

*République Algérienne Démocratique et Populaire*  
*Ministère de l'Enseignement Supérieur et de la Recherche Scientifique*  
*Université des Sciences et des Technologies Mohamed BOUDIAF d'Oran*  
*Faculté des Sciences*  
*Département d'informatique*  
*Laboratoire SIMPA*



**SPÉCIALITÉ : Informatique**

**OPTION : Reconnaissance des Formes & Intelligence Artificielle (RF -IA)**

**MÉMOIRE DE MAGISTER PRÉSENTÉ PAR :**

**M<sup>r</sup>. KADRI MUSTAPHA**

**Thème de mémoire :**

**Méthode de Reconnaissance de l'écriture arabe  
manuscrite en utilisant les réseaux neuronaux**

*Soutenu le .. /.. /2010 devant le jury composé de :*

<i>M<sup>r</sup></i>	<i>Mohamed BENYETTOU</i>	<i>Professeur</i>	<i>USTO-MB</i>	<b>Président</b>
<i>M<sup>r</sup></i>	<i>Abdelkader BENYETTOU</i>	<i>Professeur</i>	<i>USTO-MB</i>	<b>Rapporteur</b>
<i>M<sup>r</sup></i>	<i>Mohamed BENKHADDA</i>	<i>Chargé de cours</i>	<i>USTO-MB</i>	<b>Co-Rapporteur</b>
<i>M<sup>r</sup></i>	<i>Khaled BELKADI</i>	<i>Maître de conférences</i>	<i>USTO-MB</i>	<b>Examineur</b>
<i>M<sup>me</sup></i>	<i>Samira CHOURAQUI</i>	<i>Maître de conférences</i>	<i>USTO-MB</i>	<b>Examinatrice</b>

**Année : 2010**

# Table des matières

## Résumé & Abstract

### Dédicace

Remerciement

Liste des tables

Liste des figures

Abréviations

## Introduction générale ..... 1

## Chapitre I : État de l'art de la reconnaissance de l'écriture arabe manuscrite ..... 3

I.1 Introduction ..... 3

I.2 L'écriture arabe et les OCR ..... 3

I.2.1 Caractéristiques de l'écriture Arabe ..... 3

I.2.2 Conséquences techniques des caractéristiques morphologiques de l'arabe ..... 9

I.2.3 Notion de typographies Arabe ..... 10

I.3 Organisation des systèmes de reconnaissance de caractères (ocr) ..... 11

I.3.1 Phase d'acquisition ..... 12

I.3.2 Phase de prétraitement ..... 13

I.3.3 Phase de segmentation ..... 13

I.3.4 Phase d'extraction des caractéristiques ..... 14

I.3.5 Phase de classification ..... 15

I.3.6 Phase de post traitement ..... 19

I.4 Quelques travaux réalisés dans ce domaine ..... 20

I.5 Conclusion ..... 22

## Chapitre II : Acquisition et prétraitement ..... 23

II.1 Introduction ..... 23

II.2 L'acquisition d'images ..... 23

II.2.1 Les étapes d'acquisition ..... 24

II.2.2 L'image numérique ..... 24

II.2.3 Notion de voisinage ..... 25

II.3 Opérations morphologiques sur les images ..... 25

II.3.1 Transformation par Dilatation ..... 25

II.3.2 Transformation par Érosion ..... 26

II.3.3 Ouverture morphologique ..... 26

II.3.4 Fermeture morphologique ..... 26

II.3.5 Squelettisation ..... 27

II.3.6 Redressement de l'écriture ..... 27

II.3.7 Extraction des composantes connexes ..... 27

II.3.8 Normalisation ..... 28

II.4 Binarisation & nettoyage du bruit ..... 28

II.4.1 Les techniques de binarisation ..... 28

II.4.2 Les techniques de Filtrage numérique .....	30
II.4.2.1 Filtrage linéaire .....	30
II.4.2.2 Filtrage non linéaire .....	32
II.5 Amélioration et restauration d'une image .....	33
II.5.1 Amélioration des images .....	33
II.5.1.1 Diminution de bruits .....	33
II.5.1.2 Renforcement du contraste (traitement d'histogramme) .....	33
II.5.1.3 Rehaussement des contours .....	35
II.5.2 Restauration d'image .....	35
II.6 Détection des contours .....	35
II.6.1 Détecteurs autonomes .....	35
II.6.2 Détecteurs contextuels .....	35
II.6.3 Organisation des Détecteurs .....	36
II.7 Conclusion .....	36
<b>Chapitre III : La segmentation et L'extraction des caractéristiques .....</b>	<b>37</b>
III.1 Introduction .....	37
III.2 La phase de segmentation .....	37
III.2.1 Introduction .....	37
III.2.2 Les techniques de segmentation .....	37
III.2.2.1 Segmentation implicite .....	38
III.2.2.2 Segmentation explicite .....	38
III.2.3 Critères de segmentation .....	39
III.2.4 Les étapes de segmentation .....	39
III.2.5 Les difficultés de segmentation .....	41
III.2.6 Conclusion .....	41
III.3 L'extraction des primitives .....	42
III.3.1 Les approches de l'extraction des caractéristiques (primitives) .....	42
III.3.2 Les catégories des primitives .....	42
III.3.3 Les étapes de l'extraction des primitives .....	45
III.3.4 Les problèmes de l'extraction des caractéristiques .....	46
III.4 Conclusion .....	49
<b>Chapitre IV : classification et reconnaissance .....</b>	<b>50</b>
IV.1 Les Séparateurs à Vaste Marge (SVM) .....	50
IV.1.1 Introduction .....	50
IV.1.2 Principes .....	53
IV.1.2.1 Problème .....	53
IV.1.2.2 Marge .....	53
IV.1.2.3 Espace intermédiaire .....	54
IV.1.3 Séparation linéaire .....	54
IV.1.3.1 Hyperplan séparateur .....	54
IV.1.3.2 Cas non séparable .....	56

IV.1.4 Séparation non linéaire .....	57
IV.1.4.1 Noyaux .....	57
IV.1.4.2 Conditions de Mercer .....	58
IV.1.4.3 Exemples de Noyaux .....	58
IV.1.5 Extension de la formulation SVM .....	59
IV.1.5.1 Stratégies multi-classes .....	59
IV.1.5.1.1 Un contre un .....	59
IV.1.5.1.2 Un contre tous .....	59
IV.1.5.2 One-class .....	60
IV.1.6 Conclusion .....	61
IV.2 Les Réseaux de Neurones .....	61
IV.2.1 Introduction .....	61
IV.2.2 Présentation .....	62
IV.2.3 Fonctionnement d'un réseau de neurones .....	62
IV.2.4 Mise en œuvre d'un réseau .....	64
IV.2.5 Les réseaux à impulsion « SPIKE » .....	67
IV.2.5.1 Inspiration du neurone biologiques .....	67
IV.2.5.2 Codes par cadence .....	67
IV.2.5.3 Codes par impulsions .....	69
IV.2.5.4 Modèle de neurones SPIKE (Artificiels) .....	72
IV.2.5.5 Applications des réseaux de neurones impulsionnels .....	73
IV.2.6 Résumé .....	75
IV.3 Conclusion .....	76
<b>Chapitre V : Résultats expérimentaux et Discussions .....</b>	<b>77</b>
V.1 Introduction .....	77
V.2 Description de la base de données IFN/ENIT DATABASE & Notrebase .....	77
V.3 Ressources matériels et logiciels .....	78
V.4 Description et Évaluation du système AHOCR .....	79
V.5 Interprétation des résultats et critiques .....	89
V.6 Conclusion .....	90
<b>Conclusion générale et Perspectives .....</b>	<b>91</b>
<b>Annexe A</b> Algorithmes de codage et échantillons de la IFN/ENIT DATABASE	
<b>Annexe B</b> Les moments invariants	
<b>Références et bibliographie</b>	

## **RÉSUMÉ :**

Ce présent travail est inscrit dans le domaine de la reconnaissance automatique du texte manuscrit et imprimé, au sens large des termes, et en particulier la reconnaissance de l'écriture arabe manuscrite hors-lignes. Il consiste à la réalisation d'un système complet de reconnaissance de l'écriture arabe manuscrite hors-lignes en utilisant la base de données IFN/ENIT DATABASE. Le système proposé consiste en quatre étapes : le prétraitement du texte, la segmentation du mot en caractères, l'extraction des primitives, et la reconnaissance des caractères par application de deux algorithmes d'apprentissage automatique : le réseau de neurones SPIKE (SNN) (à impulsion) et les *Support Vecteur Machines* souvent traduit par l'appellation de Séparateur à Vaste Marge (SVM) multi-classes.

L'objectif de ce travail est atteint, et les résultats initiaux obtenus sont encourageants et prometteurs. Le taux de la méthode proposée de segmentation des mots en caractères est de 88 %, sur un sous corpus de mots de l'ensemble de Set\_a de la base IFN /ENIT database. Et les résultats obtenus pour les taux de reconnaissance pour les deux approches de reconnaissance sont de 76% pour la méthode SVM, et de 69% pour la méthode SNN, avec deux observations intéressantes pour ces deux algorithmes : la première c'est que la méthode SNN est plus rapide que SVM ; et la deuxième c'est que SVM est stable par rapport au réseau Spike aux variations de la forme des lettres. Ces résultats sont encourageants par rapport à la difficulté liée à la manipulation de l'écriture arabe manuscrite, à la segmentation et à l'extraction des primitives.

**Mots-clés:** *Reconnaissance d'écriture arabe manuscrite, OCR, Caractères arabe manuscrits, Segmentation, Extraction des primitives, Les réseaux de neurones Spike, SVM.*

## **ABSTRACT:**

The present work is released in the automatic recognition of handwritten, typewritten or printed text field, especially the offline Arabic handwritten recognition, its aims to presents the design and implementation of a system that recognizes Handwritten Arabic characters using IFN/ENIT DATABASE. The proposed system consists of four stages: first the preprocessing of the text, second segmentation of the word into individual characters, third feature extraction, and fourth recognition of characters based on two approaches. The first approach in recognition of characters is a *Support Vecteur Machines for multi-class* (SVM) approach, and the second is a Spike Neural Network approach (SNN).

The objective of this work is achieved, and significant experiments were carried out on the IFN/ENIT DATABASE of offline Arabic handwritten. The initial results are very encouraging and promising. The score obtained by the method proposed on the segmentation stage is 88%, which is encouraging. And the score of the recognition of character with SVM and SNN approaches is respectively: 76% and 69%. The experimental results show that the SVM gives the better rate than SNN in recognition, but the SNN is more speed than SVM. The results achieved are due to the complicated of this problem (variation of handwritten, segmentation, feature extraction).

**Keywords:** *Recognition of Arabic handwritten, OCR, manuscript Arabic character, segmentation, features extraction, Spike neural net, Support vector machin es.*

## Dédicace

*En témoignage de tout mon amour, je dédie ce mémoire :*

*A toute la **famille KADRI.***

*A Mr Benyettou Abdelkader l'encadreur et a Mr Benkhadda Mohammed  
le co-encadreur*

*A **mes chers amis et collègues.***

*A toute la promotion RF-IA*

*A tous ceux avec qui j'ai passé des bons moments tout au long de ma vie.*

**A tous qui m'aide durant mes études**

*Mustapha KADRI*

# REMERCIEMENTS

---

Mes remerciements en premier lieu à ALLAH qui m'a donné le pouvoir pour terminer ce modeste travail.

Je tiens à présenter mes remerciements les plus vifs à mon encadreur Mr le professeur **Abdelkader BENYETTOU**, directeur du laboratoire SIMPA (Signale-Image-PArole), d'avoir accepté de m'encadrer et me nourrir de ses précieux conseils, encouragements, et critiques tout au long de ce travail, malgré ses occupations en tant que vice recteur de l'USTO MB.

Je veux remercier également, mon co-encadreur Mr. **Mohamed BENKHADDA** pour leur aide, son suivi pendant la réalisation de ce travail, ses discussions qui ont aidé à résoudre des problèmes rencontrés pendant la réalisation de ce mémoire.

J'exprime également, ma reconnaissance et mon respect au Mr. le président de jury le Professeur **Mohamed BENYETTOU**, qui m'a fait l'honneur de présider le jury de mon mémoire.

Je remercie également les membres de jury : Mr **Khaled BELKADI**, **M<sup>me</sup> Samira CHOURAQUI**, en leur qualité d'examineurs, qui m'ont fait l'honneur aussi d'évaluer ce présent travail. A travers eux je remercie tous mes enseignants qui ont contribué à ma formation durant la première année poste graduation RF-IA.

Et enfin toutes mes reconnaissances à **Mr Dahane KADRI**, et à tous ceux qui, de près ou de loin, ont contribué à l'élaboration de ce travail.

## **LISTE DES FIGURES**

<b>Figure N° I-01</b> : Exemple d'écriture arabe montrant la ligne de base .....	4
<b>Figure N° I-02</b> : Exemple des boucles dans des styles différents .....	6
<b>Figure N°I-03</b> : Exemples de ligatures horizontales et verticales .....	8
<b>Figure N°I-04</b> : Le nom de ville « جنوب افريقيا » ,en forme : normale, souligne gras, italique .....	8
<b>Figure N° I-05</b> : Quelques exemples de fontes Arabe .....	11
<b>La figure N°I-06</b> : Le schéma général d'un système de reconnaissance des caractères .....	13
<b>Figure N°II-01</b> : Voisinage a 4 et Voisinage a 8 respectivement .....	25
<b>Figure N°II-02</b> : Image traitée par Dilatation .....	25
<b>Figure N°II-03</b> : Image traitée par Érosion .....	26
<b>Figure N°II-04</b> : Image traitée par Ouverture .....	26
<b>Figure N°II-05</b> : Image traitée par Fermeture .....	26
<b>Figure N°II-06</b> : La squelettisation d'une Image .....	27
<b>Figure N°II-07</b> : Illustration de la méthode de binarisation par seuil aléatoire .....	29
<b>Figure N°II-08</b> : Illustration du seuillage avec diffusion d'erreur .....	29
<b>Figure N°II-09</b> : Filtrage et traitement des bruits .....	30
<b>Figure N°II-10</b> : Effet de contraste sur les images à niveau de gris .....	34
<b>Figure N°II-11</b> : Fréquence des niveaux de gris pour les images sombres et claire .....	34
<b>Figure N°III-1</b> Exemple d'histogramme horizontal et vertical d'une ligne de texte .....	39
<b>Figure N°III-2</b> Exemple de chevauchement de PAWs respectivement de droite à gauche entre « . » et « . » .....	40
<b>Figure N°III.3</b> Les différentes étapes de l'extraction des primitives et de segmentation .....	48
<b>Figure N° IV-01</b> : Arbre de classification des méthodes d'apprentissage à base de noyaux ...	50
<b>Figure N° IV-2</b> Sous-ajustement linéaire et sur-ajustement local (proches voisins) d'un modèle quadratique .....	52
<b>Figure N° IV-3</b> Recherche d'un hyperplan de séparation optimal au sens de la marge maximale .....	53
<b>Figure N° IV-4</b> Schéma du classifieur une classe dans l'espace des caractéristiques .....	60
<b>Figure N° IV-5</b> Schéma général d'un neurone biologique et artificiel .....	62
<b>Figure N° IV-6</b> Modélisation d'un neurone artificiel .....	63
<b>Figure N°IV-7</b> Fonction de sortie de type seuil, et Fonction de sortie bornée .....	63
<b>Figure N°IV-8</b> Organisation d'un réseau de neurones en couche .....	64
<b>Figure N°IV-9</b> Présentation des données a un réseau de neurones .....	65
<b>Figure N°IV-10</b> Le neurone, ses composants et leur activité électrique .....	67

<b>Figure N°IV-11</b> Code par cadence utilisant la densité des impulsions .....	68
<b>Figure N° IV-12</b> Vague de potentiels d'action générée par une entrée donnée .....	70
<b>Figure N°IV-13</b> Circuit d'un neurone, avec inhibition (/) et entrées pondérées (A, B, C,D et E), sensible à la séquence temporelle : A, B, C, D et E .....	71
<b>Figure N°IV-14</b> Code par le temps de la première impulsion (en haut), code par corrélation et synchronie (en bas) .....	71
<b>Figure N° IV-15</b> Évolution d'un seuil dynamique .....	72
<b>Figure N°IV-16</b> Feedforward spiking neural network, and connection consisting of multiple delayed synaptic terminals .....	75
<b>Figure N° V-01</b> : Des échantillons de la base IFN/ENIT .....	77
<b>Figure N°V-02</b> : Schéma général du système AHOCR développé .....	79
<b>Figure N° V.3</b> : la fenêtre principale de notre système AHOCR .....	79
<b>Figure N°V-04:</b> Description de la méthode de segmentation proposée .....	83
<b>Figure N° V-05:</b> Organigramme de la fonction de recherche des composantes connexe .....	84
<b>La figure N° V- 06</b> : Quelques échantillons de de la base Notre base .....	88

## Liste des tableaux

<b>Tableau N°I-01</b> : L'alphabet arabe dans ses différentes formes. Les caractères additionnels, Madda, la lettre Hamza et les positions qu'elles occupent ( Alif, Waw , Ya) .....	5
<b>Tableau N°I-02</b> : Exemple de mots composé de la droite vers la gauche de 1, 2, 3 PAWs .....	6
<b>Tableau N°I-03</b> : Le PAW « » dans différents mots et différentes positions .....	7
<b>Tableau N°I-04</b> : Exemple de caractères avec et sans madda .....	7
<b>Tableau N°I-05</b> : Caractères susceptibles d'être ligaturé verticalement .....	8
<b>Tableau N° I-06</b> : Tableau récapitulatif précisant les caractéristiques et les performances de certains systèmes AOCR .....	21
<b>Tableau N° V-01</b> : décrit la répartition des mots et le nombre de scripteurs .....	77
<b>Tableau N°V-02</b> : Avec hampe: 4 caractères .....	81
<b>Tableau N°V-03</b> : Avec Jambage: 16 caractères .....	81
<b>Tableau N°V-04</b> : Avec hampe et jambage :un seul caractère .....	81
<b>Tableau N°V-05</b> : Sans hampe sans jambage : 20 lettres .....	82
<b>Tableau N° V- 06</b> : Exemple des caractéristiques de la lettre ' ' .....	86
<b>Tableau N° V-07</b> :Les valeurs des moments invariants ( $\Phi_i$ ) de certains caractères .....	87
<b>Tableau N°V-08</b> : Les résultats de segmentation .....	88
<b>Tableau N°V-09</b> : Les résultats de obtenus par les méthodes de reconnaissance .....	89
<b>Tableau N°V-10</b> : Les résultats de la reconstitution des mots .....	89

# Abbreviations

AHOCR: Arabic Handwritten Optical Character Recognition.

OCR : Optical Character Récognition.

PAW : Peace of Arabie Word.

SNN : Spike Neural Net

SVM : Support Vector Machine.

ASCII : American Standard Code Information Interchange.

ASMO : Arabic Standard Metrology Organization.

CXX : Composantes Connexes.

DSP : Décisive Segmentation point.

HMM : Hidden Markovian Model.

---

---

## *Introduction générale*

Les différents moyens de communication entre l'homme et la machine font toujours l'objet d'études actives. Dans ces dernières années, en informatique, on redouble d'effort pour fournir à l'utilisateur une interface homme-machine aussi simple et aisée à utiliser que possible.

La reconnaissance de formes vise donc à équiper les ordinateurs d'organes de sens, donc on parle à propos des moyens de perception et de raisonnement comparable à celles de l'être humain, cela signifie pas simplement la mise en place des systèmes physiques de capteurs, mais aussi et surtout l'extraction des informations pertinentes, surtout dans les domaines qui nécessitent la précision (Système sensible), par exemple : la reconnaissance des caractères imprimés bruités ou manuscrits, l'identification des plaques des automobiles, ou les numéros de téléphones prononcés (parole) ...etc.

Dès le début de la reconnaissance des formes, la reconnaissance automatique de l'écriture a rapidement pris un envol en s'impliquant avec force dans plusieurs secteurs technologiques (traitement des objets postaux, édition numérique, commerce électronique,...). Elle a pour objet de convertir les documents sur support papier qui sont compréhensibles par l'homme, en images, fichiers textes ou éléments de bases de données, en un code interprétable par un ordinateur. Elle propose des approches différentes suivant le mode d'écriture : manuscrit ou imprimé.

Deux disciplines ont été dégagées pour le mode d'écriture manuscrite qui sont: la reconnaissance *en-ligne* et la reconnaissance *hors-ligne*.

1. *L'écriture en ligne* (ou dynamique) qui est obtenue par une saisie en continu et se présente sous la forme d'une séquence de points ordonnés dans le temps (la donnée est de type signal) en utilisant un *stylet* et une *tablette* connecté au PC.
2. *L'écriture hors ligne* (ou en différé, ou encore statique) qui est obtenue par la saisie d'un texte déjà existant, à l'aide d'un scanner ou d'une caméra (la donnée est une image binaire ou en niveaux de gris).

