cas de notre projet est la fonction sigmoïde définie

$$f(\text{net}_i) = \frac{1}{1 + \exp^{-\text{net}_i}} \quad \text{III.11}$$

- pour le calcul du signal d'erreur du neurone j de la couche de sortie avec d_j est la valeur désirée du neurone j .

$$\delta_j = (d_j - o_j) o_j (1 - o_j) \quad \text{III.12}$$

Avec d_j est la valeur désirée du neurone j .

- pour le calcul du signal d'erreur du neurone j de la couche cachée avec k nœuds

$$\delta_j = o_j (1 - o_j) \sum_k w_{jk} \delta_k \quad \text{III.13}$$

Ces formules ont été dérivées de la formule de calcul de l'erreur quadratique de l'ensemble de la base d'apprentissage définie comme suit

$$E_p = \frac{1}{2} \sum_p \left(\sum_k (d_{pk} - o_{pk})^2 \right) \quad \text{III.14}$$

Où p est l'indice d'un exemple de la base et k est l'indice du noeud de sortie.

L'objectif est de minimiser cette erreur.

2.7 Evolution du perceptron multicouches

Les réseaux de neurones multicouches entraînés par l'algorithme de rétro propagation du gradient sont les modèles connexionnistes les plus étudiés et utilisés à ce jour. Les champs d'application de ces réseaux sont très vastes : classification, identification de processus et prédiction de séries temporelles, commande de processus et de robot, traitement d'images, et de paroles. Ce sont des modèles robustes et dont les entrées et les sorties peuvent indépendamment être choisies binaires ou réelles, ce qui permet de traiter de très vastes classes de problèmes. Cependant, lors de la réalisation d'une application basée sur un réseau de neurones multicouches entraîné par l'algorithme de rétro propagation, il faut prendre en compte les points suivants :

- La mise en œuvre de cet algorithme exige souvent des temps de calcul très longs qui rendent son application inconmode pour de nombreux problèmes de taille raisonnable.
- La rétro-propagation peut trouver piège dans les minima locaux, et ne peut pas donner la bonne réponse. Pour remédier à ce problème, on peut entraîner le réseau à partir de plusieurs choix initiaux de poids pour ne garder que le meilleur ou bien ajouter un bruit aléatoire puis relancer l'apprentissage.
- Le réseau apprend à partir des exemples de la base d'apprentissage. Si cette dernière concerne un fonctionnement dans un domaine réduit, le réseau ne saura répondre en dehors de ce domaine. Si le superviseur donne des informations erronées, le réseau apprendra un modèle erroné et si le processus que l'on cherche à modéliser est non stationnaire ou change brutalement, le réseau réapprendra un bon modèle.

Aujourd'hui, les perceptrons multicouches sont les réseaux utilisés par les développeurs d'applications. Ces résultats théoriques sur le mécanisme de comportement de ces réseaux sont encore très pauvres, mais des résultats

satisfaisants ont été mis en valeur dans des domaines d'applications très divers. C'est la raison pour laquelle de nombreux chercheurs étudient ce modèle et tentent de mieux le comprendre et de l'améliorer.

Conclusion

Nous avons essayé à travers ce chapitre de donner aperçu général sur les réseaux de neurones artificiels et les différentes règles d'apprentissage.

Les réseaux de neurones les plus connus présentés dans ce chapitre sont des systèmes paramétriques non linéaires qui fera l'objet des chapitres suivants avec son application au problème de reconnaissance. Les paramètres du réseau sont déterminés statistiquement lors d'une phase d'apprentissage à partir d'une base de données d'exemples, appelée base d'apprentissage.

En pratique, l'utilisation de la méthode neuronale pose certaines difficultés. La principale difficulté est l'optimisation de la phase d'apprentissage. Le choix de l'architecture adéquate ou la détermination du « pas d'apprentissage » se fait par essais successifs.

CONCLUSION GENERALE

RECONNAISSANCE DE CARACTERES LATINS

1 Introduction :

La reconnaissance de l'écriture manuscrite est un problème pour lequel un modèle de reconnaissance doit nécessairement intégrer la prise en compte d'un nombre important de variabilités. Dès lors, les méthodes basées sur les techniques d'apprentissage peuvent apporter une certaine souplesse à la construction d'un tel modèle.

Ce chapitre, sera consacré aux l'applications de réseaux de neurones à la reconnaissance de caractères latins.

2 Application à la reconnaissance des caractères latins

2.1 Base de données

Dans tout système de reconnaissance de formes, il est nécessaire d'avoir une base de données afin d'effectuer tous les traitements. La base de données que nous avons utilisée dans notre projet contient des caractères manuscrits, isolés et stockés sous forme d'images binaires. Le nombre de classes est égale à 21 (a,c,d,e ...z).*Figure IV.1*

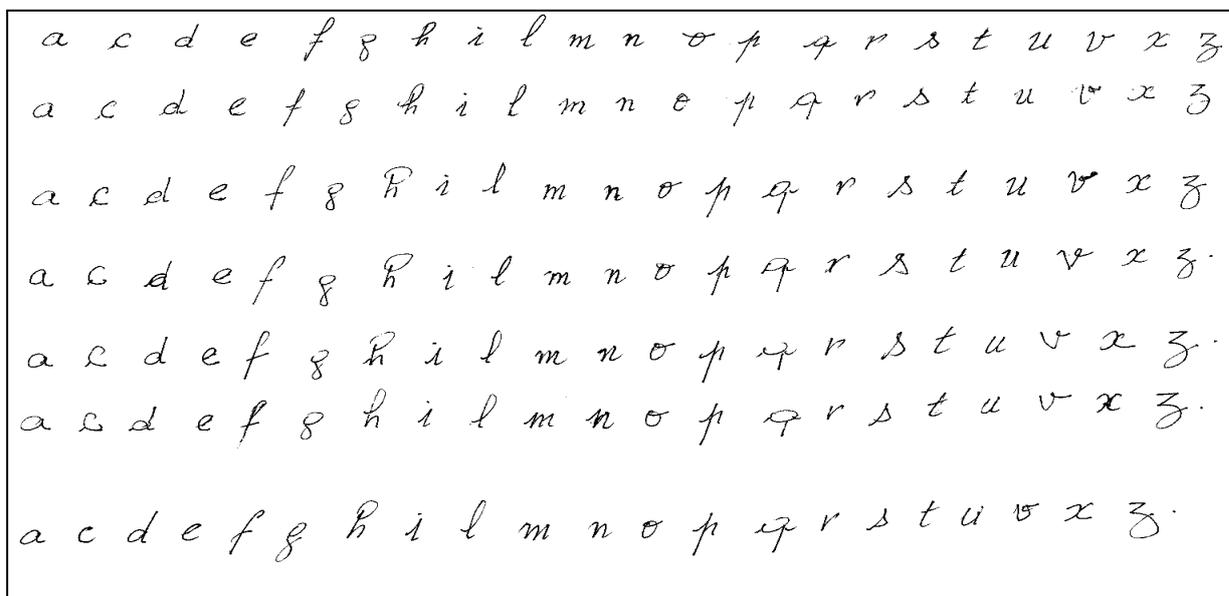


Figure IV.1 : Aperçu sur la base de donnée.