---

---

Donc selon la discipline choisie, les approches utilisées pour la segmentation, l'extraction des primitives, et la reconnaissance doivent tirer profit de la représentation de la donnée (de ses caractéristiques), et dans certains cas il est nécessaire d'introduire un mécanisme de prétraitement, et de post-traitement (aide à la décision).

Nous nous intéressons dans ce travail à la reconnaissance automatique de l'écriture arabe manuscrite *hors-ligne* (off line).

Le but de ce mémoire est la réalisation d'un OCR de l'écriture Arabe manuscrite, en utilisant la nouvelle base de données IFN/ENIT DATABASE OF HANDWRITTEN ARABIC WORDS (Benchmark libre).

Focaliser l'étude sur la phase de segmentation, sur l'extraction des primitives, et sur la puissance de nouvelles méthodes d'apprentissage automatique, notamment la méthode des séparateurs à vaste marge (SVM) appliquée à plusieurs classes (multi-classes) et un type particulier de réseau de neurones qui est le réseau de neurones à impulsions (Spike SNN), c'est notre objectif principal en traitement et en classification.

Ce mémoire est structuré en cinq chapitres :

Dans le premier chapitre, nous présenterons l'état de l'art de la reconnaissance de l'écriture arabe manuscrite, et les travaux réalisés dans ce domaine.

Le deuxième chapitre sera consacré à l'acquisition et le prétraitement de l'image de texte (binarisation, filtrage, squelettisation,...), permettant la préparation des images pour le système OCR.

Le troisième chapitre expliquera deux phases plus importantes dans l'OCR en général, c'est la phase de segmentation, et la phase de l'extraction des primitives utilisées pour la reconnaissance de l'écriture arabe manuscrite, en donnant les possibilités et les problèmes rencontrés dans chaque étape.

Le quatrième chapitre c'est une description des deux méthodes utilisées pour la reconnaissance de l'écriture manuscrite en donnant les avantages et les inconvénients.

Le cinquième et dernier chapitre permettra d'évaluer les performances des différentes méthodes proposées, et les résultats expérimentaux obtenus seront discutés.

---

---

# **Chapitre I : État de l'art de la reconnaissance de l'écriture arabe manuscrite**

## **I.1 Introduction :**

La reconnaissance de caractères et de l'écriture en général est un domaine actif de recherche pour la science informatique depuis la fin des années 1950 (le premier système réalisé par Mr. Shepard en 1953(IMR)). Au début, on pensait qu'il s'agissait d'un problème facile, mais il apparut qu'il s'agissait d'un sujet beaucoup plus intéressant. Les premiers systèmes de reconnaissance de caractères parus dans les années cinquante se limitaient aux textes anglais et à un petit nombre de polices. À petits pas, ces systèmes sont devenus assez sophistiqués pour reconnaître d'autres langues et d'autres polices. (48 caractères par IBM en 1952; 256 caractères par ASCII en 1965(8bits); 65536 caractères par Unicode en 1990(16bits)). [5]

Parmi les domaines possibles d'application, on trouve le domaine postal pour la reconnaissance des adresses et le tri du courrier, le domaine administratif pour la gestion électronique des flux de documents, les bibliothèques numériques pour l'indexation de documents et la recherche d'information, biométrie pour l'identification du scripteur...etc.

## **I.2 L'écriture arabe et les OCR**

Dans cette partie, nous présentons les caractéristiques morphologiques de l'écriture arabe. Ensuite nous exposons les principaux travaux développés en OCR arabe, tout en soulevant les problèmes majeurs rencontrés dans ce domaine.

### **I.2.1 Caractéristiques de l'écriture Arabe (Calligraphie et typographie)**

L'arabe est écrit par plus de trois cent millions de gens, dans plus de vingt pays différents. L'écriture arabe a été développée à partir d'un type d'Araméen. La langue araméenne comporte moins de consonants que l'arabe, alors de nouvelles lettres ont été créées en ajoutant des points aux lettres déjà existantes. D'autres petites marques appelées diacritiques sont utilisées pour indiquer de courtes voyelles, mais elles ne sont généralement pas utilisées [9].

L'arabe est une écriture consonantique qui utilise un alphabet de 28 lettres (Tableau I-1), auquel il faut ajouter la Hamza « ء », qui est le plus souvent considérée comme signe

---

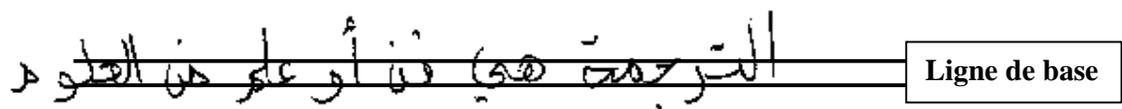
---

complémentaire [2]. La hamza « ء » a une orthographe spéciale qui dépend de règles grammaticales, ce qui multiplie les formes nécessaires à sa représentation, puisqu'elle peut s'écrire seule ou sur le support de trois voyelles ( alif, waw et ya) dont elle suit le code (Tableau I-1). De plus l'alphabet arabe comprend d'autres caractères additionnels tels que « ؤ » et « ة » de ce fait, certains auteurs considèrent que l'alphabet arabe comprend plutôt 31 lettres que 29. La considération du symbole « ~ » qui s'écrit uniquement sur le support du caractère « ء », fait apparaître d'autres graphismes (Tableaux I-1) [11].

L'alphabet arabe n'a qu'un système d'écriture dans lequel les lettres sont liées ou ne sont pas liées entre elles selon des règles précises. Il existe différents styles d'écriture, mais dans aucun d'eux il est possible de juxtaposer des lettres totalement isolées les unes des autres. Il n'y a pas de lettres d'imprimerie en arabe, il n'y a que des caractères typographiques copiés de l'écriture manuscrite. Le caractère arabe est en effet dessiné non pas en fonction des contraintes géométriques des procédés de composition pour imprimerie, mais en fonction de la main et d'une esthétique visuelle héritée de la calligraphie. La fonctionnalité et la lisibilité sont sacrifiées à l'esthétique calligraphique qui substitue l'élégance à la clarté [39].

L'écriture arabe a ainsi plusieurs spécificités que nous citons ci-après.

- Les caractères arabes s'écrivent de rayon cursif, de droite vers la gauche, aussi bien dans le cas de l'imprimé que du manuscrit.
- Les dimensions des caractères (chasse et hauteur) sont variables, même s'il s'agit des différentes formes d'un même caractère (Tableau I-1).
- Plus de la moitié des caractères arabes (16) incluent dans leur forme des points qui peuvent être au nombre de 1, 2 ou 3. ces points peuvent se situer au dessus ou en dessous du corps du caractère, mais jamais en haut et en bas simultanément [38].
  - Un trait caractéristique de l'écriture arabe est la présence d'une *ligne de base* horizontale dite encore ligne de référence ou d'écriture. C'est le lieu des caractères d'une même chaîne (figure I-1).



**Figure N° I-01** : Exemple d'écriture arabe montrant la ligne de base.

**Tableau N° I-01 : L'alphabet arabe dans ses différentes formes. Les caractères additionnels, Madda, la lettre Hamza et les positions qu'elles occupent ( Alif, Waw , Ya).**

N°	Lettre isolé	Au début	Au milieu	A la fin	Prononciation
01	/ /	/ /	/ /	/ /	Alif
02					Ba
03					Ta
04					Tha
05					Jim
06					Ha
07					Kha
08					Dal
09					Dhal
10					Ra
11					Zay
12					Sin
13					Chin
14					Sad
15					Dhad
16					TTa
17					Dha
18					Ayn
19					Ghayn
20					Fa
21					Qaf
22					Kaf
23					Lam
24					Mim
25					Noun
26		ﺀ	ﺀ	ﺀ	He
27					Waw
28		ﻱ	ﻱ		Ya
<i>Autres formes de caractères (complémentaires, combinaison) :</i>					
29		/			Hamza
30	~5		/		Madda + alif
31	/	/	/	/	Lam alif + hamza
32					Ta marbouta
33				/	Lam alif

- La forme d'une lettre écrite dépend de son contexte et le dessin du glyphe associé diffère selon que le caractère apparaît en position initiale, médiane ou isolée dans une chaîne de caractères. A chaque caractère peut correspondre jusqu'à quatre glyphes différents ce qui lève à environ 100 le nombre de formes à reconnaître. Les formes correspondantes à un même

caractère, souvent appelées « formes internes », présentent parfois de sensibles différences, dans certains cas, il est même difficile d'en déduire s'il s'agit d'une même lettre. Cependant le codage ASMO attribue un seul code pour les différentes formes d'un même caractère, contrairement au latin où le code ASCII prévoit deux codes différents pour la même lettre dans sa forme majuscule et minuscule [39]

Certains caractères arabes incluent une boucle qui peut avoir différentes formes (Figure I-02).



*Figure N° I-02 : Exemple des boucles dans des styles différents*

- Certains caractères ne peuvent être rattachés à leur gauche et de ce fait ne peuvent se trouver qu'en position isolée ou finale, ce qui donne quand ils existent, des mots composés d'une ou de plusieurs parties qu'il est convenu d'appeler généralement PAW (*PEACE OF ARABIC WORD*) ou encore pseudo-mot [11]. Un PAW correspond donc à une chaîne d'un ou de plusieurs caractères (Tableau I-2). L'écriture arabe est ainsi semi-cursive plutôt que totalement cursive.

*Tableau N° I-02 : Exemple de mots composé de la droite vers la gauche de 1, 2, 3 PAWs*

3 PAWs/mot	2 PAWs/mot	1 PAWs/mot
العامة	شغال	نقا

Comme le caractère, le PAW peut se trouver dans des mots différents à des positions différentes, mais contrairement au caractère, le PAW présente une structure morphologique stable, il garde la même calligraphie dans les différentes positions qu'il occupe (Tableau I-3)

**Tableau N°I-03 : Le PAW «    » dans différents mots et différentes positions**

Initiale	Médiane	Finale,	Isolé
ء	أ ، اء	ء	

- Pour des raisons de justification de texte et/ou d'esthétique, les ligatures horizontales peuvent être allongées en insérant entre les caractères d'une même chaîne une ou plusieurs élongations « madda » (ou tatwil), correspondant au symbole «    ». L'élongation se situe toujours à gauche du caractère coûtant. Si le trait d'allongement est associé à un caractère en position de début ou finale, le caractère prend sa forme de milieu et voit sa chasse augmenter du nombre de « madda » insérées (Tableau I -4) [39]. Au niveau du PAW, l'insertion de traits d'allongement affecte uniquement sa largeur, la morphologie reste la même [9]. Les éditeurs de texte tels que Word de Microsoft, insèrent dans les lignes de texte, le nombre approprié de « Madda », pour la justification gauche-droite d'un texte arabe.

**Tableau N°I-04 Exemple de caractères avec et sans madda**

Avec 6 maddas	Avec 3 maddas	Avec 1 maddas	Sans maddas

- le mot arabe n'a pas de longueur fixe, il peut comprendre un ou plusieurs PAWs incluant chacun un nombre différent de caractères. De plus, différentes chasses possibles peuvent être associées à un même mot, en insérant un nombre variable de traits d'allongement.

- Dans certaines fontes plusieurs caractères peuvent être écrits de façon combinée. Ces combinaisons ou ligatures, dont le nombre dépasse 1500, sont optionnelles contrairement aux ligatures horizontales qui sont obligatoires [39]. Les ligatures verticales sont utilisées pour des raisons d'esthétique. Elles dépendent du dessin de la police et du degré de qualité artistique du document. Elles peuvent être formées de deux, trois ou quatre caractères et peuvent prendre plusieurs significations selon l'emplacement des points. On parle souvent de ligature de niveau « n » où n désigne le nombre de caractères ligaturés. Les ligatures verticales, souvent composées de façon particulière, peuvent avoir lieu soit au

début ou à la fin du PAW. La ligature classique de niveau 2 peut avoir lieu avec les couples de caractères donnés dans le tableau I-5. [39]

**Tableau N°I-05** Caractères susceptibles d'être ligaturé verticalement

}	}	}	}
{	{	{	{

: ligatures obligatoires des lettres :{ }.

: ligature esthétique entre les 2 premières lettres.

: ligature esthétique entre les 3 premières lettres.

**Figure N°I-03** : Exemples de ligatures horizontales et verticales.

Ce chevauchement modifie les dimensions des PAW et souvent la morphologie de certains caractères (figure I-4). De plus, la ligne de base dans ce cas, n'est plus horizontale [39]. Ce processus dépend fortement des fontes : il existe des fontes qui ne présentent aucune ligature telles que les fontes «madin» et «simplified arabic», d'autres qui possèdent un ensemble de ligatures complètement différentes les unes des autres [39].

- Contrairement au latin, la notion de majuscule et de minuscule n'existe pas en arabe. Cependant, tous les attributs de mise en forme tels que gras, italique, souligné sont valables dans les lettres arabes (figure I-4).

**جنوب إفريقيا – جنوب إفريقيا**

**Figure N° I-04** : Le nom de ville « جنوب إفريقيا », en forme : normale, souligne gras italique.

- Les caractères arabes peuvent être voyellés. Les voyelles appelées aussi diacritiques dans certains documents et courtes voyelles dans d'autres tels que [50], peuvent se placer au dessus ou en dessous du caractère. Les voyelles sont d'une invention postérieure aux consonnes. Dans l'arabe contemporain ordinaire, on écrit seulement les consonnes et les voyelles longues. Un même mot avec différentes voyelles courtes peut être compris

---

---

comme verbe, nom ou adjectif . . . comme par d'exemple «    » peut signifier « drapeau :  
  » ou « savoir :    » ou encore « Enseigner :    », selon sa voyellation.

Il existe 8 signes de voyellation qui peuvent se placer au dessus de la ligne d'écriture, tels que fatha ( ) dammah ( ), soukoun ( ) et chaddah ( ) qui doit être accompagnée de l'une des voyellations fatha, Dammah ou kasrah, et ceux qui peuvent se placer en dessous de la ligne d'écriture tels que Kasrah ( ). De plus trois «tanwin» peuvent être formés à partir d'un double fatha ( ), d'un double dammah ( ) ou d'un double kasrah ( ).

Si en français les 5 signes orthographiques (les accents grave, aigu et circonflexe, le tréma et la cédille) modifient certaines lettres, en arabe toutes les formes de consonnes sont susceptible de porter chacune des huit signes de voyellation et souvent deux d'entre eux superposés (par exemple chaddah+voyelle et chaddah+tanwin). Outre cela et comme le montre ce paragraphe les caractères arabes voyelles nécessitent des matrices de dimensions importantes notamment en hauteur [39]

### **I.2.2 Conséquences techniques des caractéristiques morphologiques de l'Arabe :**

La plupart des critiques apportées à l'écriture arabe sont faites, en général, par comparaison à l'écriture latine imprimée. On reproche à l'écriture arabe d'être sténographique, de ne noter qu'une partie des signes nécessaires à la lecture et donc d'obliger le lecteur à suppléer les signes manquants, c'est à dire de savoir à l'avance ce qu'il doit lire, pourtant ces signes (voyelles) existent. En outre, certains savants considèrent que la superposition des voyelles rend la lecture et l'écriture difficiles et lentes. En fait, ce qu'on reproche sur ce point à l'écriture arabe, c'est de ne pas avoir de voyelles incorporées dans le corps même des mots comme c'est le cas des langues latines où on est obligés de tout écrire (consonnes et voyelles), ce qui permet de tout lire. De plus nous avons vu qu'il existe une centaine de formes différentes de caractères arabes correspondants uniquement aux consonnes, sans aucune ligature ni voyellation, ni chiffre, ni signe de ponctuation. Ainsi les normes de claviers existants (90 à 120) sont largement dépassées. Toutefois, le changement de forme d'un caractère en fonction de sa position dans le PAW, est géré dans les éditeurs de texte par un *analyseur de contexte* qui confère au caractère la bonne calligraphie. C'est aussi par comparaison implicite ou explicite avec l'écriture latine que l'on reproche

---

---

à l'arabe de ne pas avoir de lettres totalement isolées les unes des autres qui seraient utilisables par les techniques de composition de textes.

### **I.2.3 Notion de Typographie Arabe :**

#### **I.2.3.1 Définition de la notion de la fonte (police) :**

Une *police* (fonte) est un ensemble de caractères d'une même famille, de même épaisseur et pour un corps donné. Ces caractéristiques typographiques sont normalisées dans l'imprimerie, tant au niveau du symbole (dimensions et dessin qui représente la forme et l'épaisseur du caractère), qu'à celui de la chaîne (mot : suite de symboles appartenant à la même fonte ou des fontes compatibles) dans chaque ligne du texte [39]

- *La chasse* : comprend en plus de la largeur, l'espace entre caractères. La chasse dépend du dessin, du style et de la grosseur du caractère.
- *Le corps* : désigne la hauteur du caractère comprenant le blanc de séparation horizontale avec la ligne au dessus. Le corps varie en fonction de l'usage prévu pour le caractère : texte courant, titrage ou affiche. La dimension du corps s'exprime en points. Le point est l'unité de mesure typographique, équivalent 0.376 mm.

#### **I.2.3.2 Les styles de calligraphie Arabe:**

L'écriture arabe varie selon les milieux et les régions, d'une extrême simplicité formelle à la complexité exhaustive de l'arabesque. Ces deux formes les plus anciennes sont : l'une souple et cursive à l'origine du « Neskhi », l'autre plus raide et anguleuse, qu'on appela plus tard le « Koufi ». Ces deux écritures ont par la suite donné naissance à d'autres styles. La multiplicité de ces styles est due tout d'abord à la volonté des diverses populations converties à l'islam de conserver les textes coraniques et de les transcrire dans des styles adaptés à leur nature et à leur écriture d'origine. Elle est également due à l'adaptation de l'écriture aux dimensions et à la matière des supports, ainsi qu'au développement de la civilisation : on invente des styles spéciaux pour chaque usage (un style pour princesses, un pour l'administration, un autre pour la poésie etc.). Elle est enfin due à certaines nécessités. Pour des raisons stratégiques, on inventa par exemple le style pigeons voyageurs qui permettait de reconnaître l'expéditeur [39].

---

---

Il existe une centaine de styles dont seulement quelques uns sont couramment utilisés dans le monde arabo-musulman, nous citons par exemple : le Neskhi, Thoulathi, Roqa, Diwani, Koufi, Farsi... (Figure I-5). Le Neskhy demeure aujourd'hui la fonte la plus utilisée pour l'écriture imprimée.

Chaque style est régi par des lois particulières. D'un style à un autre les proportions d'une même lettre changent ; par exemple, dans le style Roqa la lettre 'alif' : « ﺍ » est plus petite que la lettre 'Ain' : « ﺀ », mais dans le style Koufi c'est l'inverse. De même les caractéristiques principales d'une lettre peuvent considérablement changer d'un style à un autre. Par exemple le style d'écriture « *Neski* » c'est une écriture cursive, souple et arrondie, sans aucun angle brusque. C'est le style le plus usité dans les livres, les journaux, il est adapté à la machine à écrire et à l'imprimerie. Il existe aussi des styles modernes qui viennent des styles classiques et qui ont été adaptés aux nécessités des moyens de reproduction. La figure I-5 Montre quelques *fontes Arabe*.

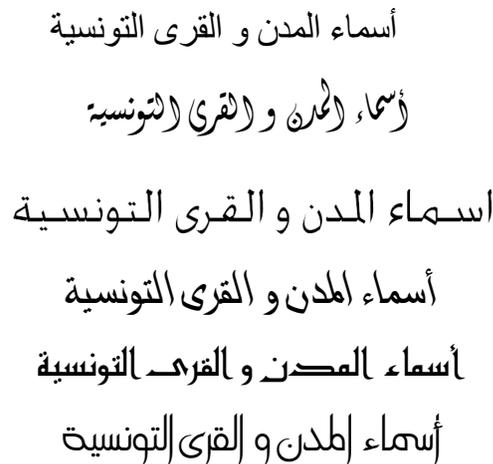


Figure N° I-05 : Quelques exemples de fontes Arabes

### I.3 Organisation générale des systèmes de reconnaissances des caractères (OCR):

L'objectif des systèmes de saisie de données est la conversion de documents sur support papier en images, fichiers textes ou éléments de bases de données. Aujourd'hui, les systèmes de saisie automatique de données remplissent parfaitement cette fonction dans un grand nombre de secteurs, leur degré d'exactitude dépendant de facteurs tels que la qualité de l'image ou la puissance du moteur de reconnaissance, ou bien de la complexité de l'écriture elle-même qui ne permet pas une bonne segmentation [5].

---

---

Ces systèmes entrent dans deux cadres différents : la reconnaissance de texte ou reconnaissance optique de caractères (OCR, Optical Character Recognition), et la reconnaissance de documents manuscrits ou reconnaissance intelligente de caractères (ICR, Intelligent Character Recognition) [2]. Ces deux techniques peuvent être appliquées de deux façons différentes :

- ✓ **Approche globale** : reconnaissance par mots à l'aide de bibliothèques de formes.
- ✓ **Approche analytique** : reconnaissance par caractères à l'aide d'un dictionnaire.