2.2 Pré-traitements

Les pré-traitements effectués sur cette base de données ont pour objectif le nettoyage des images. Autrement dit le bruit a été supprimé par les différentes techniques de pré-traitement (binarisation, lissage, etc.).

2.3 Le choix des primitives :

Nous avons effectué plusieurs simulations pour déterminer les primitives qui semblent être appropriées au système de reconnaissance de caractère isolés. Les primitives extraites sont :

- *paramètres de distribution (zonage),*
- *histogrammes de directions de contour extérieur,*
- *la densité des profils.*
- *les coefficients de Fourier afférent au contour extérieur,*
- *les sept moments invariants.*

2.3.1 Les Paramètres de Distribution (PD)

Ces paramètres caractérisent la distribution des pixels, dans des zones obtenues en divisant l'image en 9 zones de deux manières différentes. Les paramètres de distribution qui sont désignés aussi par caractéristiques zonales ou locale sont définis par :

$$R_j = \frac{N_j}{N} \quad /N$$

IV.1

N_j : nombre de pixels objets dans la région j .

N : nombre total des pixels dans le caractère.

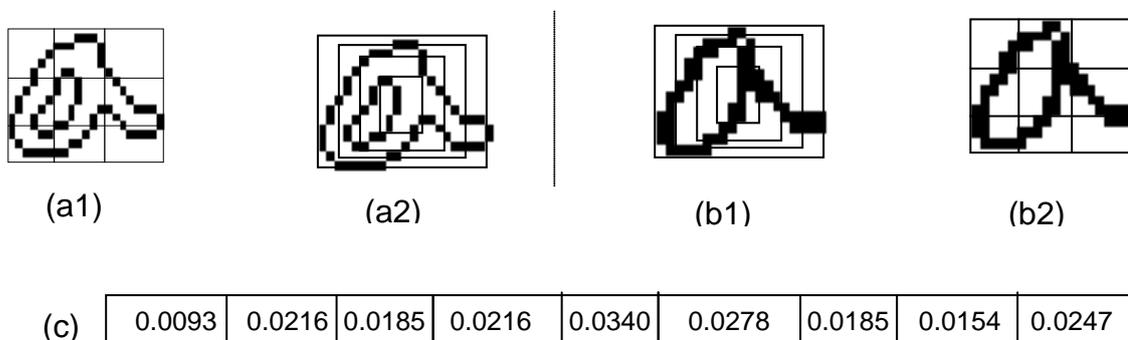


Figure IV.2: Extraction des paramètres de distribution -a1 Image de contours est divisée en 9 zones -a2 Image de contours est divisée en 4 zones -b1 Image de caractère est divisée en 4 zones -b2 Image de caractère est divisée en 4 zones -c Vecteur caractéristique associé à l'image b2.

2.3.2 Le vecteur caractéristique histogrammes de direction (HD)

Nous avons décidé de mettre en oeuvre la technique des histogrammes de directions afin de coder les contours des caractères. L'extraction de ces caractéristiques est donc réalisée sur l'image des contours. Cette dernière est disponible dans notre système au cours de la phase de pré-traitement. La première étape de l'extraction consiste à découper l'image en un certain nombre de zones (technique du "zoning") dans notre cas on a 4 zones. Un suivi de contour est alors réalisé de manière à obtenir les directions entre pixels successifs. Ces dernières sont codées par l'intermédiaire du code de Freeman. Les occurrences de ces directions sont alors dénombrées pour chaque zone individuellement. Nous présentons sur *Figure IV.3* un exemple de cette technique d'extraction. Pour ces caractéristiques, la normalisation s'effectue en divisant chaque compteur de direction d'une zone par le nombre de pixels de contours présents dans cette zone. Le vecteur caractéristique associé à cet espace de représentation contient alors huit composantes par zone.

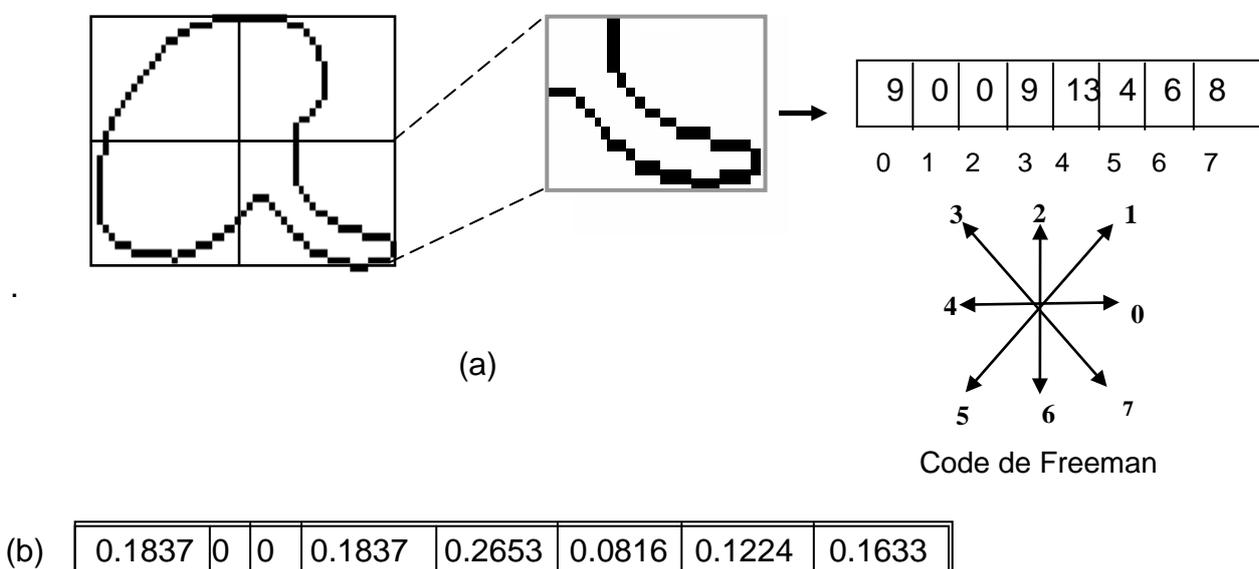


Figure IV.3 : Le vecteur caractéristique histogrammes de direction

- a Extraction des huit directions pour chaque zone
- b vecteur de direction normalisé