On part de l'image numérique réalisée par un scanner optique d'une page dont on veut récupérer le texte, elle doit être assez contrastée pour que le programme discerne facilement les caractères. Le principe du programme est de lire le document et, grâce à des propriétés pertinentes, détecte les caractères afin d'en faire correspondre la forme au caractère attendu [29]. Dans ce contexte, la reconnaissance passe par deux étapes :

1. *Extraire et coder les informations utiles.*
2. *Classer ces informations, pour établir quel caractère correspond le mieux aux caractéristiques de l'information extraite.*

Un système de reconnaissance fait appel généralement aux différentes étapes suivantes : Acquisition, prétraitements, segmentation, extraction des caractéristiques, classification, suivis éventuellement d'une phase de post-traitement. La figure N°I-06 présente le schéma général d'un système de reconnaissance des caractères, et ses différentes étapes sont expliquées par la suite.

### **I.3.1 Phase d'acquisition :**

Cette phase consiste à capter l'image d'un texte au moyen des capteurs physiques (scanner, caméra,...) et de la convertir en grandeurs numériques adaptés au système de traitement, avec un minimum de dégradation possible.

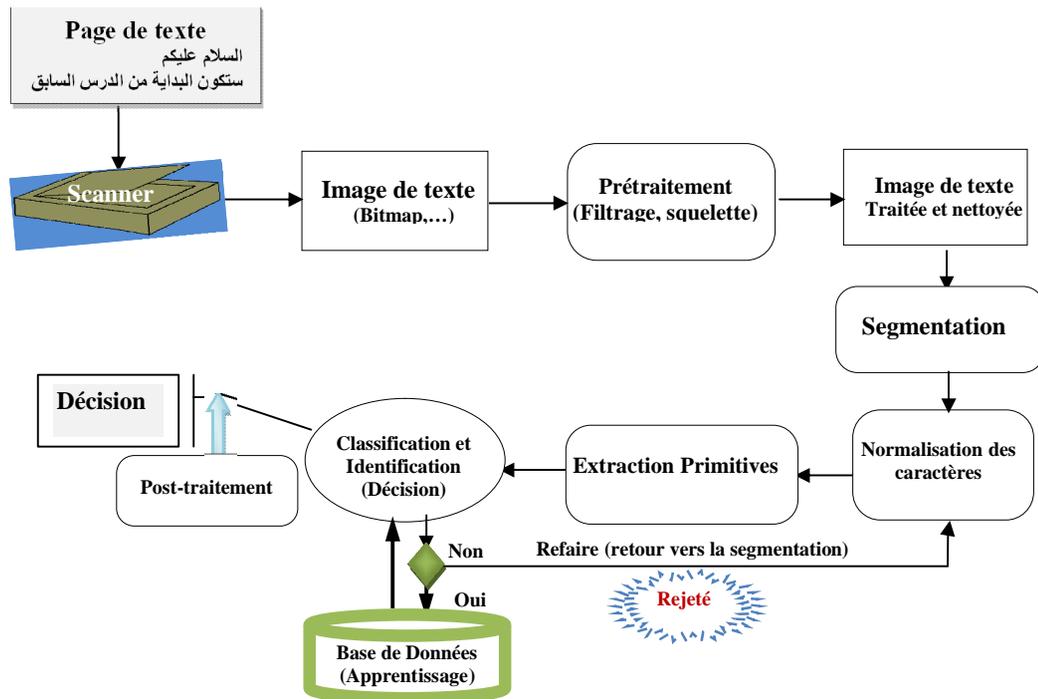


Figure N°I-06 : Schéma général du système de reconnaissance des caractères

### I.3.1.2 Phase de prétraitement :

Consiste à préparer les données issues du capteur à la phase suivante. Il s'agit essentiellement de réduire le bruit superposé aux données et essayer de ne garder que l'information significative de la forme représentée. Le bruit peut être dû aux conditions d'acquisition (éclairage, mise incorrecte du document, ...) ou encore à la qualité du document d'origine. Selon la qualité du document à traiter, le type de l'écriture et la méthode d'analyse adoptée, une ou plusieurs techniques de prétraitement sont utilisées [6]. Mais pas forcément toutes.

Parmi les opérations de prétraitement généralement utilisées on peut citer : l'amélioration, le redressement de l'écriture, le lissage, la normalisation et la squelettisation... Cette étape va être expliquée en détails dans le chapitre II.

### I. 3.1.3 Phase de segmentation :

Dans cette phase les différentes parties logiques d'une image sont extraites. A partir d'une image acquise il y'a d'abord l'extraction des lignes, ensuite à partir de ces lignes sont extraits les mots et l'extraction des composantes connexes, puis les caractères (ou parties du caractère) [9] [11]. Cette phase va être revue en détails dans le chapitre III.

---

---

### **I.3.1.4 Phase d'extraction des caractéristiques (Primitives) :**

C'est l'une des étapes les plus délicates et les plus importantes en OCR. La reconnaissance d'un caractère passe d'abord par l'analyse de sa forme et l'extraction de ses traits caractéristiques (primitives) qui seront exploités pour son identification. Les types des caractéristiques peuvent être classés en quatre groupes principaux : caractéristiques *structurelles*, caractéristiques *statistiques*, *transformations globales*, et *superposition des modèles et corrélation* (plus de détails dans le chapitre III) [9]

#### **a) Caractéristiques structurelles :**

Les caractéristiques structurelles décrivent une forme en termes de sa topologie et sa géométrie en donnant ses propriétés globales et locales. Plusieurs autres caractéristiques peuvent être tirées, suivant qu'ils soient extraits d'une courbe, un trait ou un segment de contour.

#### **b) Les caractéristiques statistiques :**

Les caractéristiques statistiques décrivent une forme eu tenue u un ensemble de mesures extraites à partir de cette forme. Les caractéristiques utilisées pour la reconnaissance de textes arabes sont: le zonage (zoning), les caractéristiques de lieu géométrique (Loci) et les moments [2] [5] .

#### **c) Les transformation globales :**

La transformation consiste à convertir la représentation en pixels en une représentation plus abstraite pour réduire la dimension des caractères, tout en conservant le maximum d'informations sur la forme à reconnaître.

Une des transformations les plus simples est celle qui représente le squelette ou le contour d'un caractère sous forme d'une chaîne de codes de directions [11]. La chaîne de code obtenue est souvent simplifiée pour réduire les redondances et les changements brusques de direction.

---

---

d) Superposition des modèles (Template matching) et corrélation :

La méthode de 'Template matching' appliquée à une image binaire (en niveaux de gris ou squelettes), consiste à utiliser l'image de la forme comme vecteur de caractéristiques pour être comparé à un modèle (Template) pixel par pixel dans la phase de reconnaissance, et une mesure de similarité est calculée [11] [7]

**I.3.1.5 Phase de classification :**

La classification dans un système OCR regroupe deux tâches : *l'apprentissage* et la *reconnaissance et décision*. A cette étape les caractéristiques de l'étape précédente sont utilisées pour identifier un segment de texte et l'attribuer à un modèle de référence [2].

A. L'apprentissage :

Il s'agit lors de cette étape d'apprendre au système les propriétés pertinentes du vocabulaire utilisé et de l'organiser en modèles de références. L'idéal serait d'apprendre au système autant d'échantillons que de formes d'écritures différentes, mais cela est impossible à cause de la grande variabilité de l'écriture qui conduirait à une explosion combinatoire de modèles de représentation. La tendance consiste alors à remplacer le nombre par une meilleure qualité des traits caractéristiques [39] [2]. L'apprentissage consiste en deux concepts différents; l'entraînement et l'adaptation. L'entraînement consiste à enseigner au système la description des caractères tandis que l'adaptation sert à améliorer les performances du système en profitant des expériences précédentes. Certains systèmes permettent à l'utilisateur d'identifier un caractère lorsqu'ils échouent à le reconnaître et ils utilisent l'entrée de l'utilisateur à chaque fois que le caractère est rencontré [31] [29].

Les processus d'apprentissage sont différents selon qu'il s'agisse de reconnaissance de caractères imprimés ou manuscrits ou de reconnaître des textes monospace ou multispaces.

D'une manière générale, on distingue deux types de techniques d'apprentissage : supervisé et non supervisé.

- L'apprentissage est dit *supervisé* s'il est guidé par un superviseur appelé professeur. Il est réalisé lors d'une étape préliminaire de reconnaissance en introduisant un grand nombre d'échantillons de référence. Le professeur indique

---

---

dans ce cas le nom de chaque échantillon. Le choix des caractères de référence est fait à la main en fonction de l'application. Le nombre d'échantillons peut varier de quelques unités à quelques dizaines, voir même quelques centaines par caractère[1][39].

- L'apprentissage *non supervisé* ou *sans professeur* consiste à doter le système d'un mécanisme automatique qui s'appuie sur des règles précises de regroupement pour trouver les classes de référence avec une assistance minimale. Dans ce cas les échantillons sont introduits en un grand nombre par l'utilisateur sans indiquer leur classe [1].

### B. Reconnaissance et décision :

La décision est l'ultime étape de reconnaissance. A partir de la description en paramètres du caractère traité, le module de reconnaissance cherche parmi les modèles de référence en présence, ceux qui lui sont les plus proches.

La reconnaissance peut conduire à un *succès* si la réponse est unique (un seul modèle répond à la description de la forme du caractère). Elle peut conduire à une *confusion* si la réponse est multiple (plusieurs modèles correspondent à la description). Enfin elle peut conduire à un *rejet* de la forme si aucun modèle ne correspond à sa description.

Dans les deux premiers cas, la décision peut être accompagnée d'une *mesure de vraisemblance*, appelée aussi *score* ou *taux de reconnaissance* [39]

Les approches de reconnaissance peuvent être regroupées en trois catégories: l'approche *statistique*, l'approche *structurelle*, l'approche *stochastique*, et à la base de ces trois dernières on peut avoir des approches *hybrides*.

#### 1) Approche statistique :

Elle est fondée sur l'étude statistique des mesures que l'on effectue sur les formes à reconnaître. L'étude de leur répartition dans un espace métrique et la caractérisation statistique des classes, permettent de prendre une décision de reconnaissance du type « plus forte probabilité d'appartenance à une classe » [2]

Les approches statistiques bénéficient des méthodes d'apprentissage automatique qui s'appuient sur des bases théoriques fondées, telles que la théorie de la décision bayésienne, les méthodes de classification non supervisées ... En reconnaissance, le problème revient à affecter

---

---

une forme inconnue à l'une des classes obtenues pendant l'apprentissage [2]. Nous pouvons citer trois méthodes statistiques parmi celles les plus couramment utilisées :

Υ **L'approche bayésienne**

L'approche consiste à choisir parmi un ensemble de caractères, celui pour lequel la suite de primitives extraites a la plus forte probabilité à posteriori par rapport aux caractères préalablement appris [1]

Υ **La méthode du plus proche voisin (KPPV)**

L'algorithme KNN (K Nearest Neighbors) affecte une forme inconnue à la classe de son plus proche voisin en la comparant aux formes stockées dans une classe de références nommée prototypes. Il renvoie 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 (selon la métrique choisie) à la forme inconnue [39]

Cette méthode présente l'avantage d'être facile à mettre en œuvre et fournit de bons résultats. Son principal inconvénient est lié à la faible vitesse de classification due au nombre important de distances à calculer.

Υ **Les réseaux de neurones :**

Un réseau de neurones est un graphe orienté pondéré. Les nœuds de ce graphe sont des automates simples appelés neurones formels. Les neurones sont dotés d'un état interne, l'activation, par lequel ils influencent les autres neurones du réseau. Cette activité se propage dans le graphe le long d'arcs pondérés appelés liens synaptiques [1]

En OCR, les primitives extraites sur une image d'un caractère (ou de l'entité choisie) constituent les entrées du réseau. La sortie activée du réseau correspond au caractère reconnu. Le choix de l'architecture du réseau est un compromis entre la complexité des calculs et le taux de reconnaissance [1]

---

---

Par ailleurs, le point fort des réseaux de neurones réside dans leur capacité de générer une région de décision de forme quelconque, requise par un algorithme de classification, au prix de l'intégration de couches de cellules supplémentaires dans le réseau [1]

## 2) Approche structurelle :

Les méthodes structurelles reposent sur la structure physique des caractères. Elles cherchent à trouver des éléments simples ou primitifs, et à décrire leurs relations. Les primitives sont de type topologiques telles que : une boucle, un arc... et une relation peut être la position relative d'une primitive par rapport à une autre [2]. Parmi les méthodes structurelles nous pouvons citer :

### Υ Les méthodes de test

Elles consistent à appliquer sur chaque caractère traité des tests de plus en plus fins sur la présence ou l'absence de primitives, de manière à répartir les échantillons en classes. Le processus le plus habituel consiste à diviser à chaque test l'ensemble des choix en deux jusqu'à n'obtenir qu'une seule forme correspondant au caractère entre. Ce choix dichotomique est très rapide et très simple à mettre en œuvre, mais il est très sensible aux variations du tracé [39]

### Υ La comparaison de chaînes

Les caractères sont représentés par des chaînes de primitives. La comparaison du caractère traité avec le modèle de référence, consiste à mesurer la ressemblance entre les deux chaînes et à se prononcer sur celui-ci. La mesure de ressemblance peut se faire par calcul de distance ou par examen de l'inclusion de toute ou une partie d'une chaîne dans l'autre [42]

### Υ L'approche syntaxique :

En représentation syntaxique, chaque caractère est représenté par une phrase dans un langage où le vocabulaire est constitué de primitives. Les caractères d'une même famille sont représentés par une grammaire. La reconnaissance consiste à déterminer si la phrase de description du caractère peut être générée par la grammaire. L'inconvénient de cette méthode est l'absence d'algorithmes efficaces pour l'inférence grammaticale directe [41].

---

---

### 3) Approche stochastique :

Contrairement aux méthodes précédemment décrites, l'approche stochastique utilise un modèle pour la reconnaissance, prenant en compte la grande variabilité de la forme. La distance communément utilisée dans les techniques de « comparaison dynamique » est remplacée par des probabilités calculées de manière plus fine par apprentissage. La forme est considérée comme un signal continu observable dans le temps à différents endroits constituant des états « d'observations ». Le modèle décrit ces états à l'aide de probabilités de transitions d'états et de probabilités d'observation par état. La comparaison consiste à chercher dans ce graphe d'états, le chemin de probabilité forte correspondant à une suite d'éléments observés dans la chaîne d'entrée [42]. Ces méthodes sont robustes et fiables du fait de l'existence d'algorithmes d'apprentissage efficaces [31]. Si l'apprentissage est lent, la reconnaissance est par contre très rapide car les modèles comprennent généralement peu d'états et le calcul est relativement immédiat. Les méthodes les plus répandues dans cette approche sont les méthodes utilisant les modèles de Markov cachés (HMM).

### 4) Approche hybride :

Pour améliorer les performances de reconnaissance, la tendance aujourd'hui est de construire des systèmes hybrides qui utilisent différents types de caractéristiques, et qui combinent plusieurs classifieurs en couches [20].

#### **I.3.1.6 Phase de post traitement :**

L'objectif du post-traitement est l'amélioration du taux de reconnaissance des mots (par opposition au taux de reconnaissance du caractère). Cette phase est souvent implémentée comme un ensemble d'outils relatifs à la fréquence d'apparition des caractères dans une chaîne, aux lexiques et à d'autres informations contextuelles. Comme la classification peut aboutir à plusieurs candidats possibles, le post-traitement a pour objet d'opérer une sélection de la solution en utilisant des niveaux d'informations plus élevés (syntaxiques, lexicale, sémantiques...) [39]. Le post-traitement se charge également de vérifier si la réponse est correcte (même si elle est unique) en se basant sur d'autres informations non disponibles au classifieur.

---

---

## I.4 Quelques travaux réalisés dans ce domaine (Avancées en OCR Arabe)

La reconnaissance de l'écriture arabe (AOOCR : Arabie OCR) remonte aux années 70. Depuis, plusieurs solutions ont été proposées. Elles sont aussi variées que celles utilisées dans le latin. Dès les premiers travaux de reconnaissance de l'écriture arabe, les deux modes de reconnaissance, statique et dynamique ont été considérés [2] [20]. L'intérêt a été d'autant porté sur les travaux dans le domaine de l'écriture manuscrite que l'écriture imprimée. Cependant les travaux en-ligne restent relativement peu nombreux.

Dans le tableau suivant nous présentons des travaux réalisés dans cet axe de recherche, la reconnaissance des caractères, mots et textes par différentes méthodes de segmentation et différents mécanismes de reconnaissance et classification.

Le tableau, enrichi par des travaux récents, regroupe certains systèmes de reconnaissance de l'écriture arabe en précisant pour chacun le mode utilisé en -ligne ou hors-ligne, l'approche de reconnaissance globale ou analytique, le type de segmentation, la représentation choisie ainsi que les scores réalisés.

**Tableau N° I-06:** Tableau récapitulatif précisant les caractéristiques et les performances de certains systèmes AOCR[9].

	Système(Hors-ligne)	Primitives	Classification	Performance
Almuallim,87	manuscrit mot	Statistiques	Syntaxique/ Distance	RC 91%
Alaa, 01	Manuscrit	Structurelles	Réseaux de neurones	SC 69.72 %
Aissaoui, 96	MF	Statistiques	Réseaux de neurones	RC64-100%
Aissaoui, 97	MF	Statistiques	Distance quadratique	RC 87.87–95.24%
Burrow, 04	Manuscrit	KNN, moments	-	RC 94%
Al-yousefi, 92	manuscrit.	Moments	Classifieur bayésien.	RC 99.5 %
Ameur, 93	manuscrit MS	structurelles	Arbre de décision	SC98.9- C83%
Ameur, 97	Manuscrit	Structurelles/ statistiques	KNN	RC 82.5 %
Amin, 89	MF	Chaîne de codes	Arbre de décision	RC 90 %
Amin,96	caractères MS	Structurelles	Réseaux de neurones	RC 90-92 %
Amin, 97	Mots	Structurelles	Réseaux de neurones	RC 98%
Azmi ,01	Perse	-		SC 93-98.9 %
Benamara, 95		géométriques		SC 99-100 %
Bouslama 97	caractères isolés	Structurelles/ variables	Logique floue	RC 100 %
Bouslama, 99	caractères isolés	Fuzzy linguistiques	Logique floue	RC 100%
Jambi, 93	Manuscrit	Structurelles	Dictionnaire	SC 95 %
Fahmy, 01	Manuscrit	Géométriques	Neural net Logique floue	RC 69.7 %
Fehri, 94	MF	Structurelles /statistiques	Programmation dynamique	RC 98 %
Sari, 02	Manuscrit	-		SC 86 %
Miled, 96	Manuscrit	Structurelles	-	SC 98.52 %
Miled, 98	Manuscrit	Topologiques/ statistiques	HMMs	RC 79.5-82.5 %
Motawa, 97	Manuscrit	Structurelles	Morphologie mathématique	SC 81.88 %
Olivier, 96	Manuscrit Ms	Chaîne de codes	-	SC 97.41%
Souici, 97	Manuscrit	Statistiques/ Structurelles	Réseaux de neurones	RC76. 17-85.75%
Zahour, 91	Manuscrit	Structurelles	Dictionnaire	RC 86%
Zahour, 98	Manuscrit	Structurelles	Mise en correspondance	RC 87 %

RC : Reconnaissance Caractère, RM : Reconnaissance Mot, SC : Segmentation de caractères, MF : Multifonte, MS : Multiscripteur

---

---

## **I.5 Conclusion :**

Nous avons présenté dans ce chapitre : certains concepts généraux liés à la reconnaissance optique des caractères et de l'écriture en générale, en précisant les principales méthodes de reconnaissance, les principaux problèmes rencontrés par l'OCR, et les principales propriétés morphologiques et typographiques de l'écriture arabe ont été présentées.

A la fin ce chapitre, nous avons présenté un revue de certains travaux qui ont été réalisés dans le domaine de la reconnaissance de l'écriture arabe.

---

---

## **Chapitre II : Acquisition et Prétraitements d'Images**

### **II.1 Introduction:**

Parmi les étapes de la reconnaissance des caractères et de l'écriture c'est la phase de *Prétraitements* qui est relativement standard dans la plupart des systèmes de reconnaissance, et elle a plus d'influence sur la fiabilité de la phase de segmentation et d'extraction des primitives, cette dernière est la plus importante, plus sensible et qui est une étape primordiale dans ce type de système. Dans cette partie les prétraitements des données (images) vont être abordés en détail.

Une fois l'image numérisée, une série de prétraitements est appliquée, d'abord pour extraire des lignes de texte, éventuellement segmentées en mots ou en caractères en détectant les espaces. Cette phase comprend aussi des processus de traitement d'image en général (atténuation du bruit, binarisation..) et d'autres plus spécifiques au traitement de l'écriture manuscrite. Traditionnellement, les prétraitements contribuent à diminuer la variabilité intra-classes par des traitements de normalisation. Toutefois, l'orientation actuelle vise à diminuer le nombre de prétraitements pour gérer la variabilité au niveau des modèles [6]. Les prétraitements utilisés dans les OCR ont les objectifs suivant : amélioration de l'image, le filtrage (atténuation des bruits), la normalisation des caractères, la réduction de la taille des données...etc.

### **II.2 L'acquisition de l'image :**

Nous commençons par un rappel de quelques notions de base qui sont nécessaires pour la suite de cette partie. Le texte est scanné avec une résolution de 200 dpi ( et parfois de 300 dpi) et est stocker sous forme d'image binaire en général. Avant d'être analysée l'image subit quelques prétraitements, dont le redressement, le lissage, et la normalisation.

#### **II.2.1 Les étapes d'acquisition :**

Elle consiste en deux phases :

- **L'échantillonnage (numérisation)** d'une image est spatial, par découpage en pixels.
- **La quantification (codage)** c'est une valeur numérique donnée a l'intensité lumineuse, c'est un niveau de gris, appelé la dynamique de l'image.

---

---

Cette *dynamique* est donnée comme suite :  $2^m$ , ou 'm' est le nombre de bits. Par exemple : le niveau de gris 256 est codé sur 8 bits, l'image couleur est codée sur 24 bits (1 octet pour chaque couleur (R, V, B)) [6].

### II.2.2 L'image numérique :

L'image est un ensemble de points, chaque point est affecté d'une grandeur physique (luminance, couleur, intensité...). Cet ensemble peut être continu : « image analogique » ou discret : « image digitale ». On distingue deux types d'images les images à plusieurs niveaux de gris (y compris les images couleurs) et les images binaires constituées de pixels noirs et blancs [18].

Chaque point (x,y) du plan image est appelé pixel, en anglais « Picture élément ». Un pixel est l'unité élémentaire de l'image qui représente le niveau de gris du point correspondant à sa position.

Alors il y a un compromis entre la définition de l'image et le temps de traitement nécessaire. La section suivante présente les formats de fichiers graphiques les plus courants. Le format que vous choisirez dépend des programmes avec lesquels vous avez l'intention d'utiliser ces images, et de votre désir de compresser les images ou non avant de les enregistrer [80].

1. **BMP**: Le format "Bitmap" est le format graphique standard de Windows. La taille et les traitements sont moins coûteux. Elle se caractérise par : nombre de colonnes, nombre de ligne, la dynamique (nombre de bits par pixel : 8bits pour le niveau de gris 256, et 1 bits pour le Noir et Blanc), la résolution spatiale (nombre de pixels par unité de surface, indiquant la distance que représente un pixel).
2. **EPS**: Le format EPS (*Encapsulation PostScript*) est un format de fichier standard, couramment reconnu par de nombreux programmes de mise en page et de création d'images.
3. **JPEG**: Le format défini par le *Joint Photographic Experts Group* est fréquemment utilisé pour les images destinées à être affichées dans un navigateur Internet. Le format JPEG permet de réduire la taille de fichier de l'image en rejetant certaines données peu importantes de l'image. Par contre, le JPEG conserve la totalité des informations de couleurs d'une image et offre divers degrés de compression.

- 
- 
4. **GIF**: Le format *Graphics Interchange Format* est lui aussi fréquemment utilisé pour les images destinées à être affichées dans un navigateur Internet, et il comporte son propre algorithme de compression, qui permet de supprimer certaines informations de couleurs pour réduire la taille du fichier.

### II.2.3 Notion de voisinage :

- Voisinage a 4 : l'ensemble de pixels 'p' qui ont un coté en commun avec le pixel considéré.

- Voisinage a 8 : l'ensemble de pixels 'p' qui ont au moins un point de liaison avec le pixel considéré.

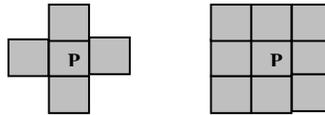


Figure N°II-01 : Voisinage a 4 et Voisinage a 8

### II.3 Les opérations morphologiques sur les images:

Le but des opérations morphologiques est la correction, l'amélioration et la préparation de l'image à traiter; parce que chaque fois qu'une image est convertie d'une forme à une autre, traitée, ou affichée, sa qualité est plus ou moins affectée [18].

Ce type de traitement vise donc à éliminer les bruits, et à corriger les dégradations et les distorsions géométriques lors de l'acquisition de l'image afin d'améliorer et de restaurer l'image à traiter [22]. Elles englobent les transformations qui suivent :

- Transformation par *érosion* et Transformation par *dilatation* d'image (lissage d'image).
- L'*ouverture* et la *fermeture* morphologique appliquée à une image.
- *Squelettisation* d'une image, le *redressement* de l'écriture et la *normalisation*

#### II.3.1 Transformation par dilatation:

Si un pixel 'p' est noir ( $p(x,y)=0$ ), et il y a au moins 3 pixels, ou 7 pixels dans les 4-voisins, ou les 8-voisins respectivement, qui sont blancs ( $P_{voisin}(x,y)=1$ ), alors on affecte

a ce pixel la couleur blanche ( $p(x,y)=1$ ), c. -à-d on efface ce pixel, (C'est un lissage d'un ou deux pixels d'une forme connexe) [35].

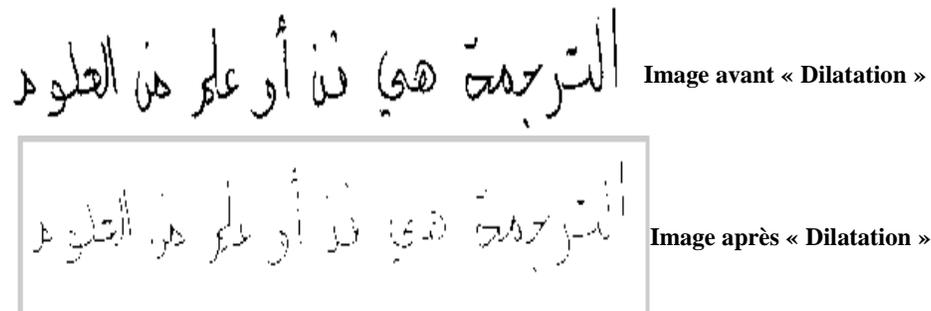


Figure N°II-02 : Image traitée par Dilatation

### II.3.2 Transformation par Érosion :

C'est l'inverse de la transformation par dilatation, si un pixel 'p' est blanc ( $p(x,y)=1$ ), et il y a au moins 3 pixels, ou 7 pixels dans les 4-voisins, ou les 8-voisins respectivement, qui sont noir ( $Pvoisin(x,y)=0$ ), alors on affecte a ce pixel la couleur r noir ( $p(x,y)=0$ ) [35].

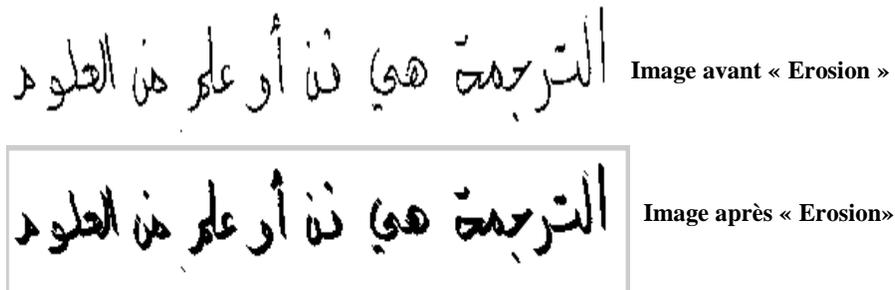


Figure N°II-03 : Image traitée par Érosion

### II.3.3 Ouverture morphologique appliquée a une image:

C'est une combinaison d'opérations : érosion suivie d'une dilatation d'une image par le même élément structurant. Servant à adoucir les contours, simplifie r les formes en lissant les bosses tout en conservant l'allure globale [35].

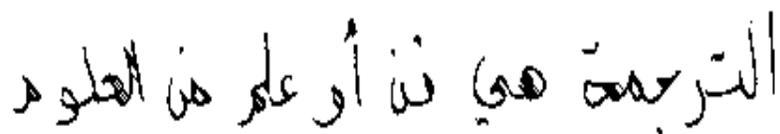


Figure N°II-04 : Image traitée par Ouverture

---

---

### II.3.4 Fermeture morphologique appliquée a une image:

C'est le contraire de l'ouverture, elle consiste en une combinaison d'opérations : une dilatation suivie d'érosion d'une image par le même élément structurant. Elle permet de simplifier les formes, en comblant les creux [35].

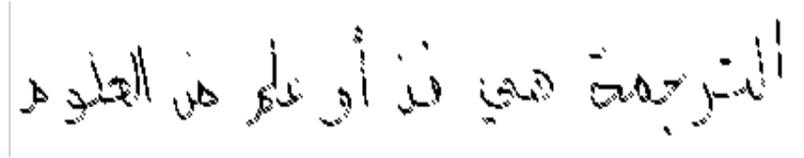


Figure N°II-05 : Image traitée par Fermeture

### II.3.5 Squelettisation d'une image :

Très utilisée dans le domaine de la 'reconnaissance des formes', cette opération consiste à transformer une image binaire en un 'squelette'. Le squelette est un ensemble de lignes d'épaisseurs infiniment petites. La squelettisation doit préserver la connexité de l'image. En d'autres termes, cette opération ne doit ni séparer les éléments connexes, ni raccorder les éléments non connexes. Le but est de simplifier l'image du caractère en une image à « ligne » plus facile à traiter en la réduisant au tracé du caractère [80].

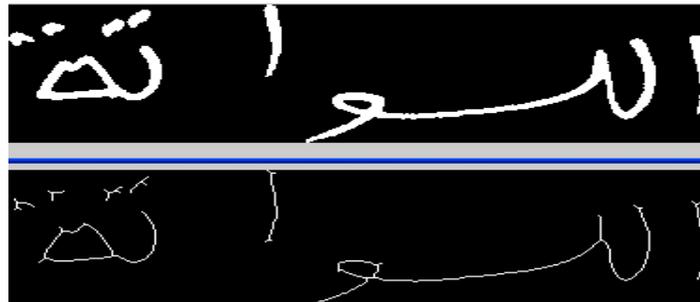


Figure N°II-06 : La squelettisation d'une Image

### II.3.6 Redressement de l'écriture :

L'un des problèmes rencontrés en OCR est l'inclinaison des lignes du texte, qui introduit des difficultés pour la segmentation. L'inclinaison peut être intrinsèque au texte provenir de la saisie, ou si le document a été placé en biais. Il convient alors de le redresser afin de retrouver la structure des lignes horizontales d'une image texte. Si 'a' est l'angle

---

d'inclinaison, pour redresser l'image, une rotation isométrique d'angle '-a' est opérée grâce à la transformation linéaire suivante [80]:

$$x' = x \cos a + y \sin a \quad , \quad y' = y \cos a + x \sin a$$

### II.3.7 Extraction de composantes connexes :

Une composante connexe (CXX) est un ensemble de points liés dans le plan. Elle peut correspondre à un point diacritique, un accent, au corps d'un caractère ou d'une chaîne de caractères... Une fois localisés les CXX sont regroupées pour former les mots. Cette technique est utilisée pour le repérage des points diacritiques dans les images de textes arabes [39] [7].

### II.3.8 Normalisation :

Après la normalisation de la taille, les images de tous les caractères se retrouvent définies dans une matrice de même taille, Pour faciliter les traitements ultérieurs. Cette opération introduit généralement de légères déformations sur les images. Cependant certains traits caractéristiques tels que la hampe dans des caractères ( par exemple) peuvent être éliminées à la suite de la normalisation, ce qui peut entraîner à des confusions entre certains caractères [5] [35].

## II.4 Binarisation et Amélioration d'Image :

### II.4.1 Les techniques de binarisation :

Plusieurs techniques ont été développées dans le but de transformer une image a niveaux de gris ou en couleur, en image binaire. Toutes ces techniques sont basées sur le principe de seuillage comme le montre l'équation suivante :

$$I_b(x,y) = \begin{cases} 0 & \text{si } I_n(x,y) < T \\ 1 & \text{si } I_n(x,y) \geq T \end{cases}$$

$I_b(x, y)$  décrit l'intensité a "n" niveaux de gris a chaque point de l' image,  $I_b(x, y)$  représente l'intensité a deux niveaux et T est le seuil de binarisation. Si  $I_n(x,y)$  est supérieur a la valeur de seuil alors en attribut le point image correspondant a la valeur

---

---

d'intensité maximale (le blanc). Dans le cas contraire, le point est considéré comme noir et en lui attribut la valeur d'intensité minimale [35].

Le but de toutes les techniques de binarisation, est de minimiser la perte de la qualité visuelle que subit l'image, diminuer l'espace occupé par les images résultantes.

Parmi les méthodes de binarisation on a cité quelques technique telles que :

#### II.4.1.1 Méthode de seuil aléatoire :

Dans cette technique, le signal a seuiller est la somme de l'image originale a niveau de gris  $I_n(x,y)$  et d'un bruit  $n(x,y)$ . Ce bruit est pseudo aléatoire, le signal de sortie  $I_b(x,y)$  est une version bruitée de  $I_n(x,y)$  [7]. L'avantage de cette méthode est permettre relativement une bonne différenciation des légères variations de luminosité (élimination des faux contours). Cependant, elle fournit une image bruitée peu plaisante à l'œil.

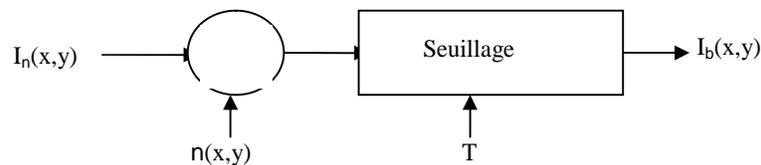


Figure N°II-07 : Illustration de la méthode de binarisation

#### II.4.1.2 Méthode a seuil ordonné :

Dans cette méthode le seuil n'est plus totalement aléatoire, mais il est compose de la répétition bidimensionnelle d'une matrice de seuils ('dither matrix').le calcul du seuil en fonction de la matrice est donne par :

$$T(x, y) = a D_{i,j} + b$$

Dans cette relation  $T(x,y)$  représente la valeur du seuil au point  $(x,y)$ ,  $D_{i,j}$  décrit la matrice de seuils,  $n$  est la taille de la matrice  $D_{i,j}$ ,  $i,j$  sont les coordonnées de  $x$  et de  $y$  modulo  $n$ ,  $a$  et  $b$  permettent de l'adaptation de la dynamique des éléments de la matrice de seuils a celle de l'image. Après le calcul du seuil dépendant de la position, l'image binaire est obtenue par le seuillage (déjà expliquer) [80].

### II.4.1.3 Méthode de la diffusion de l'erreur :

Les méthodes décrites précédemment sont des opérations ponctuelles qui ne font intervenir que le niveau de gris d'un point de l'image d'entrée et un seuil. La méthode de diffusion de l'erreur se différencie de ces techniques par le fait que la différence entre la valeur du niveau de gris et le seuil, appelée 'erreur' est distribuée sur les points voisins les plus proches de l'image ; ceux qui n'ont pas été encore seuils [80].

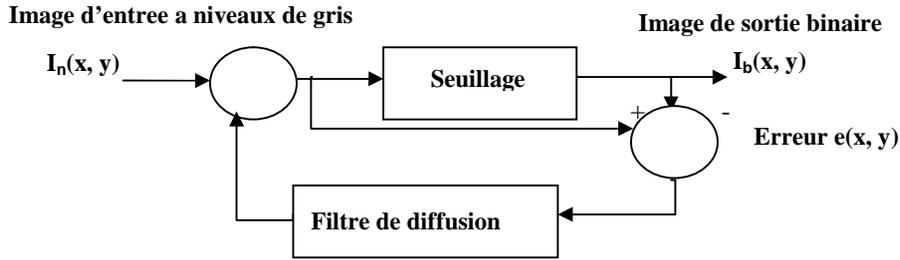


Figure N°II-08: Illustration du seuillage avec diffusion d'erreur

### II.4.2 Les techniques de Filtrage numérique:

Les filtres permettent d'éliminer certaines informations (des supports dont le fond n'a pas une couleur unie), pour mettre en évidence celles qui restent (isoler le mot du motif d'arrière-plan); cela est particulièrement utile pour réduire le bruit (présent sur le support) ou pour arrondir les contours, le filtrage se base sur l'information dans le voisinage autour d'un pixel considéré. Il y a deux types de filtres numériques: non linéaire et linéaire [64].

Dans ce qui suit, nous verrons un exemple de réduction de bruit, par un filtre qui va être défini plus tard :

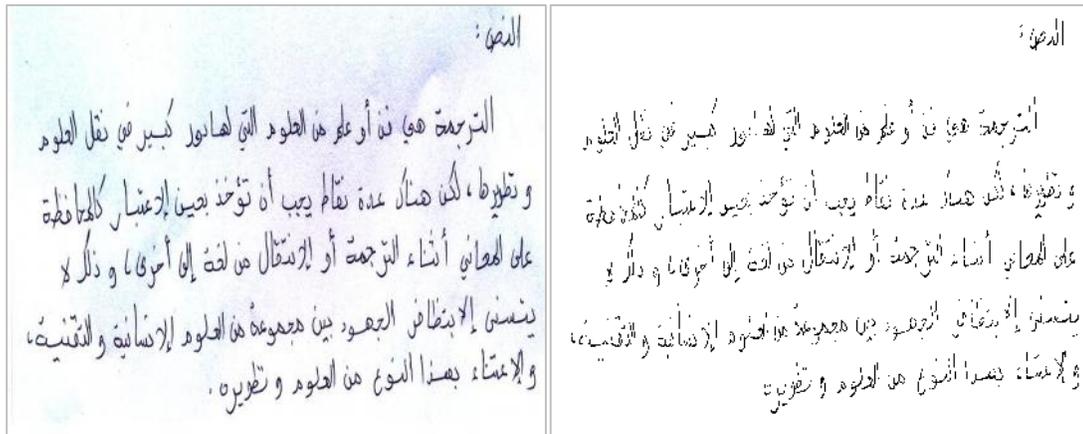


Image bruitée

Image restaurée avec un filtre Médian

Figure N°II-09 : Filtrage et traitement des

### II.4.2.1 Filtrage linéaire :

C'est une technique simple, qui consiste à remplacer chaque point par une moyenne pondérée calculée sur son voisinage. Notons  $P_i$ , le pixel considéré, sa valeur est :

$$P_i = \{A_j * P_j\} ; j=1,n ; \text{ le masque (voisinage):}$$

$X_4$	$X_3$	$X_2$
$X_5$	$X_0$	$X_1$
$X_6$	$X_7$	$X_8$

Tel que :  $A_j$  représente les coefficients du masque utilisé.  $P_j$  représente les valeurs du voisinage considéré [64].

#### a) Filtre passe bas :

C'est un filtre linéaire où les coefficients du masque sont positifs et leur somme est supérieure ou égale à 1 [64]. La formule est la suivante :

$$G(X_0) = \frac{\sum_{i=0}^{i=8} f(X_i) * V(X_i)}{\sum_{i=0}^{i=8} V(X_i)}$$

$G(X_0)$  : la nouvelle valeur du pixel (le centre du masque) de l'image améliorée.

$V(X_i)$  : la fonction de voisinage (valeur du masque du filtre)

$f(X_i)$  : la valeur du pixel de l'image originale.

Il y a des exemples des fonctions de voisinage (masques) pour le filtrage linéaire passe bas, et leur description est la suivante :

<b>H1</b> <table border="1" style="display: inline-table; border-collapse: collapse; text-align: center;"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	1	1	1	1	1	$G(X_0) = \frac{1}{9} \sum_{i=0}^{i=8} f(X_i)$	<b>H2</b> <table border="1" style="display: inline-table; border-collapse: collapse; text-align: center;"> <tr><td>1</td><td>2</td><td>1</td></tr> <tr><td>2</td><td>1</td><td>2</td></tr> <tr><td>1</td><td>2</td><td>1</td></tr> </table>	1	2	1	2	1	2	1	2	1	$G(X_0) = \frac{\sum_{i=0}^{i=8} f(X_i)}{\sum_{i=0}^{i=8} V(X_i)}$
1	1	1																			
1	1	1																			
1	1	1																			
1	2	1																			
2	1	2																			
1	2	1																			
<b>H3</b> <table border="1" style="display: inline-table; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>8</td><td>0</td></tr> <tr><td>2</td><td>1</td><td>2</td></tr> <tr><td>0</td><td>2</td><td>0</td></tr> </table>		0	8	0	2	1	2	0	2	0	$G(X_0) = \frac{\sum_{i=0}^{i=8} f(X_i)}{\sum_{i=0}^{i=8} V(X_i)}$										
0	8	0																			
2	1	2																			
0	2	0																			

---

---

b) **Filtre passe haut :**

Un filtre linéaire ou les coefficients du masque peuvent être positifs ou négatifs et leur somme égale à 0. Ce type de filtre est utilisé pour la détection et le renforcement des contours [65].