2.3.3 La densités des profiles (DP) :

Parmi les caractéristiques métrique mises en œuvre sont les huit profils des caractères, ces derniers décrivent bien les formes externes des caractères et laissent distinguer entre un grand nombre de lettres. Après les étapes d'extraction des différents profiles (gauche, droite, supérieur, inférieur et orienter). *Figure IV.4.* Pour chaque profil on détermine la répartition des points constituant ce profil, enfin une normalisation s'effectue en divisant chaque répartition par le nombre total de pixels noir dans l'image.

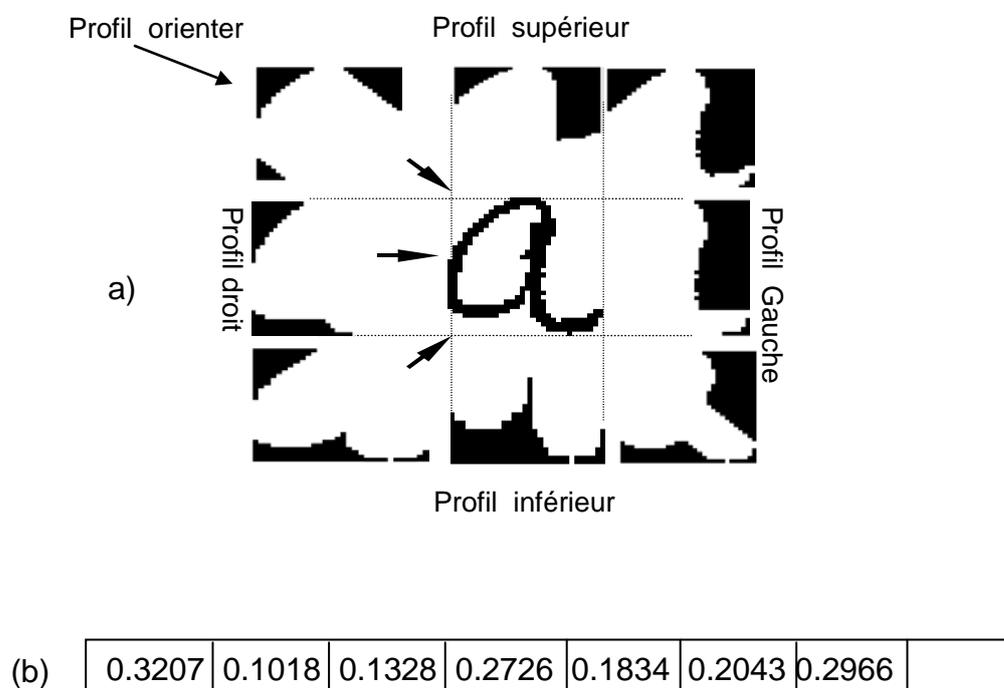


Figure IV.4: Les huit projections des Profils pour la lettre 'a'.

-a Les huit projections des profils pour la lettre 'a'.

-b Vecteur caractéristique associé aux huit profils.

2.3.4 Les sept moments invariants (MI)

Les moments géométriques permettent de trouver les caractéristiques propres aux formes à reconnaître telles que les axes principaux d'inertie, le centre de gravité, l'orientations de ses formes ainsi que d'autre. L'exploitation de ces informations pourrait permettre de distinguer les différentes formes entre elles.

L'Algorithme utilisé pour le calcul des sept moments invariants est :

POUR chaque caractère **FAIRE**

- lecture du modèle de caractère à partir du fichier
- calcul des moments m_{pq} (*équation II.5*)
- calcul des moments centrés μ_{pq} (*équation II.6*)
- calcul des moments invariants φ_i

FIN FAIRE

Stockage du résultat de φ_{ik}

Avec

$$i = 1 \rightarrow 7$$

k : représente le nombre de caractères.

2.3.5 Les coefficients de Fourier (DF)

Le choix des descripteurs de Fourier dans notre système de reconnaissance est dû à leurs propriétés d'invariabilité et de stabilité. En effet, à partir de la fonction du contour d'une image, on peut générer un ensemble complet de nombres complexes dits descripteurs de Fourier ou harmoniques.

Le vecteur caractéristique est composé de 16 coefficients, ce dernier est suffisant pour la reconstruction du contour du caractère *Figure II.20*.

L'algorithme utilisé pour le calcul des seize coefficients est :

POUR chaque caractère **FAIRE**

- Lecture du modèle de caractère à partir du fichier.
- Extraction des N points ordonnés de contour extérieur.
- Représentation complexe $Z(n)$ de ce contour (*équation II.15*).
- Calcul de la FFT du signal complexe (*équation II.16*).
- Sélection des 16 premières harmoniques.

FIN FAIRE**2.4 Constitution du vecteur de caractéristiques**

On obtient un vecteur de 79 caractéristiques qui regroupe les primitives réparties de la manière suivante :

09 primitives de type paramètres de distribution par région.

07 primitives de type moments invariant appliquées à l'image binaire.

07 primitives de type moments invariant appliquées au contour extérieur du caractère.

16 primitives des coefficients de Fourier.

32 primitives de l'histogramme de direction.

08 primitives de la densité des profils.

2.5 Classification et reconnaissance

2.5.1 Implémentation de réseaux

Le classifieur que nous avons utilisé est un PMC *Figure III.10* à rétropropagation de gradients d'erreurs à une couche cachée.

Les 79 primitives qui représentent les caractéristiques possibles pour chaque caractère, développées précédemment sont les entrées du réseau.

La couche cachée est composée de 18 neurones.

Les classes à discriminer sont 21 classes qui représentent tous les caractères des montants littéraux, d'où le choix de 21 neurones pour la couche de sortie. La fonction d'activation des neurones est la fonction sigmoïde unipolaire.