- *Filtre de Prewitt :*

1	1	1
1	-2	1
-1	-1	-1

- *Filtre de Sobel :*

	<table border="1"><tr><td>-1</td><td>-2</td><td>-1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>2</td><td>1</td></tr></table>	-1	-2	-1	0	0	0	1	2	1	
-1	-2	-1									
0	0	0									
1	2	1									
$S_x$		$S_y$									
		<table border="1"><tr><td>-1</td><td>0</td><td>1</td></tr><tr><td>-2</td><td>0</td><td>2</td></tr><tr><td>-1</td><td>0</td><td>1</td></tr></table>	-1	0	1	-2	0	2	-1	0	1
-1	0	1									
-2	0	2									
-1	0	1									

• *Filtre de Robert :*

1	0	0
0	0	0
0	0	-1

• *Filtre Laplacien :*

0	-1	0
-1	4	-1
0	-1	0

Les deux filtres de Sobel et Laplacien sont utilisés pour la détection et le renforcement des contours.

#### II.4.2.2 Filtrage non linéaire:

Affecte au pixel considéré une valeur particulière de son voisinage ; au lieu de la moyenne pondérée [80].

a) **Filtre Médian :**

Consiste à trier les pixels d'une fenêtre placée, sur l'image sujet de au filtrage (par ordre croissant), puis remplacer le pixel considéré par la valeur située au milieu de la liste triée.

b) **Filtre Maximum :** de la même manière que le filtre Médian, trier les pixels et remplacer le pixel du centre par le maximum.

c) **Filtre Minimum** : le même principe que les filtres précédents, en remplaçant le pixel du centre par le minimum des valeurs.

d) **Filtre par élimination** :

Il se base sur la formule suivante : Si  $\left| \frac{\sum_{i=0}^{i=8} X_i}{N} - X_0 \right| > S$  alors  $X_0 = \frac{\sum_{i=0}^{i=8} X_i}{N}$

e) **Filtre de gauss** :

Il est défini par l'équation suivante pour calculer les nouvelles valeurs de l'image améliorée :

$$Imag_{Améliorée} = Imag_{Originale} * G ;$$

$$Tel\ que : G(x,y) = \frac{1}{(2\sigma)^2} * Exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

## II.5 Amélioration et restauration des images :

Chaque fois qu'une image est convertie d'une forme à une autre, conçue, copiée ou affichée, sa qualité est plus ou moins affectée [35].

**II.5.1 Amélioration** : Consiste à la Diminution du bruit, le Renforcement du contraste, et le Rehaussement des contours.

### II.5.1.1 Diminution de bruit :

Le bruit est dû à des erreurs lors de la transformation de l'image, il se traduit généralement par des points blancs isolés et repartis au hasard dans l'image [18].

Les approches les plus utilisées pour la réduction du bruit sont les suivantes :

1. Détecter les points incriminés en comparant l'intensité de chaque pixel avec la moyenne des intensités de leurs huit voisins, si l'intensité du pixel s'écarte trop de l'intensité moyenne, on considère que l'on a détecté un parasite, pour l'éliminer on remplace sa valeur par la valeur moyenne, toute fois il est judicieux de noter que fixer un seuil est nécessaire à la détection des points parasites reste un problème à résoudre.
2. Usage d'un filtre médian ou morphologique.
3. Usage des filtres passe bas ou lisser les images.

---

---

### II.5.1.2 Renforcement du contraste :

Un contraste est dit 'faible', quand il y a une faible amplitude entre les valeurs des pixels .ceci peut être directement perceptible a partir de l'histogramme, qui représente le nombre de points dont l'intensité a une valeur donnée ; il donne donc, la fréquence d'apparition de chaque niveau de gris [18].

- ✓ L'Image est dite 'faiblement contrastée', si elle n'utilise pas toute la dynamique des 256 niveaux disponibles.
- ✓ Un Histogramme concentré dans un intervalle de plus bas niveau de gris, indique une image 'très sombre'
- ✓ Un Histogramme concentré dans un intervalle de plus haut niveau de gris, indique une image 'très claire'.

a) **L'histogramme de l'image** : Représente le nombre de points dont l'intensité a une valeur donnée ; il donne donc, la fréquence d'apparition de chaque niveau de gris.

b) **Les opérations d'éclatement et d'égalisation d'histogramme** :

Les solutions proposées sont les suivantes :

▪ **L'égalisation de l'histogramme** : Pour des raisons de lisibilité des régions sombres et claires ; cette technique sert a transformer l'image de telle façon que l'histogramme de l'image résultante, suit une forme demandée « formes gaussiennes » [35]

▪ **Éclatement** : Il s'agit de transformer un niveau de gris 'u', en un niveau 'v'. Supposons qu'on opère sur l'intervalle [S1, S2], l'éclatement étale l'intervalle [S1, S2] et compresse les intervalles [0, S1], [S2, 256] [35].

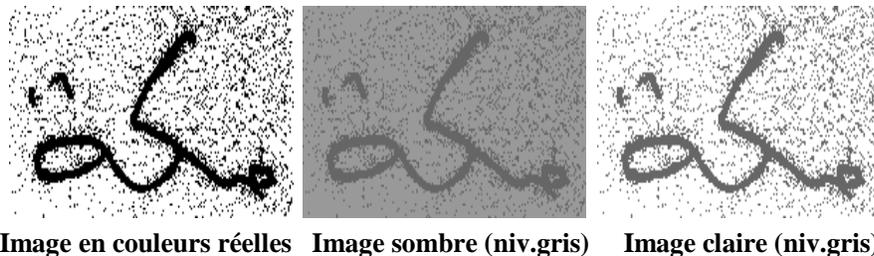
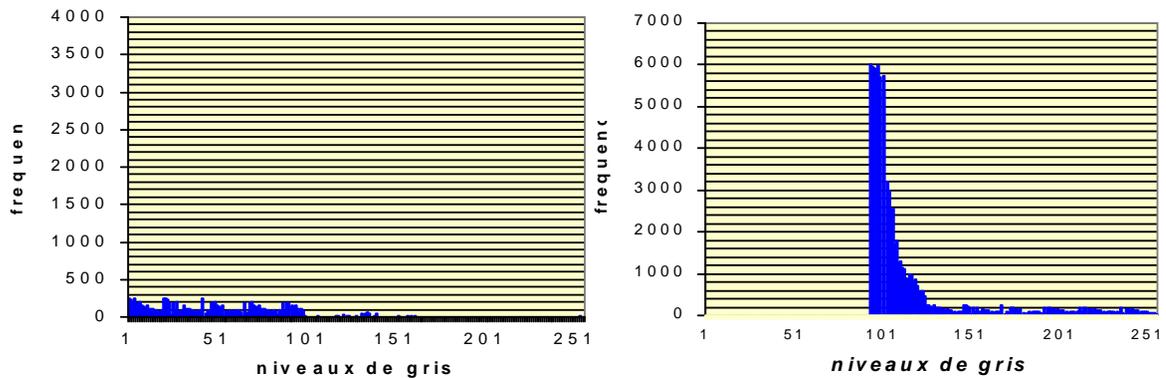


Figure N°II-10 : Effet de contraste sur les images à niveau de



**Histogramme de l'image sombre**

**Histogramme de l'image claire**

**Figure N°II-11: Fréquence des niveaux de gris pour les images sombres et claire**

### II.5.1.3 Rehaussement des contours :

Il sert à améliorer l'apparence de l'image et non détecter explicitement les contours [35].

- 1) Usage des filtres linéaires passe haut.
- 2) Différenciation statique : l'intensité de chaque pixel est divisée par l'écart type des intensités calculées dans le voisinage de ce pixel, puis elle est normalisée.

### II.5.2 Restauration d'image:

Consiste à réparer l'image, qui a été acquise dans des conditions défavorables ; en appliquant l'inverse du phénomène de dégradation, ce qui nécessite une certaine connaissance de ce dernier. Enfin, notons qu'il est parfois nécessaire, d'appliquer une combinaison des méthodes citées, ci-dessus pour remédier à la dégradation de l'image [35].

### II.6 Détection de contours :

La détection de contours est appliquée a une image afin de prélever des zones, régions homogènes ou tout simplement des objets. Cette opération est souvent la première étape dans le traitement des images, en particulier dans l'imagerie statique. L'extraction de contour est un sujet très important en traitement d'images [65] .

---

---

La grande variabilité des données, l'absence d'une définition précise, sont à l'origine de la coexistence de plusieurs méthodes. Elles sont groupées en méthodes directes qui visent à obtenir les contours directement et des méthodes indirectes qui déterminent d'abord les régions et ensuite déduisent les contours comme frontières directes [18].

### II.6.1 Détecteurs Autonomes :

Le détecteur autonome est un module, qui n'est pas influencé par les autres modules. Ses avantages sont la généralité et la portabilité [35].

### II.6.2 Détecteurs Contextuels :

Dans ce cas, le détecteur est un module guidé par les résultats des autres modules, sur le contour de l'image. Par conséquent, le détecteur a une connaissance du contour contenu dans l'image et ne pouvant s'exécuter que dans un contexte précis [80].

### II.6.3 Organisation des détecteurs :

Le détecteur de contours est structuré de la manière suivante [80] :

- ✓ **Un traitement de lissage :** Se résumant à une réduction du bruit, par une convolution de l'image avec des filtres linéaires passe bas.
- ✓ **Une différenciation :** Consistant à détecter les points où l'image présente une variation forte d'amplitude (points de contours), via des opérateurs du premier et du deuxième ordre. Ces opérateurs ne sont autres que la première et la seconde dérivée de la fonction image.
- ✓ **Un seuillage :** En exécutant les opérateurs cités ci-dessus on estime le module  $G(x,y)$ , qui est le module de la dérivée à chaque position du plan de l'image. Il s'agit donc d'utiliser cet attribut pour trouver les points ayant un  $G(x,y)$  supérieur à un certain seuil fixé, sont considérés comme des 'points de contour'. Le seuil fixé peut être local ou global :

*Seuil global :* un seuil unique est appliqué à toute l'image

*Seuil local :* Il s'agit de désigner plusieurs seuils, chacun relatif à une région limitée de l'image, afin de prendre en considération les caractéristiques locales de l'image.

- 
- 
- ✓ **Une correction** : Son rôle est supprimer les faux points de contours, par fermeture des trous dans les contours résultant.

## **II.7 Conclusion:**

Pour permettre une reconnaissance fiable un prétraitement est al ors nécessaire pour améliorer la qualité de l'image obtenue et éviter les erreurs. L'image est donc filtrée et améliorée, pour augmenter l'efficacité du traitement, et les caractéristiques locales et globales des formes extraites (caractères) sont générale ment utilisées. Aussi la nécessité d'un squelette binaire de l'image filtrée et normalisée. Dans la suite de ce mémoire nous nous intéresserons à la phase de segmentation et l'extraction des caractéristiques.

---

---

## **Chapitre III : La segmentation et l'Extraction des primitives**

### **III.1 Introduction :**

La segmentation est le processus qui consiste à extraire (isoler) les caractères d'un mot, pour qu'il soit prêt à la phase de reconnaissance. La segmentation des caractères est l'étape la plus cruciale et la plus difficile dans l'OCR Arabe. En outre, elle est la plus importante, la plus sensible et a le plus d'influence sur les résultats et la fiabilité du système. Une mauvaise segmentation conduit à une mauvaise reconnaissance ou bien carrément à un rejet.

Cette phase peut être divisée en trois étapes : la segmentation du texte en lignes, la segmentation des lignes de texte en mots et pseudo mots (PAW=Peace of Arabic Word), et la segmentation des mots en caractères isolés [9].

La phase d'extraction des caractéristiques pertinentes et significatives (primitives), qui est une étape primordiale dans ce type de système qui se trouve directement après la segmentation, et à pour but la réduction de l'information à traiter pour minimiser les calculs et le temps [11].

### **III.2 La phase de Segmentation :**

#### **III.2.1 Introduction :**

La reconnaissance du mot (ou PAW) nécessite d'abord son extraction de la page, ceci suppose une décomposition de la page au préalable, qui consiste à retrouver la structure physique du document en délimitant les différentes parties homogènes (texte, graphe, photographie . . .).

La segmentation est une opération appliquée à l'image qui consiste à subdiviser une scène réelle, en parties constituantes ou objets, en projetant une scène réelle sur un plan. Elle est la première opération réalisée dans « la reconnaissance des formes ». Il faut donc, disposer d'un certain nombre d'attributs, représentatifs des régions que l'on cherche à extraire, pour procéder à la classification individuelle des points [9].

---

---

### III.2.2 Les techniques de la segmentation :

Il existe deux techniques permettant de mis en œuvre de la segmentation. La première, connue sous le nom de segmentation implicite, et la deuxième c'est la segmentation explicite.

#### III.2.2.1 *Segmentation implicite* :

Les méthodes de segmentation implicite s'inspirent des approches utilisées dans le domaine de la parole, ou le signal est divisé en intervalles de temps réguliers, et procèdent a une sur-segmentation importante de l'image du mot a pas fixe (un ou quelques pixels). Cela permet d'assurer un taux de présence important des points de liaison entre lettres parmi les proposés. La segmentation s'effectuée pendant la reconnaissance et est guidée par cette dernière. Le système recherche dans l'image, des composantes ou des groupements de graphèmes qui correspondent à ses classes de lettres [20]. Classiquement, il peut le faire de deux manières :

- ✓ *soit par fenêtrage* : le principe est d'utiliser une fenêtre mobile de largeur variable (qui n'est pas facile a déterminer) pour trouver des séquences de points de segmentation potentiels qui seront confirmés ou non par la reconnaissance de caractères. Elle nécessite deux étapes : la génération d'hypothèses de segmentation (séquences de points obtenus par le fenêtrage) ; la deuxième est le choix de la meilleure hypothèse de la reconnaissance (validation).
- ✓ *soit par recherche de primitives* : il s'agit de détecter les combinaisons de primitives qui donneront la meilleure reconnaissance.

#### III.2.2.2 *Segmentation explicite* :

Cette approche, souvent appelée dissection, est antérieure a la reconnaissance et n'est pas remise en cause pendant la phase de reconnaissance. Les hypothèses des caractères sont déterminées à partir des informations de bas niveau présentes sur l'image. Ces hypothèses sont définitives, et doivent être d'une grande fiabilité car la moindre erreur de segmentation remet en cause la totalité des traitements ultérieurs. Les approches de segmentation explicite, s'appuient sur une analyse morphologique du mot manuscrit pour localiser des points de segmentation potentiels. Elles sont particulièrement adaptées à l'analyse de la représentation bidimensionnelle et donc plus souvent utilisées dans les

---

---

systèmes de reconnaissance hors-ligne de mots. Certaines méthodes de segmentation explicite sont basées sur : analyse par morphologiques mathématiques, exploitent les concepts de régularité et singularité du tracé, analyse des contours supérieurs/inferieurs du mot. Les points de segmentation potentiels détectés sont confirmés à l'aide de diverses heuristiques [20].

### III.2.3 Critères de segmentation :

La segmentation divise l'image en un ensemble de régions, en se basant sur les propriétés intrinsèques de la région, qui sont les suivantes :

- ✓ Le niveau de gris ou couleur dans le cas de l'image en couleur.
- ✓ La texture, qui est des motifs répétitifs apparaissant sur la surface d'un objet.
- ✓ Le décalage ou mouvement détecté par une différence entre deux images.

### III.2.4 Les étapes de segmentation :

#### III.2.4.1 Segmentation du texte en lignes :

Les méthodes de traitement de l'arabe utilisent souvent la projection horizontale pour extraire les lignes. Cependant la présence des points diacritiques complique cette extraction et conduit parfois à la confusion des lignes [38]. Ce problème a lieu quand l'interligne est calculé par une simple moyenne des différents interlignes (figure II-9). Pour remédier à ce problème, certains auteurs tels que [47] identifient d'abord les différentes lignes d'écriture, ensuite regroupent les blocs de texte d'après leur proximité par rapport aux lignes d'écriture déjà localisées.

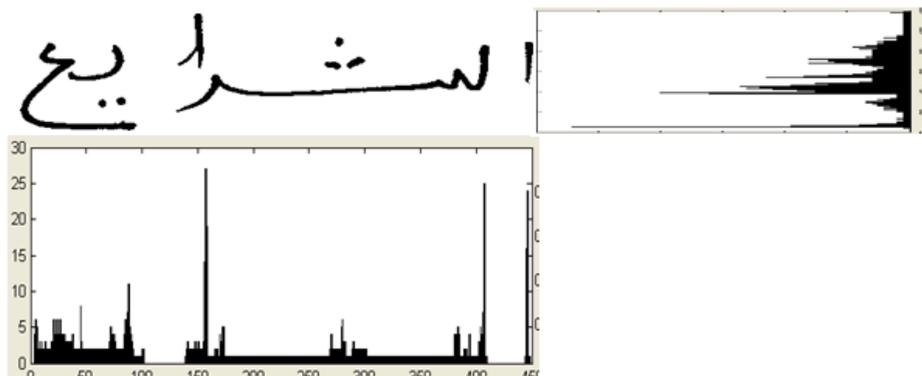


Figure III-1-Exemple d'histogramme horizontal et vertical d'une ligne de texte.

---

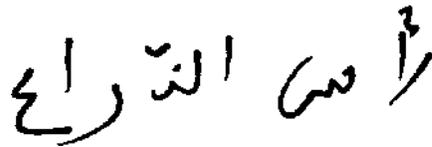
---

Comme pour le latin, une fusion des lignes est aussi possible à cause des hampes et des jambages. En cas de fusion, une méthode empirique de correction consiste à localiser d'abord la ligne qui contient le maximum de pixels noirs [20]. Les parties au dessus et en dessous de cette ligne sont ensuite analysées en se basant sur les densités de pixels noirs des différentes lignes.

Si la fusion a eu lieu par exemple dans la partie supérieure, la ligne ayant le minimum de densité de pixels dans cette partie, correspond à la frontière entre les lignes fusionnées.

#### III.2.4.2 Segmentation en Pseudo mot ( PAW :Peace of Arabic Word)

Elle est réalisée en déterminant l'histogramme des projections verticales des différentes lignes de texte [39]. Cependant, cette méthode n'est pas efficace dans le cas où les PAWs se chevauchent verticalement. Dans ce cas, d'autres techniques sont utilisées telles que la détermination du contour [41], du squelette [20], ou encore des composantes connexes [63]. Le choix de la technique est souvent guidé par la méthode d'analyse [20] [39].



*Figure III-2- Exemple de chevauchement de PAWs respectivement entre «رأس» et «الذراع».*

#### III.2.4.3 Segmentation de la ligne de texte en mots

En AOCR, la segmentation est souvent réservée à l'extraction des PAWs, le mot est plutôt considéré dans la phase de post-traitement (si elle est prévue) pour valider les résultats trouvés ou corriger les erreurs de reconnaissance. Par ailleurs afin d'éviter le problème d'une segmentation erronée en mots, certains auteurs introduisent dans leurs systèmes, un seul mot à la fois [20] [39].

---

---

#### III.2.4.4 Segmentation du mot en caractères

La segmentation en caractères (ou en graphèmes) constitue le problème le plus ardu lié à la reconnaissance de l'écriture arabe. Pour cette raison, ce sujet a fait l'objet principal de notre recherche. Les difficultés rencontrées à ce niveau sont du même type que celles affrontées lors de la reconnaissance du latin manuscrit (cursif), mais souvent plus complexes à cause de la diversité des formes du caractère arabe, de la courte liaison qui existe entre les caractères successifs, de l'allongement des ligatures horizontales et de la présence des ligatures verticales [39] [4].

#### III.2.5 Les difficultés de la segmentation :

La segmentation intervient après la détermination par composantes connexes des rectangles à l'intérieur desquels sont inscrits les caractères [11].

Ces rectangles sont de différentes largeurs, depuis celui qui entoure le i bâton jusqu'à ceux qui renferment des groupes de lettres plus ou moins importants.

La première difficulté consiste à déterminer si le rectangle circonscrit contient un ou plusieurs caractères.

La seconde difficulté c'est due à la variabilité de la chasse, car les espaces blancs qui séparent les mots ne correspondent pas forcément à la chasse.

La troisième difficulté est de déterminer à l'intérieur de ces rectangles les endroits où les caractères sont joints, la ligne qui les sépare et enfin l'algorithme qui assurera cette séparation.

Une dernière difficulté vient s'ajouter : il est nécessaire de s'assurer que la segmentation effectuée ne risque pas de couper certains caractères.

#### III.2.6 Conclusion :

En résumé, nous avons présenté des méthodes de segmentation d'une ligne de texte /mots/caractères en trois étapes: l'extraction de la ligne de texte, puis la segmentation en mots, et ensuite la segmentation des mots en caractères. L'extraction de la ligne de texte par projection horizontale (après un prétraitement, redressement et d'orientation). La

---

---

segmentation de la ligne de texte en mots (et pseudo mots PAW) est faite par projection verticale pour séparer les mots (paw) les un des autres, et on peut introduire une technique d'extraction (séparation) des composantes connexes pour le cas de chevauchement.