2.5.2 L'apprentissage

La base d'apprentissage représentant 21 caractères (classes) différents et pour chaque caractère nous avons pris 151 images de prises de vues différentes.

Donc, pour l'apprentissage nous avons utilisé une base constituée de 315 images. Après avoir effectué plusieurs tests pour fixer les caractéristiques adéquates du réseau, nous avons procédé à la phase d'apprentissage en utilisant un nombre d'itérations considérable pour minimiser l'erreur, ce qui nous a mené au graphe de la *Figure IV.5*.

5.5.3 Reconnaissance

Pour tester la fiabilité de notre système, nous avons effectué des tests sur la base d'apprentissage elle-même, nous avons obtenu des taux de reconnaissance de 100% pour un seuil de rejet égal à 0.002, Ensuite, nous avons effectué les tests de reconnaissance sur une base de test de 100 mots. Les résultats sont illustrés aussi dans le *tableau IV.1*.

CHAPITRE IV

RECONNAISSANCE DE CARACTERES LATINS

APPROCHES GLOBALES POUR LA RECONNAISSANCE DES MOTS ARABES MANUSCRITS

1. Introduction

Nous allons présenter dans ce chapitre les résultats obtenus sur l'extraction des primitives caractérisant l'écriture manuscrite arabe, ainsi que les résultats obtenus pour la reconnaissance en utilisant le PMC.

2. Caractéristiques de l'écriture arabe

Les caractères arabes diffèrent des autres types d'écriture (latine, Chinoise,...) par leur propre structure et leur mode de liaison pour former un mot, on peut citer :

- Le texte arabe est écrit de droite à gauche.
- L'arabe a 29 caractères de base et il n'y a pas la notion de minuscules et de majuscule.
- Les caractères arabes peuvent avoir quatre différentes formes, selon ces positions dans un mot (début, à l'intérieur du mot, extrémité ou isolé).
- Contrairement aux caractères latins la plupart des caractères arabes d'un mot sont reliés le long d'une grande ligne. Beaucoup de caractères arabes ont des points au-dessus placés à une distance appropriée ou au-dessous du corps de lettre appelé points diacritiques. Les points peuvent être un seul point, deux, ou trois. Différentes lettres arabes peuvent avoir le même corps et différer dans le nombre de points les identifiant *Figure V.1.*

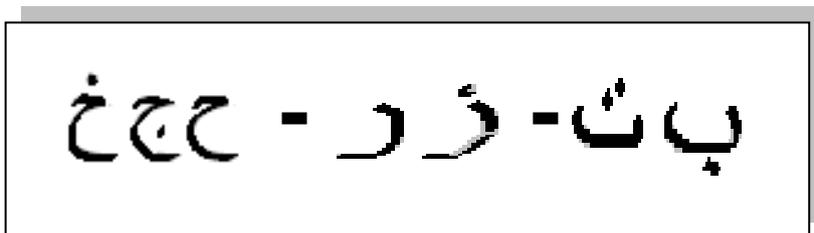


Figure V.1: Caractères arabe ont un même corps.

- Quelques caractères arabes utilisent des marques spéciales pour modifier l'accent de lettre, tel que **hamza**(ء), **mada**(آ), qui sont encore placés à une certaine distance des lettres [2].
- Comme l'écriture arabe est une écriture calligraphique, six styles graphiques différents sont utilisés :Tholoth, Naska, Requeh, Dewan et coufique .

-Un mot arabe peut être composé d'un ou de plusieurs composantes connexes 'c.c' (groupes des points liés entre eux) *Figure V.2 -a-*

-L'Arabe emploie un autre type de caractère spécial en tant qu'ils sont un peu voyelles, qui sont désigné sous le nom de diacritiques. Bien que différents diacritiques sur les mêmes caractères pourraient mener à différents mots, un lecteur arabe est formé pour déduire la signification d'un texte non diacritique, c'est pourquoi des diacritiques ne sont pas employés en journaux ou dans la correspondance de bureau. Cependant, quand des diacritiques sont employés ils apparaissent au-dessus ou au-dessous des caractères *Figure V.2 -b-*.

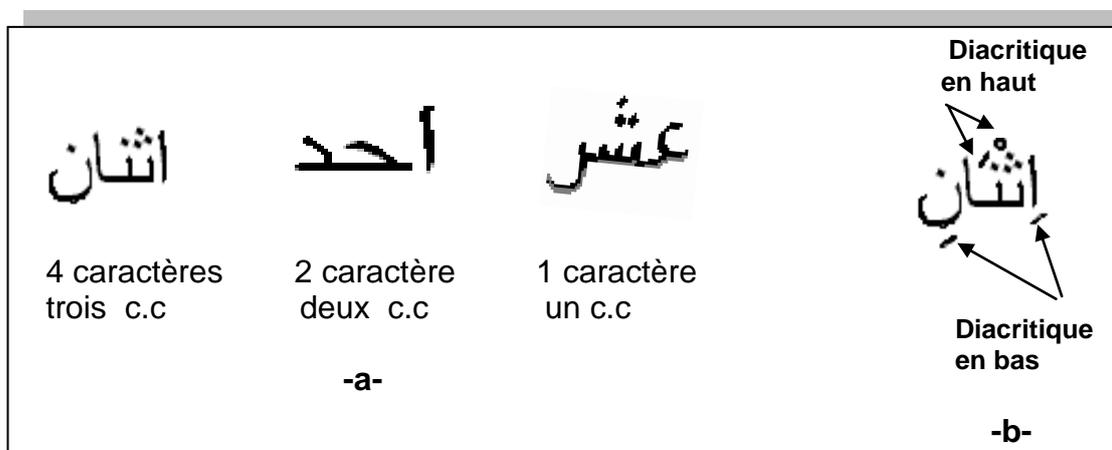


Figure V.2: -a- différentes composantes connexes (c.c) -b- les diacritiques.

Plusieurs recherches et des études intensives ont été entreprises dans le domaine de l'écriture manuscrite, et plus spécifiquement à la reconnaissance de caractères latins. Par contre la reconnaissance de l'écriture arabe n'est pas assez abordée comme les autres langues (l'anglais par exemple) avec la même intensité.