La troisième étape consiste à segmenter les mots (paw) en caractères, par la combinaison des projections verticale, horizontale et les caractéristiques structurelles et géométriques des caractères. Finalement, les graphiques mal segmentés ont un aspect visuel texturé proche de celle de la calligraphie .

### **III.3 L'extraction des primitives (informations pertinentes) :**

L'extraction des primitives ou caractéristiques pertinentes, a pour but de réduire la quantité de l'information à manipuler dans l'image. Ceci, revient à sélectionner les primitives les plus significatives [04]. Les objectifs et les étapes de l'extraction des primitives sont discutés dans cette partie.

#### **III.3.1 Les approches de l'extraction des primitives**

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 [36] [20].

☞ *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 à 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.

---

---

### III.3.2 Les catégories de primitives

Nous avons distingué quatre catégories principales de primitives: les primitives topologiques, statistiques, structurelles et globales [11].

#### *Les primitives topologiques ou métriques*

Le terme métrique désigne la mesure d'une distance. La topologie est "l'étude des propriétés de l'espace (et des ensembles) du seul point de vue qualitatif".

Concrètement, la topologie consiste, à l'aide de sondes appliquées directement sur l'image brute, à effectuer par exemple sur l'échantillon les mesures et les tests suivants :

- ✓ Compter dans une forme le nombre de trous,
- ✓ Évaluer les concavités,
- ✓ Mesurer des pentes et autres paramètres de courbures et évaluer des orientations principales,
- ✓ mesurer la longueur et l'épaisseur des traits,
- ✓ détecter les croisements et les jonctions des traits,
- ✓ mesurer les surfaces, les périmètres,
- ✓ déterminer le rectangle délimitant l'échantillon, ou le polygone convexe,
- ✓ évaluer le rapport d'élongation (ou allongement) longueur/largeur, ...
- ✓ rendre compte de la disposition relative de ces primitives.

#### *Les primitives structurelles*

A la différence des primitives topologiques, les primitives structurelles sont généralement extraites non pas de l'image brute, mais à partir d'une représentation de la forme par le squelette ou par le contour. Ainsi, on ne parle plus de trous, mais de boucles ou de cycles dans une représentation filiforme du caractère. Cependant, pour le reste, les primitives structurelles correspondent à peu près aux primitives topologiques, il s'agit principalement :

- ✓ des segments de droite,
- ✓ des arcs, boucles et concavités, des pentes,
- ✓ des angularités, points extremum et points terminaux, jonctions et croisements.

---

---

### *Les primitives statistiques*

Elles véhiculent une information qui est distribuée sur toute l'image. L'histogramme, qui représente le nombre de pixels sur chaque ligne ou colonne de l'image, en est un exemple classique et simple à calculer. L'histogramme directionnel est plus long à calculer car il nécessite par exemple l'utilisation d'un algorithme permettant de compter le nombre de pixels contenus sur une ligne de direction quelconque de l'image. L'histogramme des transitions, comme l'indique son nom, ne retient que le nombre des transitions 0-1 et 1-0.

Par l'intersection de la forme avec un réseau de droites pour la reconnaissance des chiffres manuscrits, les histogrammes de transition sont construits seulement avec un échantillon de quelques droites au lieu de la totalité de celles-ci, ce qui permet une réduction a priori des données caractéristiques [11].

Pour s'affranchir du choix arbitraire de ces droites (espacement et orientation des droites du réseau) qui doit nécessairement être fixe entre l'apprentissage et la reconnaissance, on peut faire intervenir les probabilités en calculant la fréquence des intersections de la forme avec une série de droites aléatoires [11].

On utilise aussi une autre primitive statistique basée sur un moyennage des pixels situés à l'intérieur d'un masque rectangulaire. La construction d'une matrice de masque recouvrant la totalité de la forme permet une représentation statistique à partir d'un nombre très réduit de valeurs correspondant à chaque masque.

### *Les primitives globales*

Elles sont naturellement basées sur une transformation globale de l'image. La transformation consiste à convertir la représentation en pixels en une représentation plus abstraite pour réduire la dimension des caractères, tout en conservant le maximum d'informations sur la forme à reconnaître. Par exemple : la transformée de Hough, la transformée de Fourier, et les moments invariants [20].

### *Les moments invariants :*

Dans notre système la technique des moments invariants est utilisée pour construire un espace caractéristique. L'utilisation des combinaisons non linéaires des moments

géométriques nous a permis d'avoir un ensemble des moments invariants, qui ont la propriété d'être invariants par rapport à la translation d'image, à l'échelle, et à la rotation [2]. Nous avons utilisé cette technique dans notre système, bien sûr avec autres primitives. Les moments centraux qui sont invariants ont la définition mathématique suivante :

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

Pour que cette formule soit appliquée aux images (cas discret), la formule est transformée comme suit :

$$M_{pq} = \sum_{x=0}^m \sum_{y=0}^n (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

Tout ce qui concerne les moments invariants est détaillé dans l'annexe B.

### III.3.3. Les étapes de l'extraction des primitives

Il peut y avoir plusieurs étapes intermédiaires pour parvenir à l'objectif fixé. La première étape importante de la reconnaissance est le changement de représentation de l'information. A partir de l'image numérique brute, qui est la représentation originelle, on relève quatre autres représentations possibles de l'information [5] [7], qui sont :

1. L'extraction des composantes connexes
2. L'extraction du contour
3. La détermination du squelette (au sens large du terme)
4. Les transformations mathématiques qui opèrent globalement sur l'image.

C'est à partir d'une de ces représentations qu'est ensuite réalisée l'étape d'extraction des primitives proprement dite [36].

On le voit, le vocabulaire lié à l'extraction des primitives est riche et varié. Aussi, afin d'ordonner toutes les étapes suivant la progression normale du traitement de l'information et en les situant les unes par rapport aux autres, nous avons regroupé sur un schéma l'ensemble de ces opérations suivant deux axes indépendants qui représentent la **segmentation** et la **réduction** de l'information. La réduction vise à l'élimination de

---

---

l'information redondante ou du bruit. Son axe va dans le sens de la reconnaissance (l'identification est la réduction ultime de l'information), ce qui n'est pas le cas de la segmentation [2] [11].

### **III.3.4 La problématique de l'extraction de l'information**

La problématique de cette étape a pour origine le risque de perte d'information signifiante pour la reconnaissance. La minimisation de ce risque est conditionnée par deux dilemmes liés chacun à un paradoxe de causalité. Il s'agit du dilemme de la réduction et du dilemme de la segmentation [11].

Nous allons développer dans cette partie la nature de ces dilemmes en présentant les différentes origines de perte d'information lors du processus de la reconnaissance de l'écriture manuscrite.

La première étape du processus est l'acquisition de l'information. Il est évident que l'on ne peut pas extraire une information qui n'est pas présente dans le tracé. C'est pourquoi, avant de traiter de la perte d'information pendant la phase d'acquisition, nous considérerons la perte d'information pendant la phase de production de l'écriture [2].

#### ***- La perte d'information pendant la phase de production de l'écriture***

Qui a lieu pendant l'écriture elle-même, est due au facteur humain. Cette question, qui s'écarte un peu de notre sujet, a cependant des conséquences déterminantes sur la reconnaissance.

Nous pensons que l'origine de la perte d'information est due à la projection dans un espace de dimension inférieure, du concept des mots lors de leur tracé linéaire dans le plan.

Le caractère subjectif de l'écriture tient au fait que le lecteur étant aussi le scripteur, il peut ne pas s'apercevoir que son écriture est illisible !

L'auteur a en outre la liberté de s'appliquer à ne bien tracer que les lettres permettant de distinguer les mots. Cette attitude a une conséquence importante sur la stratégie de lecture à adopter pour la reconnaissance. En effet, elle implique d'avoir la connaissance du vocabulaire parmi lequel l'auteur a estimé pouvoir distinguer les mots.

---

---

Cette perte d'information peut se manifester par une grande variation dans le tracé de l'écriture, par rapport au modèle scolaire ou à un tracé moyen .

La limite à considérer pour ce type de perte d'information est que si un humain ne peut pas lire un texte, un ordinateur ne le pourra probablement pas non plus. Cela fixe une borne inférieure à la qualité d'écriture admissible pour un système de reconnaissance.

#### ***- La perte d'information pendant la phase d'acquisition***

L'objectif de cette étape est d'acquérir le maximum d'informations, afin d'obtenir des images numériques les plus précises. La perte d'information pendant l'acquisition est liée le plus souvent à la faiblesse des capteurs (scanner ou éventuellement caméra), ou aux conditions d'acquisition [6].

A ce niveau du processus, il reste à déterminer si l'information significative est accessible ou non parmi toute l'information acquise.

#### ***- La perte d'information pendant la phase de réduction***

Elle consiste à extraire l'information significative qui est noyée dans le bruit . Dans le cas de la classification, l'objectif est d'éliminer davantage de bruit que d'information afin de faciliter la discrimination.

Dans le cas de l'approximation, l'objectif est de modéliser en minimisant la perte d'information. Mais, plus la modélisation est précise, plus la procédure de reconnaissance est complexe et plus la représentation est sensible aux variations de tracé (fioritures non significatives pour la reconnaissance). Comment s'affranchir de cette forte variabilité du tracé grâce à l'approximation, sans perdre l'information significative nécessaire à la reconnaissance ?

Ce dilemme de la réduction repose sur le paradoxe que l'on ne peut savoir qu'une information est significative ou non tant que la forme n'est pas reconnue. Il en résulte une certaine subjectivité de l'étape de réduction, ce qui implique un risque de perte d'information.

Le changement de représentation opéré grâce aux transformées mathématiques, en particulier celles de Fourier et Hough, peut apporter une aide précieuse dans ce cas.

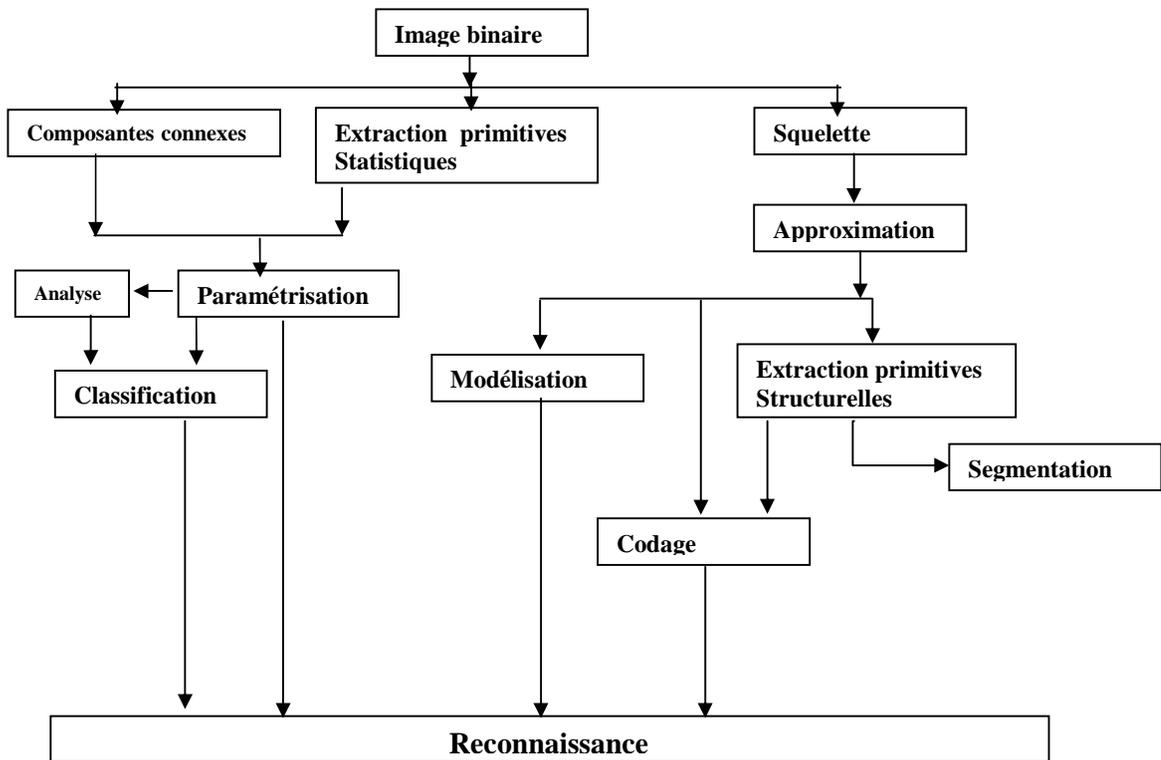
---

---

**- La perte d'information pendant la phase de segmentation**

Une autre perte d'information se produit également au cours de la phase de segmentation. A l'instar du dilemme de la réduction, le problème de la segmentation nous ramène une seconde fois au paradoxe de causalité.

Pour pouvoir segmenter le mot en lettres afin de les reconnaître isolément, il est nécessaire au préalable de les localiser, or comment localiser ces lettres sans les avoir reconnues auparavant?



**Figure III.3 Les différentes étapes de l'extraction des primitives et de segmentation (Segmentation et réduction de l'information)**

L'ensemble des échantillons d'écriture traités dans notre travail est constitué par des images numériques de type binaire. Or, dans le cas où les échantillons sont numérisés avec une caméra CCD ou un scanner multiniveau, les images obtenues sont de type niveaux de gris, ce qui nécessite souvent une étape préliminaire de binarisation ce qui n'est pas sans poser le problème du choix du seuil lorsque le fond de l'image a un niveau uniforme. Certaines études [7], [12], [14] montrent alors l'intérêt de pratiquer l'étape d'extraction des primitives directement sur l'image en niveaux de gris.

---

---

En revanche, lorsque le fond de l'image est uniforme, la binarisation n'entraîne pas une perte d'information significative, à condition que la résolution soit suffisante.

D'autre part, nous devons mentionner un problème entraînant une étape supplémentaire de segmentation, que nous n'avons pas fait figurer dans notre schéma, lorsqu'une information non signifiante pour la reconnaissance est superposée au texte à décrypter. Cette information peut être par exemple les barres obliques dans le cas des chèques ou encore un bruit quelconque. Le traitement consiste alors à extraire l'information utile du reste de l'image, [14] [15].

### **III.4 Conclusion :**

L'étape de l'extraction des primitives est une étape très importante dans n'importe quel système de reconnaissance. Les performances des classifieurs dépendent directement de la sélection et l'extraction des primitives. Le grand avantage de ces méthodes est l'élimination de la redondance des données, et la représentation de l'image par un ensemble de valeurs numériques (dont le but est la compression des données sans perte). Ces primitives sont ensuite présentées aux algorithmes de classification.

Dans le dernier chapitre nous présenterons les techniques proposées dans notre travail pour la segmentation et l'extraction des primitives.

---

---

## Chapitre IV : Classification et reconnaissance

### IV.1. Les Séparateurs à Vaste Marge (SVM)

#### IV.1.1. Introduction

L'origine des *Support Vector Machines*, traduit par Séparateur à Vaste Marge (SVM) (ou bien les machines à vecteurs de support), remonte à 1975 lorsque Vapnik et Chervonenkis proposèrent le principe du risque structurel et la dimension VC pour caractériser la capacité d'une machine d'apprentissage. A cette époque, ce principe n'a pas trouvé place et il n'existait pas encore un modèle de classification solidement appréhendé pour être utilisable. Il a fallu attendre jusqu'à l'an 1982 pour que Vapnik propose un premier classificateur basé sur la minimisation du risque structurel baptisé SVM. Ce modèle était toutefois linéaire et l'on ne connaissait pas encore le moyen d'induire des frontières de décision non linéaires [1]. En 1992, Boser et al. proposent d'introduire des noyaux non-linéaires pour étendre le SVM au cas non-linéaire. En 1995, Cortes et al. proposent une version régularisée du SVM qui tolère les erreurs d'apprentissage tout en les pénalisant.

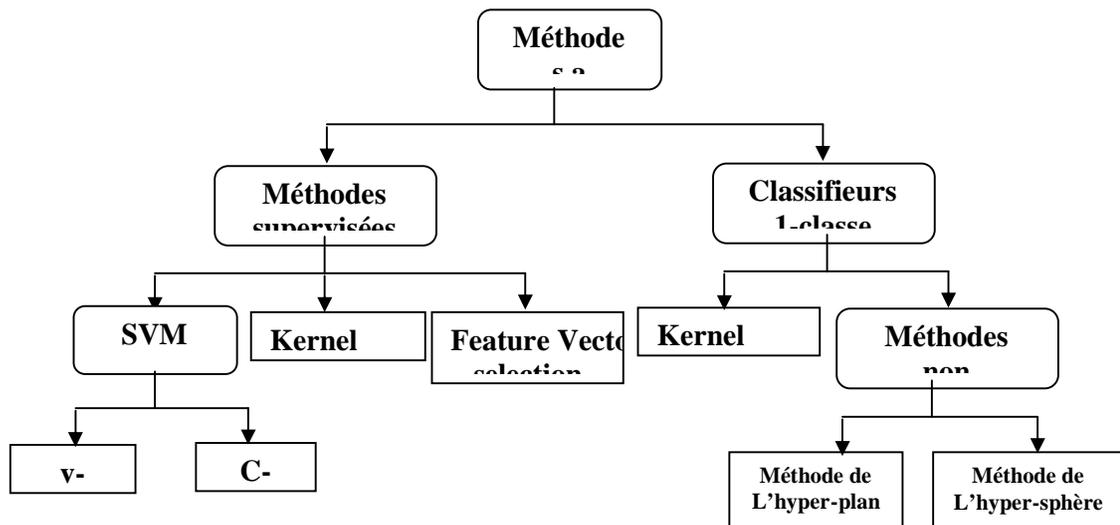


Figure N° IV-01 : Arbre de classification des méthodes d'apprentissage à

Ils ont mis en évidence la ressemblance entre le SVM et la théorie de régularisation, et ils ont aussi démontré qu'associer un noyau particulier à un SVM revient à considérer une pénalisation différente de l'erreur d'apprentissage en maximisant la marge.

---

---

Ce qui nous permet de dire que la maximisation de la marge dans l'espace augmenté est une forme de régularisation de l'apprentissage [78].

Dès lors, le SVM permet de répondre à deux problèmes centraux de la théorie de l'apprentissage statistique que sont : [1]

- ✓ le contrôle de la capacité du classifieur,
- ✓ le sur-apprentissage des données.

Cette approche découle directement des travaux de Vapnik en théorie de l'apprentissage à partir de 1995. Elle s'est focalisée sur les propriétés de généralisation (ou prévision) d'un modèle en contrôlant sa complexité. Le principe fondateur des SVM est justement d'intégrer à l'estimation le contrôle de la complexité c'est-à-dire le nombre de paramètres qui est associé dans ce cas au nombre de vecteurs supports. L'autre idée directrice de Vapnik dans ce développement, est d'éviter de substituer à l'objectif initial : la discrimination, un ou des problèmes qui s'avèrent finalement plus complexes à résoudre comme par exemple l'estimation non paramétrique de la densité d'une loi multidimensionnelle en analyse discriminante. [19]

Le principe de base des SVM consiste de ramener le problème de la discrimination à celui, linéaire, de la recherche d'un hyperplan optimal [1]. Deux idées ou astuces permettent d'atteindre cet objectif :

- ☞ La première consiste à définir l'hyperplan comme solution d'un problème d'optimisation sous contraintes dont la fonction objectif ne s'exprime qu'à l'aide de produits scalaires entre vecteurs et dans lequel le nombre de contraintes "actives" ou vecteurs supports contrôle la complexité du modèle.
- ☞ Le passage à la recherche de surfaces séparatrices non linéaires est obtenu par l'introduction d'une fonction noyau (*kernel*) dans le produit scalaire induisant implicitement une transformation non linéaire des données vers un espace intermédiaire (*feature space*) de plus grande dimension. D'où l'appellation couramment rencontrée de machine à noyau ou *kernel machine*. Sur le plan théorique, la fonction noyau définit un espace hilbertien, dit auto-reproduisant et isométrique par la transformation non linéaire de l'espace initial et dans lequel est résolu le problème linéaire.

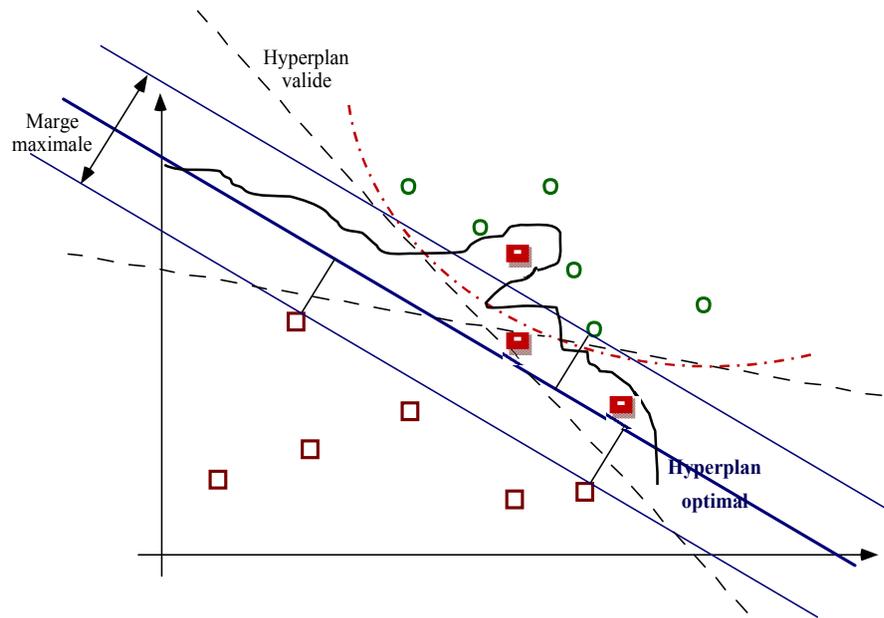
---

---

Depuis, les SVMs (le pluriel est utilisé pour désigner les différentes variantes du SVM) n'ont cessé de susciter l'intérêt de plusieurs communautés de chercheurs de différents domaines d'expertise. L'introduction de noyaux, spécifiquement adaptés à une problématique donnée, lui confère une grande flexibilité pour s'adapter à des situations très diverses. Par exemple, application des SVM à la reconnaissance de chiffres manuscrits isolés, la reconnaissance d'orateur, et la reconnaissance de visages [19].

Dans la plupart des cas, la performance du SVM égale ou dépasse celle de modèles classiques déjà établis, il s'avère un concurrent sérieux des algorithmes les plus performants (agrégation de modèles). L'estimation de densité de probabilité a été aussi explorée. D'autres auteurs ont aussi étudié l'apport de la connaissance a priori pour ce type de classificateurs [78].

A noter que, sur le plan algorithmique, ces algorithmes sont plus pénalisés par le nombre d'observations, c'est-à-dire le nombre de vecteurs supports potentiels, que par le nombre de variables. Néanmoins, des versions performantes des algorithmes permettent de prendre en compte des bases de données volumineuses dans des temps de calcul acceptables [19].



**Figure N° IV-2 Sous-ajustement linéaire et sur-ajustement local (proches voisins) d'un modèle quadratique.**

---

Les livres de référence sur ce sujet sont celui de [1], [86]. De nombreuses introductions et présentations des SVM sont accessibles sur le web comme par exemple: [www.kernel-machines.org](http://www.kernel-machines.org).

### IV.1.2. Principes

#### IV.1.2.1. Problème

Comme dans toute situation d'apprentissage, on considère une variable  $Y$  à prédire mais qui, pour simplifier cette introduction élémentaire, est supposée dichotomique à valeurs dans  $\{-1,1\}$ .

Soit  $X = X^1, \dots, X^p$  les variables explicatives ou prédictives et  $\phi(x)$  un modèle pour  $Y$ , fonction de  $x = \{x^1, \dots, x^p\} \in \mathbb{R}^p$ . Plus généralement on peut simplement considérer la variable  $X$  à valeurs dans un ensemble  $F$ . On note

$$Z = \{(x_1, y_1), \dots, (x_n, y_n)\}$$

Un échantillon statistique de taille  $n$  et de loi  $F$  inconnue. L'objectif est donc de construire une estimation  $\hat{b}$  de  $b$ , fonction de  $F$  dans  $\{-1,1\}$ , de sorte que la probabilité :

$$P(\hat{\phi}(X) \neq Y) \text{ soit minimale.}$$

Dans ce cas ( $Y$  dichotomique), le problème se pose comme la recherche d'une frontière de décision dans l'espace  $F$  des valeurs de  $X$ . De façon classique, un compromis doit être trouvé entre la *complexité* de cette frontière, qui peut s'exprimer aussi comme sa capacité à *pulvériser* un nuage de points par la VC dimension, donc la capacité d'*ajustement* du modèle, et les qualités de *généralisation* ou prévision de ce modèle. Ce principe est illustré par la Figure N° IV-2.

#### IV.1.2.2. Marge

La démarche consiste à rechercher, plutôt qu'une fonction  $\phi^b$  à valeurs dans  $\{-1,1\}$ , une fonction réelle  $f$  dont le signe fournira la prévision :  $\hat{\phi} = \text{signe}(f)$

L'erreur s'exprime alors comme la quantité :  $P(\hat{\phi}(X) \neq Y) = P(Yf(X) \leq 0)$

De plus, la valeur absolue de cette quantité  $|Y f(X)|$  fournit une indication sur la confiance à accorder au résultat du classement [19].

On dit que  $Y f(X)$  est la *marge* de  $f$  en  $(X, Y)$ .

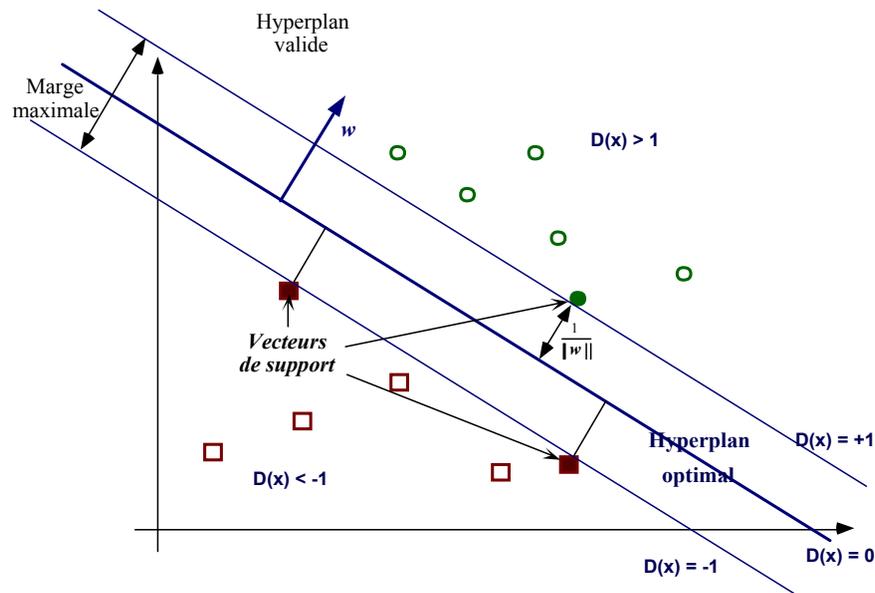


Figure N° IV-3 Recherche d'un hyperplan de séparation optimal au sens de la marge maximale.

#### IV.1.2.3. Espace intermédiaire

Une première étape consiste à transformer les valeurs de  $X$ , c'est-à-dire les objets de  $F$  par une fonction  $\Phi$  à valeurs dans un espace  $H$  intermédiaire (*feature space*) muni d'un *produit scalaire*. Cette transformation est fondamentale dans le principe des SVM, elle prend en compte l'éventuelle non linéarité du problème posé et le ramène à la résolution d'une séparation linéaire [19]. Ce point est détaillé dans une section ultérieure. Traitons tout d'abord le cas linéaire c'est-à-dire le cas où  $\Phi$  est la fonction identité.

#### IV.1.3. Séparation linéaire

##### IV.1.3.1. Hyperplan séparateur

La résolution d'un problème de séparation linéaire est illustrée par la figure IV.3. Dans le cas où la séparation est possible, parmi tous les hyperplans solutions pour la séparation des observations, on choisit celui qui se trouve le plus "loin" possible de tous les

exemples, on dit encore, de *marge* maximale [1]. Dans le cas linéaire, un hyperplan est défini à l'aide du produit scalaire de H par son équation :  $\langle w, x \rangle + b = 0$  Où  $w$  est un vecteur orthogonal au plan tandis que le signe de la fonction  $f(x) = \langle w, x \rangle + b$  Indique de quel côté se trouve le point  $x$  à prédire [19]. Plus précisément, un point est bien classé si et seulement si :  $y \cdot f(x) > 0$

Mais, comme le couple  $(w, b)$  qui caractérise le plan est défini à un coefficient multiplicatif près, on s'impose :  $y \cdot f(x) \geq 1$

Un plan  $(w, b)$  est un séparateur si :  $y_i f(x_i) \geq 1 \forall i \in \{1, \dots, n\}$

La distance d'un point  $x$  au plan  $(w, b)$  est donnée par :  $d(x) = \frac{\langle w, x \rangle + b}{\|w\|} = \frac{|f(x)|}{\|w\|}$

Et, dans ces conditions, la marge du plan a pour valeur  $2 / \|w\|^2$ .

Chercher le plan séparateur de marge maximale revient à résoudre le problème ci-dessous d'optimisation sous contraintes (problème primal):

$$\begin{cases} \min_w \frac{1}{2} \|w\|^2 \\ \text{avec } \forall i, \quad y_i (\langle w, x_i \rangle + b) \geq 1 \end{cases}$$

Le problème dual est obtenu en introduisant des multiplicateurs de Lagrange. La solution est fournie par un *point-selle*  $(w^*, b^*, \lambda^*)$  du lagrangien :

$$L(w, b, \lambda) = \frac{1}{2} \|w\|_2^2 - \sum_{i=1}^n \lambda_i [y_i (\langle w, x_i \rangle + b) - 1]$$

Ce point-selle vérifie en particulier les conditions :

$$\lambda_i^* [y_i (\langle w^*, x_i \rangle + b^*) - 1] = 0 \quad \forall i \in \{1, \dots, n\}$$

Les *vecteurs supports* sont les vecteurs  $x_i$  pour lesquels la contrainte est active, c'est-à-dire les plus proches du plan, et vérifiant donc :

$$y_i (\langle w^*, x_i \rangle + b^*) = 1$$

Les conditions d'annulation des dérivées partielles du lagrangien permettent d'écrire les relations que vérifie le plan optimal, avec les  $\lambda_i^*$  non nuls seulement pour les points supports :

$$w^* = \sum_{i=1}^n \lambda_i^* y_i x_i \text{ et } \sum_{i=1}^n \lambda_i^* y_i = 0$$

Ces contraintes d'égalité permettent d'exprimer la formule duale du lagrangien :

$$w(\lambda) = \sum_{i=1}^n \lambda_i - \frac{1}{2} \sum_{i,j=1}^n \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle$$

Pour trouver le point-selle, il suffit alors de maximiser  $W(\lambda)$  avec  $\lambda_i \geq 0$  pour tout  $i \in \{1, \dots, n\}$ . La résolution de ce problème d'optimisation quadratique de taille  $n$ , le nombre d'observations, fournit l'équation de l'hyperplan optimal :

$$\sum_{i=1}^n \lambda_i^* y_i \langle x, x_i \rangle + b^* \text{ avec } b^0 = -\frac{1}{2} [\langle w^*, sv_{classe+1} \rangle + \langle w^*, sv_{classe-1} \rangle]$$

Pour une nouvelle observation  $x$  non apprise présentée au modèle, il suffit de regarder le signe de l'expression :

$$f(x) = \sum_{i=1}^n \lambda_i^* y_i \langle x, x_i \rangle + b^*$$

Pour savoir dans quel demi-espace cette forme se trouve, et donc quelle classe il faut lui attribuer [19].

#### IV.1.3.2. Cas non séparable

Lorsque les observations ne sont pas séparables par un plan, il est nécessaire d'assouplir les contraintes par l'introduction de termes d'erreur  $\xi_i$  qui en contrôlent le dépassement [1]:

$$y_i (\langle w, x_i \rangle + b) \geq +1 - \xi_i \quad \forall i \in \{1, \dots, n\}$$

Le modèle attribue ainsi une réponse fautive à un vecteur  $x_i$  si le  $\xi_i$  correspondant est supérieur à 1. La somme de tous les  $\xi_i$  représente donc une borne du nombre d'erreurs.

Le problème de minimisation est réécrit en introduisant une pénalisation par le dépassement de la contrainte :

$$\begin{cases} \min_w \frac{1}{2} \|w\|^2 + \delta \sum_{i=1}^n \xi_i \\ \text{avec } \forall i, \quad y_i((w, x_i) + b) \geq +1 - \xi_i \end{cases}$$

**Remarques :**

- ✓ Le paramètre  $\delta$  contrôlant la pénalisation est à régler. Plus il est grand et plus cela revient à attribuer une forte importance à l'ajustement. Il est le paramètre qui ajuste le compromis entre bon ajustement et bonne généralisation.
- ✓ Le problème dans le cas non séparable se met sous la même forme duale que dans le cas séparable à une différence près : les coefficients  $\lambda_i$  sont tous bornés par la constante  $\delta$  de contrôle de la pénalisation.
- ✓ De nombreux algorithmes sont proposés pour résoudre ces problèmes d'optimisation quadratique. Certains, proposant une décomposition de l'ensemble d'apprentissage, sont plus particulièrement adaptés à prendre en compte un nombre important de contraintes lorsque  $n$ , le nombre d'observation, est grand.
- ✓ On montre par ailleurs que la recherche des hyperplans optimaux répond bien au problème de la "bonne" généralisation. On montre aussi que, si l'hyperplan optimal peut être construit à partir d'un petit nombre de vecteurs supports, par rapport à la taille de la base d'apprentissage, alors la capacité en généralisation du modèle sera grande, indépendamment de la taille de l'espace.
- ✓ Plus précisément, on montre que, si les  $X$  sont dans une boule de rayon  $R$ , l'ensemble des hyperplans de marge fixée  $\delta$  a une VC dimension bornée par

$$\frac{R^2}{\delta^2} \text{ avec } \|x\| \leq R$$

- ✓ L'erreur par validation croisée (*leave-one-out*) est bornée en moyenne par le nombre de vecteurs supports. Ces bornes d'erreur sont bien relativement prédictives mais néanmoins trop pessimistes pour être utiles en pratique.

---

---

#### IV.1.4. Séparation non linéaire

##### IV.1.4.1. Noyau

Revenons à la présentation initiale du problème. Les observations faites dans l'ensemble  $F$  (en général  $\mathbb{R}^p$ ) sont considérées comme étant transformées par une application non linéaire  $\Phi$  de  $F$  dans  $H$  muni d'un produit scalaire et de plus grande dimension. Le point important à remarquer, c'est que la formulation du problème de minimisation ainsi que celle de sa solution :  $f(x) = \sum_{i=1}^n \lambda_i^* y_i \langle x, x_i \rangle + b^*$

Ne fait intervenir les éléments  $x$  et  $x_0$  que par l'intermédiaire : *produits scalaires*:  $\langle x, x' \rangle$ . En conséquence, il n'est pas nécessaire d'expliciter la transformation  $\Phi$ , ce qui serait souvent impossible, à condition de savoir exprimer les produits scalaires dans  $H$  à l'aide d'une fonction  $k: F \times F \rightarrow \mathbb{R}$  symétrique appelée *noyau* de sorte que :

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle$$

Bien choisi, le noyau permet de matérialiser une notion de "proximité" adaptée au problème de discrimination et à sa structure de données [19].

##### *Exemple*

Prenons le cas trivial où  $x = (x_1, x_2)$  dans  $\mathbb{R}^2$  et  $\Phi(x) = (x_1^2, \sqrt{2} x_1 x_2, x_2^2)$  est explicite. Dans ce cas,  $H$  est de dimension 3 et le produit scalaire s'écrit :

$$\begin{aligned} \langle \Phi(x), \Phi(x') \rangle &= x_1^2 x_1'^2 + 2 x_1 x_2 x_1' x_2' + x_2^2 x_2'^2 \\ &= (x_1 x_1' + x_2 x_2')^2 \\ &= \langle x, x' \rangle^2 = k(x, x') \end{aligned}$$

Le calcul du produit scalaire dans  $H$  ne nécessite pas l'évaluation explicite de  $\Phi$ .

##### IV.1.4.2. Condition de Mercer

Une fonction  $k(., .)$  symétrique est un noyau si, pour tous les  $x_i$  possibles, la matrice de terme général  $k(x_i, x_j)$  est une matrice définie positive c'est-à-dire quelle définit une matrice de produit scalaire [1]. Dans ce cas, on montre qu'il existe un espace  $H$  et une fonction  $\Phi$  tels que :

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle$$

---

Malheureusement, cette condition théorique d'existence est difficile à vérifier et, de plus, elle ne donne aucune indication sur la construction de la fonction noyau ni sur la transformation  $\Phi$ . La pratique consiste à combiner des noyaux simples pour en obtenir des plus complexes (multidimensionnels) associés à la situation rencontrée [1].

#### IV.1.4.3. Exemples de noyaux

↳ Linéaire :  $k(x, x') = \langle x, x' \rangle$

↳ Polynomial :  $k(x, x') = (c + \langle x, x' \rangle)^d$

↳ Gaussien :  $k(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}}$

Beaucoup d'articles sont consacrés à la construction d'un noyau plus ou moins exotique et adapté à une problématique posée : reconnaissance de séquences, de caractères, l'analyse de textes... La grande flexibilité dans la définition des noyaux, permettant de définir une notion adaptée de similitude, confère beaucoup d'efficacité à cette approche à condition bien sûr de construire et tester le bon noyau. D'où apparaît encore l'importance de correctement évaluer des erreurs de prévision par exemple par validation croisée.

*Attention*, les SVM à noyaux RBF gaussiens, pour lesquels, soit on est dans le cas séparable, soit la pénalité attribuée aux erreurs est autorisée à prendre n'importe quelle valeur, ont une VC-dimension infinie.

#### IV.1.5. Extension de la formulation SVM

##### IV.1.5.1. Stratégies multi-classes

Lorsque plusieurs classes sont définies dans le problème initial, une stratégie multi-classes est définie. Il existe principalement deux approches en utilisant des classifieurs binaires [78] :

###### IV.1.5.1.1. un contre un

Pour  $n$  classes,  $n(n-1)/2$  classifieurs sont entraînés, chacun opposant une classe à une autre. Pour classer des nouvelles données, celles-ci sont testées pour chaque classifieur et la classe finale est attribuée par un vote [19].

###### IV.1.5.1.2. un contre tous

Pour  $n$  classes,  $n$  classifieurs sont entraînés. Les ensembles d'apprentissage sont constitués par une seule classe pour les exemples positifs, toutes les autres classes sont définies comme exemples négatifs. Les nouvelles données sont classées selon la prédiction la plus forte donnée parmi tous les classifieurs [19].

#### IV.1.5.2. One-class

Une variante de la formulation SVM propose ainsi de définir un classifieur dont l'hyperplan contient la frontière d'une seule classe. La figure IV-4 présente un exemple de frontière obtenue. L'intérêt de cette méthode permet ainsi de résoudre le difficile problème de la conception de la base d'apprentissage, notamment pour la définition des exemples négatifs. L'idée originale a ainsi permis d'appliquer les SVM à la détection de nouveautés en ligne. La figure IV-4 illustre ainsi l'hyperplan obtenu dans l'espace des caractéristiques. Cet hyperplan peut se résumer à la plus petite sphère englobant les données contenues dans la classe. Les points supports de la classe sont positionnés sur le contour de la sphère. La fonction de décision s'écrit alors :

$$f(x) = \text{sign}\left(\sum_{i=1}^n \alpha_i x_i^T x - p\right)$$

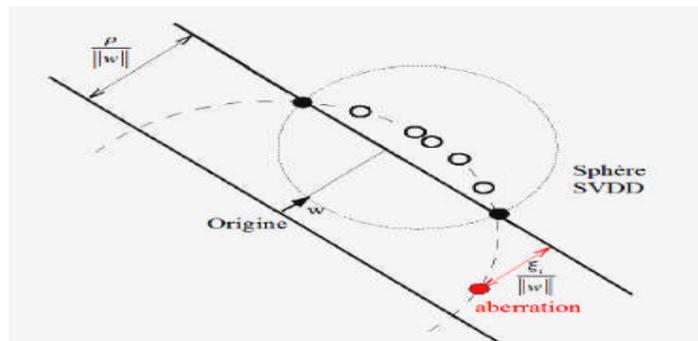


Figure N° IV-4 Schéma du classifieur une classe dans l'espace des caractéristiques.

Elle est obtenue en minimisant :

$$\begin{cases} \min_{w, w_0, \xi, p} \frac{1}{2} \|w\|^2 - p + \frac{1}{\nu} \sum_{i=1}^n \xi_i \\ \text{avec } x_i^T w \geq p - \xi_i, i = 1, n \\ \xi_i \geq 0, i = 1, n \end{cases}$$

La résolution aboutit au problème dual :

$$\begin{cases} \min_{w, w_0, \xi, p} \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j x_i^T x_j \\ \text{avec } 0 < \alpha_i < \frac{1}{\nu}, i = 1, n \\ \sum_{i=1}^n \alpha_i = 1, i = 1, n \end{cases}$$

---

---

#### IV.1.6. Conclusion

Le classifieur SVM *Support Vector Machine* est un classifieur de type machine à noyau et constitue l'état de l'art actuel en matière de discrimination. Concepts clés des SVM : maximiser la marge et exploiter l'astuce des noyaux.

↳ Une liste de réalisations de SVM se trouve à <http://www.kernel-machines.org/software.html>.