La nature de l'écriture arabe pose de nouveaux problèmes au système de reconnaissance automatique (des problèmes qui ne sont pas rencontrés lors de la reconnaissance des caractères latins), comme on peut trouver des mots qui ne sont pas séparés tout le long d'une ligne (la plupart de caractères arabes d'un mot sont reliés le long d'une grande ligne), l'absence de séparation des caractères dans un mot écrit en arabe, et en plus la présence fréquente des ligatures les rendre difficilement adaptée surtout aux approches de reconnaissance basées sur la segmentation des caractères [14].

3. Phases de conception du système de reconnaissance

L'architecture globale de notre système de reconnaissance est structurée

comme suit *Figure V.3*

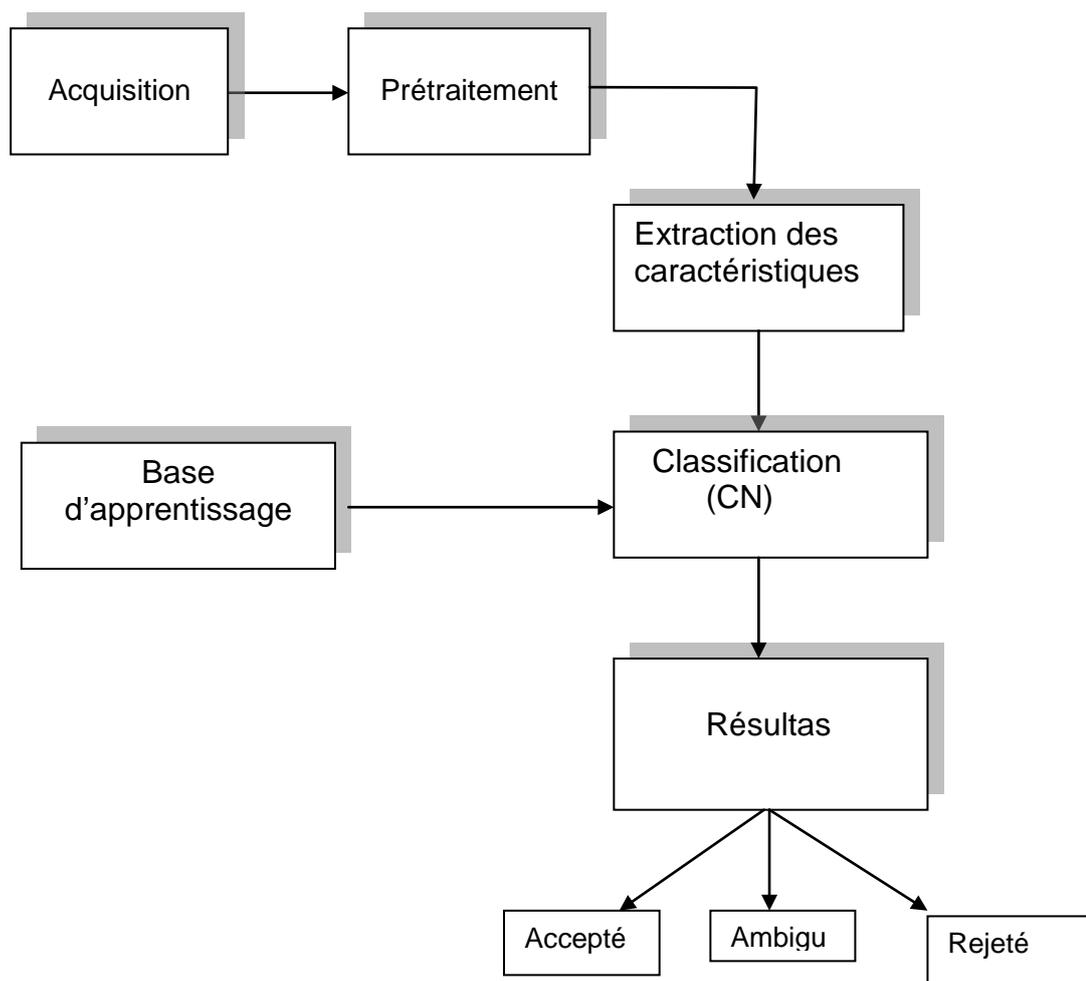


Figure V.3: Architecture générale du système de classification neuronale.

Nous devons mentionner que nous nous intéressons dans ce travail à l'élaboration d'un système de reconnaissance de l'écriture à vocabulaire limité car l'application visée derrière cette architecture est la classification des montants littéraux des chèques postaux.

Pour cela, on a adopté l'approche globale qui se base sur la reconnaissance de tout le mot en le considérant comme une seule entité sans être obligé de passer par la phase de segmentation en caractères. Cette dernière qui est un vrai problème complexe à résoudre car elle provoque une multitude de difficultés, et de confusion de choix des points de segmentation.

Dans ce qui suit nous détaillons les différentes phases adoptées pour la réalisation de ce système.

3.1 Acquisition :

Comme nous n'avons pas une base réelle de montants littéraux de chèques, nous les avons écrit sur de pages de papier blanc que nous avons scanné avec une résolution de 200 dpi.

Pour un montant de chèque écrit en arabe ne diffère des autres que par le vocabulaire utilisé. Quelque soit ce montant, il est limité à un vocabulaire de **48 mots** voir *Tableau V.1.*

Mot	code	Mot	code	Mot	Code	Mot	code
عشر	1	اثنا	13	مليوناً	25	ستون	37
عشرة	2	اثتان	14	مليونان	26	سنة	38
احد	3	جزائري	15	ملياراً	27	ثلاثة	39
آلاف	4	خمسة	16	ملياران	28	ثلاثمائة	40
ألف	5	خمسمائة	17	مليار	29	ثلاثون	41
ألفا	6	خمسون	18	عشرون	30	ثمانية	42
ألفان	7	مائتا	19	سبعة	31	ثمانمائة	43
أربعة	8	مائتان	20	سبعمائة	32	ثمانون	44
أربعمائة	9	مائة	21	سبعون	33	تسعة	45
أربعون	10	ملايين	22	سنتيم	34	تسعمائة	46
دنانير	11	ملايير	23	سنتيمات	35	تسعون	47
دينار	12	مليون	24	ستمائة	36	و	48

Tableau V.1: Vocabulaires des montants littéraux de chèques.

3.2 Prétraitement

Les opérations de prétraitements préparent le fichier de l'image pour les étapes suivantes du processus de reconnaissance. Parmi les différentes opérations citées dans le paragraphe 5.1 du Chapitre II, on a retenu dans notre système les suivantes : la binarisation, la correction de l'inclinaison, le lissage et l'extraction des contours extérieurs.

3.3 Détection et choix des primitives :

Un des problèmes fondamentaux de la reconnaissance de l'écriture est de

déterminer quelles caractéristiques employer pour avoir le bon résultat de la classification, surtout si l'approche appliquée de reconnaissance est globale comme dans notre cas.

Les caractéristiques que nous avons choisies dans notre système sont :

3.3.1 Caractéristiques structurelles :

Nous avons choisi un ensemble de caractéristiques structurelles pour décrire les différents mots de notre lexique, qui sont

- Les nombres des composantes connexes.
- Le nombre de boucles.
- Le nombre de hampe.
- Le nombre de jambage.
- Le nombre de points diacritiques en haut.
- Le nombre de points diacritiques en bas.

3. 3.1.1 Les nombres des composantes connexes :

La limitation du lexique est un avantage de la reconnaissance de chèques ; chacun des **48 mots** de notre lexique est composé de : *une, deux, trois, ou quatre composantes connexes*

Nous avons remarqué que la majorité des scripteurs respectent en général deux règles dans l'écriture des mots arabes ; qui sont :

- les présences des signes diacritiques.
- La séparation des composantes connexes.

Ces avantages ont été le soutien de la décomposition du vocabulaire des 48 mots en quatre sous ensembles, selon le nombre de composantes connexes. Le tableau v.3 montre les quatre sous ensemble.

Par la suite la détection du nombre de composantes connexes nous sert comme clé de référence au sous vocabulaire concerné.

3. 3.1. 2. Le nombre de boucles pour chaque c.c :

La détermination des boucles, s'effectue après la détection des contours de la composante connexe, (paragraphe 5.3.1.1.2, chapitre II).

Notant que chaque composante connexe contient une ou deux boucles au maximum.

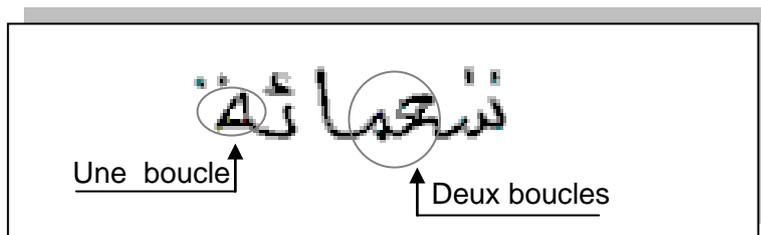


Figure V.4: Exemple d'un c.c. contient un et deux boucles.

3. 3.1.3. Le nombre de hampe et jambage pour chaque cc.

La méthode la plus simple utilisée pour détecter les hampes, les jambages et médiane est basée sur l'analyse de l'histogramme horizontal de l'image.

3. 3.1.4. Le nombre de points diacritiques

L'écriture arabe est riche en points diacritiques. L'écriture de ces points est généralement, respectée par les scripteurs car ces derniers permettent de distinguer entre les caractères ayant les même corps principal. Ces points peuvent être un seul point, deux, ou trois *Figure V.5*. De plus ces points sont en dehors de la zone d'information principale (zone médiane) et ils sont simples à détecter. En contre partie les points simples sont de taille assez faible, ce qui les rend sensibles aux

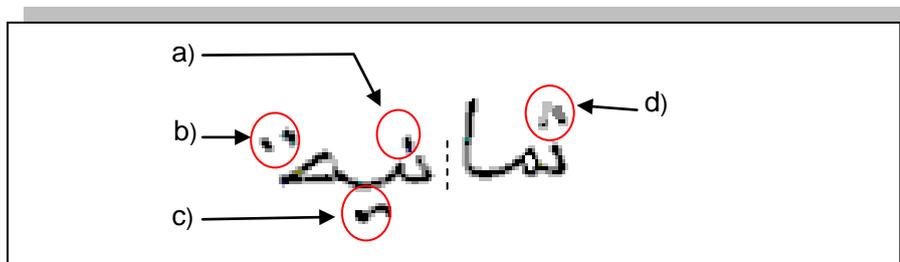


Figure V.5: Détection de diacritique - a) Un point simple - b) Deux points simples - c) Deux points liés - d) Trois points liés.

bruits d'acquisition. Les points multiples par contre, sont des formes complexes, car ils présentent un regroupement de points d'un même caractère. Nous détectons les points par leurs dimensions dans le mot.

A titre d'exemple, le nombre de points diacritique de chaque c.c pour la *Figure V.5* est illustrée dans le tableau suivant :

	Code	c.c1	c.c2	c.c3	cc4
Un point bas	Pb	0	0	0	0
Un point haut	Ph	0	1	0	0
Deux points bas	Dpb	0	1	0	0
Deux points hauts	Dph	0	1	0	0
Trois points hauts	Tph	1	0	0	0
2 Deux points hauts	2Dph	0	0	0	0
2 Deux points bas	2Dpb	0	0	0	0

Tableau V.2 : Représentation de points diacritique de chaque cc
De la figure V.5.

Mots composés de quatre Composantes connexes	Mots composés de trois composantes connexes	Mots composés de deux composantes connexes	Mots composés d'une composante connexe
أربعون	اثنان	ستون	خمسة
أربعمائة	أربعة	ثلاثة	سنة
ملياران	عشرون	ثمانية	سبعة
جزائري	ثلاثون	تسعمائة	عشر
	ثمانون	تسعون	سنتيم
	مائتان	ستمائة	و
	ثلاثمائة	سنتيمات	
	ثمانمائة	سبعون	
	ألفان	سبعمائة	
	آلاف	مليار	
	ملونان	اثنان	
	مليارا	مليون	
	دينار	ملايير	
	دنانير	ملايين	
		مائة	
		مائتا	
		خمسون	
		خمسمائة	
		عشرة	
		احد	
		ألف	
		ألفا	

Tableau V.3 : Décomposition du vocabulaire.

3.3.2 Caractéristiques géométrique

Dans notre application nous avons choisi : *le vecteur caractéristique histogrammes de direction (HD)*.

L'extraction de ces caractéristiques est réalisée sur l'image des contours de chaque composante connexe *Figure V.6*

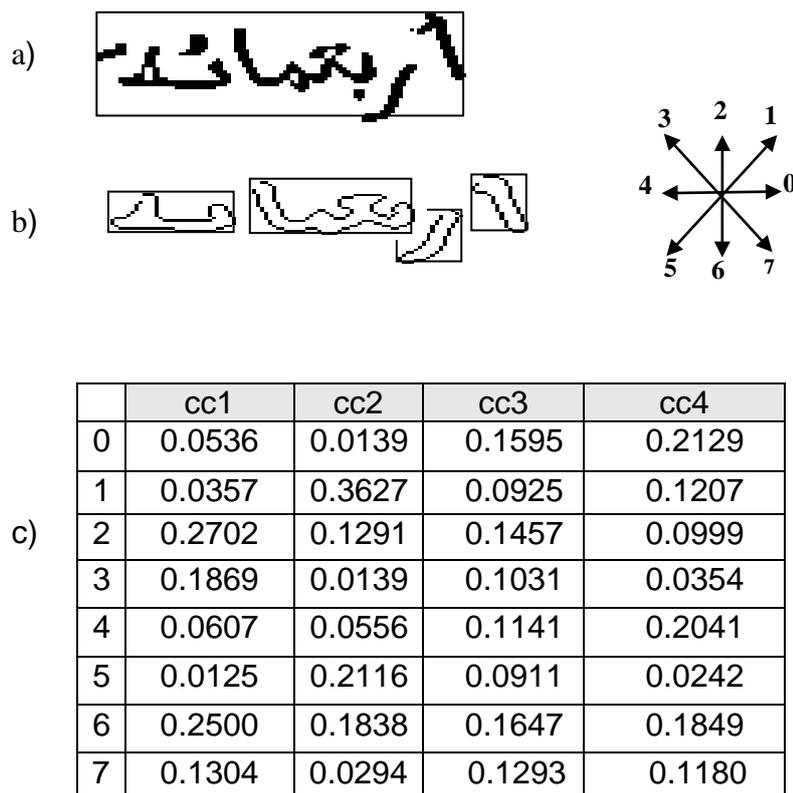


Figure V.6: Caractéristique histogrammes de direction

- a)- Image originale.
- b)- Le contour extérieur de chaque c.c.
- c)- Matrice de direction normalisée.

3.4 Constitution du vecteur de caractéristiques

Pour chaque trait de composante connexe le nombre et un vecteur de 20 primitives sont obtenus (13 caractéristiques structurelles et 8 caractéristiques géométriques). Ce vecteur gardera cette cardinalité pour toutes les autres c.c.

Le format de vecteur caractéristique est illustré dans la *Figure V.8*.

	caractéristique	code
Structurelles	Un boucle	u_b
	Deux boucles	D_b
	Hampe	H
	jambage	J
	Hampe et jambage	H_J
	Médiane	M
Géométriques	Un point haut	Ph
	Un point bas	Pb
	Deux points haut	Dph
	Deux points bas	Dpb
	Trois points haut	Tph
	2de deux point haut	2Dph
	2 de deux points bas	2Dpb
	<i>8 caractéristiques</i>	
	HD	

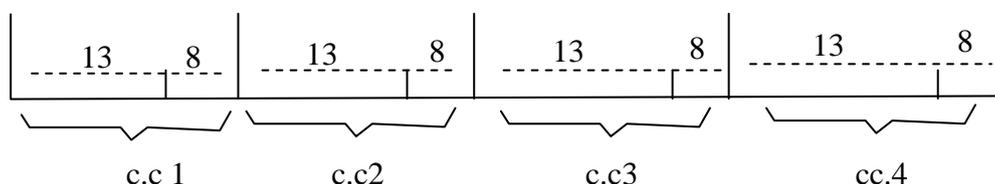


Figure V.7: Présentation du vecteur de primitive.

4. Classification :

4.1 Implémentation de réseaux

Le classifieur que nous avons utilisé est un PMC à rétro-propagation de gradient d'erreur à une couche cachée. Les 84 primitives qui présentent les 21 caractéristiques possibles pour chacune des 4 composantes connexes développées précédemment sont les entrées du réseau. La couche cachée est composée de 25 neurones. Les classes à discriminer sont les 48 qui représentent les mots du vocabulaire des montant littéraux, d'où le choix de 48 neurones pour la couche de sortie. La fonction d'activation des neurones est la fonction sigmoïde unipolaire.

4.2 Apprentissage

La base d'apprentissage représentant 48 mots différents et pour chaque mot nous avons pris 5 images de prises de vues différentes. Donc, pour l'apprentissage nous avons utilisé une base constituée de 240 images. Après avoir effectué plusieurs tests pour fixer les caractéristiques adéquates du réseau, nous avons procédé à la phase d'apprentissage en utilisant un nombre d'itérations considérable pour minimiser l'erreur, ce qui nous a mené au graphe de la *Figure V.8* où l'erreur a atteint la valeur 0.0011.

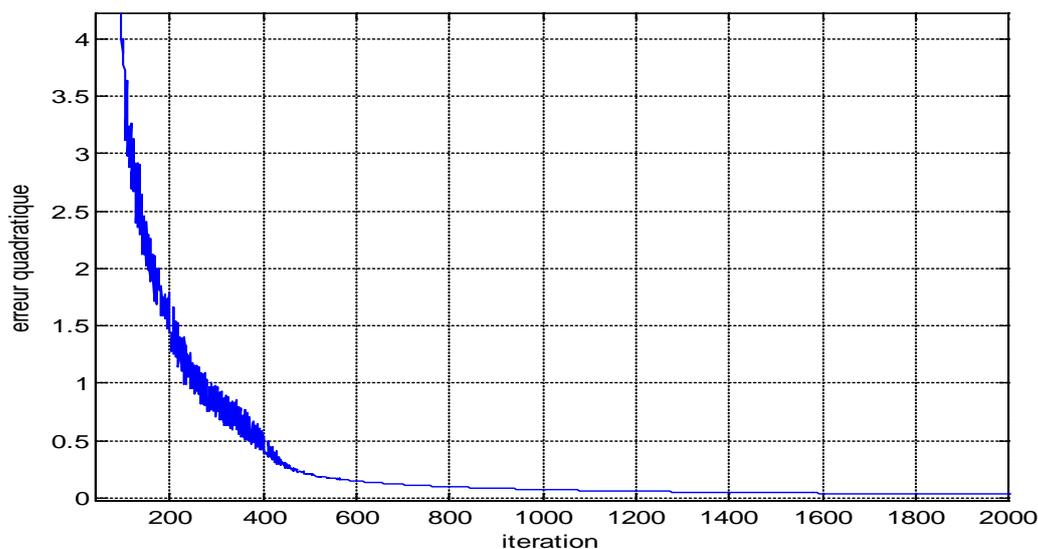


Figure V.8: Evolution de l'erreur en fonction du nombre d'itérations.

4.3 Reconnaissance

Pour tester la fiabilité de notre système, nous avons effectué des tests sur la base d'apprentissage elle-même, nous avons obtenu des taux de reconnaissance de 100% pour un seuil de rejet égal à 0.001. Ensuite, nous avons effectué les tests de reconnaissance sur une base de test de 144 mots. Les résultats sont illustrés aussi dans le *tableau V.4*.

	Taux de Rec (en%)	Taux de Rejet (en%)
Base d'apprentissage	100%	0%
Base de test	81,25%	18.75%

Tableau V.4 : Taux de reconnaissance sur les ensembles d'apprentissage et de test.

CONCLUSION GENERALE

Malgré les efforts et les travaux réalisés dans le domaine de la reconnaissance d'écriture manuscrite aucun système n'est jugé fiable à 100%. Aussi, c'est un domaine qui reste ouvert aux propositions et aux expérimentations.

C'est dans ce cadre que se place notre travail. En effet, les travaux présentés dans ce mémoire abordent les différentes étapes nécessaires à la construction d'un système de reconnaissance de montants littéraux, à savoir : les pré-traitements, la segmentation, l'extraction des primitives et la classification.

Notre orientation s'est focalisée sur la segmentation et la classification. Ainsi, dès l'étape d'extraction des primitives, la sélection des primitives pertinentes et du système est effectuée. Cette sélection consiste à réduire les entrées du classifieur (réseau de neurones de type MLP) tout en améliorant ou en maintenant le taux de reconnaissance de la classification.

Cette modeste expérience s'est avérée intéressante et encourageante mais elle n'a pas la prétention d'être parfaite car on peut toujours envisager des possibilités d'extensions futures, telles que :

- L'utilisation d'une base réelle d'images extraites à partir de vrais chèques.
- L'automatisation de la segmentation du montant littéral.
- L'automatisation de l'étape de compilation : règles du réseau neuronal.
- L'élargissement des champs d'application à d'autres vocabulaires de mots arabes ou latins.

BIBLIOGRAPHIE

- [1] M. Leroux, J-C. Salome et J. Badard: "Recognition of Cursive Script Words in A Small Lexicon". in Proc. Int'l. Conf. Document Analysis and Recognition, pp.774, Saint Malo, France, Sep. 1991.
- [2] B .Lemarieé, M.Gilloux, M.Leroux : "Un modèle neuro-markovien contextuel pour la reconnaissance de l'écriture manuscrite". Actes du congrès AFCET'96 de reconnaissance des formes et intelligence artificielle, 1996.
- [3] H.Miled : "Stratégies de reconnaissance de l'écriture semi cursive: Application aux mots manuscrits arabes" Thèse de doctorat, Université de Rouen, 1998
- [4] M.Chrieriet et C.Y. Suen : "Un système neuro-flou pour la reconnaissance de montants numériques de cheques arabes". Pattern Recognition letters 14(1993),pp, 1009-1017.
- [5] Alessandro L. Koerich: "Large vocabulary off-line handwritten word recognition".Thèse de Docteur Ecole de Technologie Supérieur Université du Québec.2002
- [6] E.Marisa: "Automatic recognition of handwritten dates on brazilian bank cheques". Thèse docteur école de technologie supérieure université du Québec, 2003.
- [7] A.Nazif,F.T.Yarmam-Vural : " Optical Character Recognition for Cursive Handwriting". IEEE Transactions on Pattern Analysis and Machine Intelligence VOL 24, NO 6,JUNE 2002.
- [8] Alceu de S.Britto, R. Sabourin, F.Bortolozzi,C.Y.Suen: "The recognition of handwritten numeral string a two-stage HMM-based method". International Journal on document analysis and recognition, 2003.
- [9] I.Bazzi, R. Schwartz, J.Makhoul : "An Omnifont Open-Vocabulary OCR System for English and Arabic." IEEE Transactions on Pattern Analysis and Machine Intelligence ,VOL 24,NO 6,JUNE 1999.
- [10] A.Kornal: "Expermental HMM-Based Postal OCR System." Proc. Int. conf. Acoustics, Speech, Signal processing, Vol 4, 1997.
- [11] Alin Faure: " perception et reconnaissance des formes". Edition tests 1985.

-
- [12] A.Belaïd et Y. Belaïd : “reconnaissance des formes Méthodes et application ” 1992.
- [13] N.Zermi : “ Reconnaissance de mots manuscrites arabes par un modèle neuromarkovien“. Thèse de Magister d’université, Badji Mokhtar-Annaba, 2000.
- [14] N.ben Amara, N.Ellouze : “A Robust approach for arabic printed character segmentation“. IEEE. Proc. 3rd International conference on document analysis and recognition (ICDAR’95), pp.865-868, Montreal, Canada, 1995.
- [15] S.N.Srihari, E.Cohen, J.J. Hull, Lkuan: “A system to locate and recognize ZIP codes in handwritten addresses” Int. journal of research and engineering, postal application, 1989.
- [16] Théo pavlidis: “Algorithms for graphics and image processing”. Hurray Hill, New Jrsay, 1981.
- [17] Allessandrol.Koerich: “Large vocabulary off-line handwritten word recognition” Thèse de Docteur Ecole de Technologie Supérieur, 2002.
- [18] R.C.Gouzalez et P.Wintz: ”Digital image processing”. Adison-wesley 1987 second edition. Puplicing Company.
- [19] Abdenaïm El-Yacoubi : ”Modélisation Markovienne de l’écriture manuscrite – Application à la reconnaissance des adresses postales”. PhD thesis, Université de Rennes I, 1996.
- [20] Murat Kunt : “Traitement de l’information : reconnaissance des formes et analy-se de scènes”. Vol 3, 2000
- [21] Leroy : “Reconnaissance d’écriture manuscrite dynamique par approche descendante – caractérisation du style de l’écriture et application”. Thèse de Docteur d’Université de Rennes1, avril 1997
- [22] M.Cattoen: “Traitement numériques des images applications pratiques”, Edition.2001.
- [23] N.Ragot: “Reconnaissance de formes par modélisation mixte intrinsèque /discriminante à base de système d’inférence floue hiérarchisés”. Thèse de docteur de l’université de Rennes1, octobre 2003.