↳ IL existe plusieurs boîtes à outil Matlab pour les SVM.

Nous utiliserons cet algorithme pour la classification des codes vecteur de la base de données des caractères arabe manuscrits.

#### IV.2. Les Réseaux Neuronaux

##### IV.2.1. Introduction et historique (Rappel)

En 1943, deux bio-physiciens de l'université de Chicago, Mac Culloch et Pitts proposent le premier modèle de neurone biologique (formel), aussi appelé neurone à seuil, est inspiré des récentes découvertes en biologie [1]. Ce sont des neurones logiques (0, 1).

En 1949, le psychologue Donald Hebb introduit le terme connexionisme pour parler de modèles massivement parallèles et connectés. Il propose la célèbre " règle de Hebb" qui mise à jour les poids. En 1958, le psychologue Frank Rosenblatt propose le premier perceptron. Ce réseau, capable d'apprendre à différencier des formes simples et à calculer certaines fonctions logiques, est inspiré du système visuel [1].

Mais en 1969, deux scientifiques américains de renom, Minsky et Papert démontrent les limites du perceptron proposé par Rosenblatt [1].

Les travaux ralentissent considérablement jusqu'aux années 80. En 1982, Hopfield démontre l'intérêt des réseaux entièrement connectés. Parallèlement, Werbos conçoit un mécanisme d'apprentissage pour les réseaux multicouches de type perceptron: la rétropropagation (Back-Propagation). Depuis ces travaux, les applications des réseaux de neurones n'ont cessé de croître [13].

---

---

## IV.2.2 Présentation

Le fonctionnement d'un réseau de neurones est inspiré de celui du cerveau humain. Il reçoit des impulsions, qui sont traitées, et en s'ortie d'autres impulsions sont émises pour activer les muscles [13].

Il existe deux types de réseaux neurones artificiels :

- ↳ à apprentissage supervisé où la réponse est connue.
- ↳ à apprentissage non supervisé où le réseau ne connaît pas le résultat.

## IV.2.3 Fonctionnement d'un Réseau de Neurones

Les composants sont :

- ↳ Le neurone formel,
  - ↳ Une règle d'activation,
  - ↳ Une organisation en couches,
  - ↳ Une règle d'apprentissage.

### IV.2.3.1 Neurone formel

Le *neurone formel* calcule la somme pondérée par son poids de chaque entrée transmise par le reste du réseau  $S = \sum(X_i * W_i)$ , selon le schéma suivant :

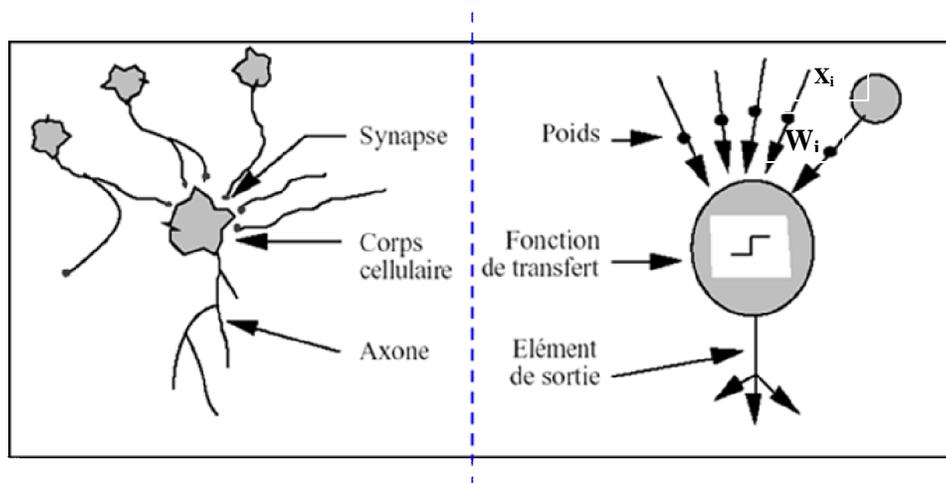


Figure N° IV-5 Schéma général d'un neurone biologique et artificiel

### IV.2.3.2 Règle d'activation

A chaque neurone formel du réseau est associée une *règle d'activation* avec une fréquence  $T$  et une fonction, généralement *sigmoïde* (voir schéma suivant), qui a pour résultat d'activer ou non le neurone de sortie [23].

$$\text{Si } \sum(X_i \times W_i) > \text{Seuil} \Rightarrow \text{Activation Du Neurone De Sortie.}$$

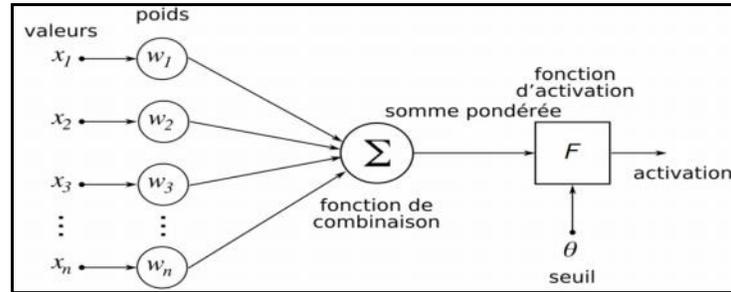


Figure N° IV-6 Modélisation d'un neurone artificiel

Où la fonction d'activation est une fonction  $f: \mathbb{R} \rightarrow \mathbb{R}$ , continue. En général, deux types de fonctions d'activation peuvent être utilisés :

- ☞ **L'Identité** :  $f: \mathbb{R} \rightarrow \mathbb{R}$  tel que  $f(u)=u$ .
- ☞ **La Sigmoïde**: On dit qu'une fonction est *Sigmoïde* si elle est **continue**, **croissante** et **bornée** dans  $\mathbb{R}$ .

Les fonctions *sigmoïdes* les plus utilisées sont :

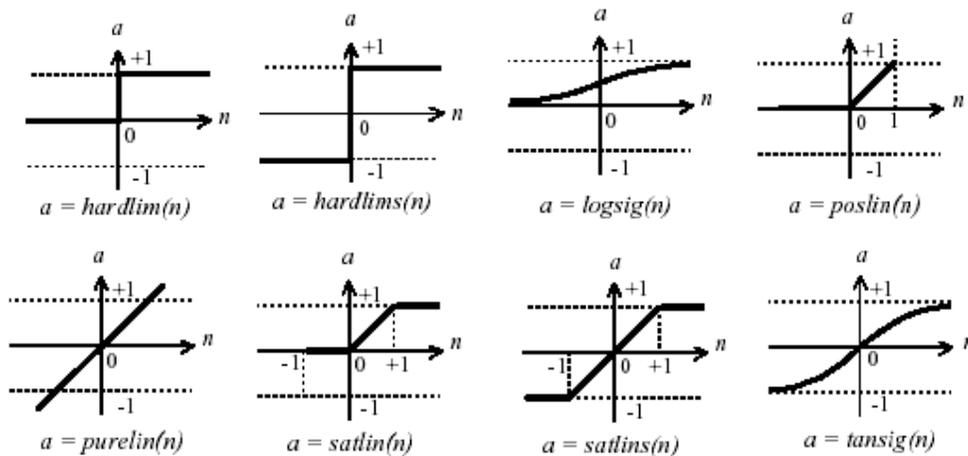


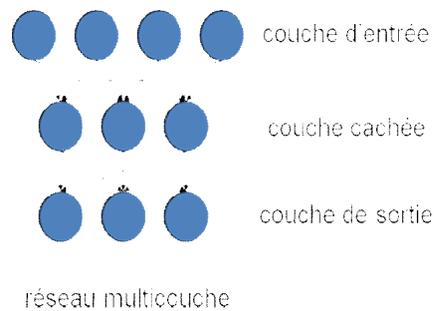
Figure N°IV-7 Fonction de sortie de type seuil, et Fonction de sortie bornée

---

---

### IV.2.3.3 Organisation en couches

Afin de résoudre des problèmes de toute complexité, le réseau a une organisation en couches. La couche d'entrée transmet ses résultats à la couche supérieure qui, ayant de nouvelles données et de nouveaux poids retransmet ses données résultantes à la couche suivante et ainsi de suite jusqu'à la couche de sortie qui fournit le neurone de sortie. On appelle les couches intermédiaires les couches cachées ( voir schéma suivant) [13].



**Figure N°IV-8 Organisation d'un réseau de neurones en couche**

### IV.2.3.4 Règle d'Apprentissage

La règle d'apprentissage permet au réseau d'évoluer dans le temps en tenant compte des expériences antérieures. Les poids des connexions sont modifiés en fonction des résultats précédents afin de trouver le meilleur modèle par rapport aux exemples donnés [1].

### IV.2.4 Mise en Œuvre d'un Réseau de Neurones

Le réseau de neurones est constitué des étapes suivantes [13] :

- ↳ Préparation des données,
- ↳ Détermination des paramètres,
- ↳ Phase d'apprentissage,
- ↳ Phase d'optimisation,
- ↳ Phase de test,
- ↳ Phase d'exploitation.

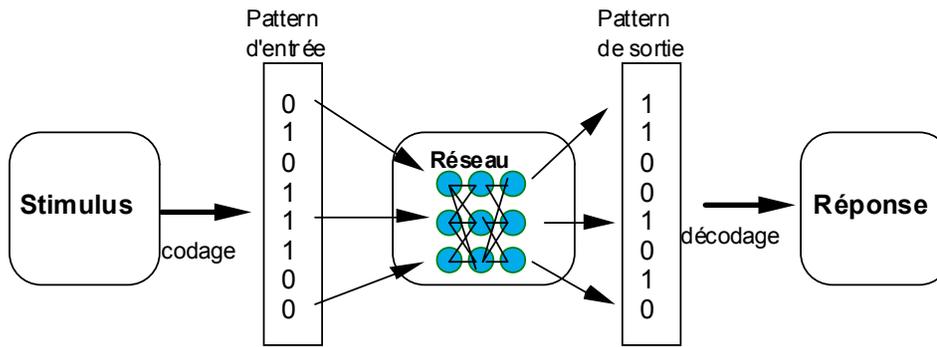


Figure N°IV-9 Présentation des données a un réseau de neurones

#### IV.2.4.1 Préparation des Données

La préparation des données consiste à déterminer les entrées et les sorties, constituer la base d'exemples, coder les entrées et enfin optimiser le jeu de données [13].

#### IV.2.4.2 Détermination des Entrées Et des Sorties

Il n'y a pas de règles en la matière, tout dépend du problème à traiter et des informations disponibles.

##### a) Constitution de La Base D'exemples

Le choix des exemples conditionne, bien entendu les résultats. Le but étant d'expliquer un phénomène. Si une action est positive dans 90 % des cas, ce n'est pas pour autant qu'il faille prendre un échantillon contenant 90 % des cas positifs aussi. Mieux vaut équilibrer et faire 50 / 50 afin de forcer le réseau à comprendre les 10 % négatifs [23].

##### b) Codage des entrées

Le codage des entrées est l'association des variables aux neurones. Ainsi une valeur continue sera représentée par un neurone alors qu'une variable discrète prenant, par exemple, les valeurs actives ou non actives seront codées par deux neurones [23].

##### c) Optimisation du jeu de Données

Le jeu de données constitué à l'étape précédente, peut contenir soit beaucoup trop de variables, qui alourdirait le travail d'apprentissage, soit des valeurs aberrantes qui fausseraient les calculs. Il est donc important de retravailler cet échantillon [23].

---

---

#### ***d) Détermination des paramètres***

***Architecture*** : détermination du nombre de couches et du nombre de neurones de chaque couche.

***Fonction de Sommation*** : définition de la nature de l'opérateur qui combine les poids des  $W_i$  (somme, minimum, maximum, majorité, produit). La plus utilisée est la somme.

***Fonction de Transfert*** : détermination de la fonction à appliquer sur le résultat de la fonction de sommation (sigmoïde, linéaire, gaussienne). La plus utilisée est la sigmoïde.

***Fonction de Normalisation*** : transformation de la sortie soit pour la rendre compréhensible par l'utilisateur, soit pour la mettre dans des limites acceptables pour les couches cachées.

***Fonction de Compétition*** : détermination de la ou des sorties qui seront présentes à la couche suivante (la plus forte, les deux plus fortes, ...).

***Calcul de l'Erreur (Err)*** : calcul de l'écart entre la sortie donnée ou désirée et la sortie calculée (***Err*** moyenne, ***Err*** absolue, ***Err*** quadratique, sens de variation de l'erreur).

#### **IV.2.4.3 Phase d'Apprentissage**

L'apprentissage consiste à descendre le réseau de façon itérative en ajustant les poids à chaque passage selon le calcul d'erreur jusqu'à ce qu'il n'y ait plus d'amélioration. Pour cela, un algorithme de rétro-propagation ( *backpropagation* ) de l'erreur est mis en œuvre[23]. Principes de l'algorithme de rétro-propagation ( *backpropagation* ) de l'erreur :

- Initialiser la matrice des poids au hasard et Choisir un exemple en entrée
- Propager le calcul de cette entrée à travers le réseau
- Calculer la sortie de cette entrée
- Mesurer l'erreur de prédiction par différence entre sortie réelle et sortie prévue
- Calculer la contribution d'un neurone à l'erreur à partir de la sortie
- Déterminer le signe de modification du poids  $W_i$
- Corriger les poids des neurones pour diminuer l'erreur

Le processus est itéré jusqu'à ce qu'un taux d'erreur minimal soit atteint (critère d'arrêt).

---

---

#### IV.2.5. Les réseaux de Neurones à impulsion « SPIKE »

Ce n'est que récemment que les modèles de neurones à impulsions, inspirés des neurones biologiques, sont utilisés dans le domaine de la reconnaissance de l'écriture, et spécialement l'écriture arabe manuscrite. Comme est indiqué dans l'introduction nous avons restreint notre recherche aux neurones Spike (à décharges) et à leur application.

##### IV.2.5.1. Inspiration du neurone biologique

Le neurone biologique (figure IV -10) possède en général trois parties : l'arbre dendritique, le noyau et l'axone. Brièvement, les signaux provenant de s autres neurones arrivent à l'arbre dendritique et sont transmis au corps cellulaire et à l'axone. La zone de transition entre le corps cellulaire et l'axone est d'un grand intérêt. En effet, c'est à cet endroit que l'essentiel du traitement non linéaire est effectué. Si l'excitation des entrées est suffisante, un signal de sortie est émis et propagé à travers l'axone et ses branches aux autres neurones. La jonction qui relie une branche de l'axone et la dendrite d'un neurone récepteur se nomme synapse. D'ailleurs, un neurone transmetteur est communément appelé un neurone pre-synaptique et un neurone post-synaptique fait référence à un neurone récepteur [23].

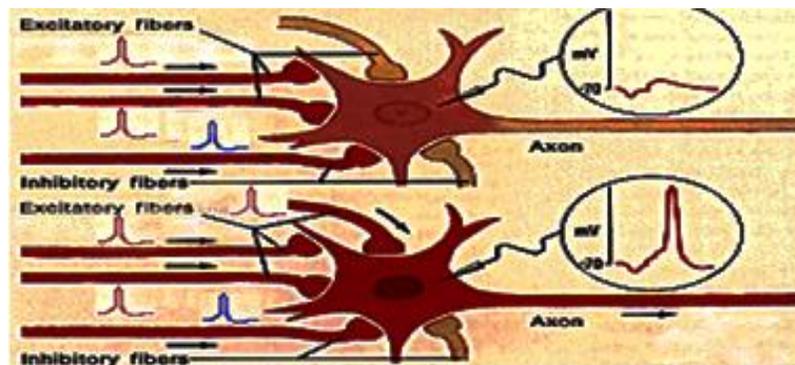


Figure N°IV-10 Le neurone, ses composants et leur activité électrique.

On notera en particulier en haut l'évolution du potentiel membranaire sous le seuil et l'émission de spikes (des potentiels d'amplitude plus faible), et en bas au niveau de l'axone, seuls les spikes sont visibles. En observant la différence de potentiel de l'axone, le signal observé est une séquence de courtes impulsions appelées potentiel d'action. En général, les impulsions produites à partir d'un même stimulus présentent un grand degré de variabilité.

---

---

On sait depuis un certain temps que les neurones utilisent ces potentiels d'action pour transmettre des signaux sur de longues distances.

Comme les impulsions d'un neurone sont semblables, la forme de l'impulsion renferme peu d'information. Les chercheurs s'accordent pour dire que la nature binaire de ces potentiels d'action signifie qu'ils codent l'information par leur présence ou absence et non pas par leur taille ou par leur forme. De plus, le nombre et la chronologie de ces impulsions constituent de bonnes sources d'information [30].

La biologie de l'intégration neuronale des impulsions, spécialement au niveau des dendrites, est aussi importante que le codage neuronal. Ce calcul local aux arbres dendritiques est accompli par la conductance de la membrane active.

#### **IV.2.5.2. Codage par cadence**

##### ***La cadence par le compte des impulsions***

Est essentiellement le calcul du nombre d'impulsions dans un intervalle de temps  $T$  divisé par la grandeur de cet intervalle [30].

##### ***La cadence par la densité des impulsions***

Fonctionne pour des stimuli stationnaires ou dépendants du temps. Dans ce cas, l'expérimentateur enregistre les impulsions générées par les neurones en simulant ceux-ci à l'aide d'une séquence d'entrées. Cette séquence est répétée plusieurs fois et le résultat est placé dans un histogramme. Pour chaque intervalle de temps court avant, pendant et après la simulation, l'expérimentateur compte le nombre de fois qu'une impulsion est produite et les additionne sur toutes les répétitions de l'expérience. Ensuite, le nombre d'occurrences des impulsions est divisé par le nombre de répétitions et par l'intervalle de l'histogramme pour obtenir la densité pour cet intervalle (figure IV -11) [30].

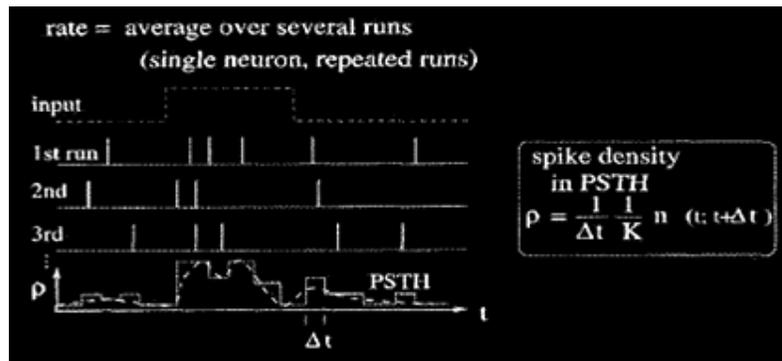


Figure N°IV-11 Code par cadence utilisant la densité des impulsions.

### *La cadence par l'activité d'une population*

Est justifiée, car souvent plusieurs neurones possèdent des propriétés et des réponses aux stimuli similaires. Il est alors possible de définir la cadence des impulsions à l'aide d'un ensemble (une population) de neurones [30].

#### IV.2.5.3. Codage par impulsions

##### *Le temps de la première impulsion*

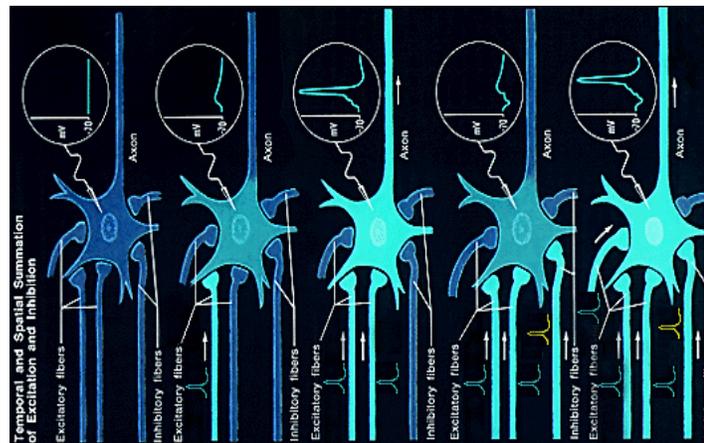
Est une méthode de codage par impulsions. Dans la version la plus pure de cette méthode, uniquement la première impulsion de chaque neurone compte. Toutes les impulsions subséquentes sont jugées inutiles. De la même façon, on peut assumer que chaque neuro ne n'émet qu'une impulsion par saccade et qu'elles sont par la suite inhibées. Il est évident que dans un tel cas, l'information est située dans la chronologie et non dans le nombre des impulsions [30].

##### *Le codage par ordre de rang (Rank Order Coding)*

Est une stratégie inspirée du système visuel au niveau duquel de nombreux types de codes auraient de la difficulté à opérer à la vitesse impressionnante que l'on retrouve pour traiter certaine information. Pour ce type de code, l'information est distribuée à t travers une grande population de neurones et elle est représentée par le temps de décharge relatif des impulsions dans une seule vague de potentiels d'action. En général, le taux de décharge d'un neurone est une fonction monotone de l'intensité de l'entrée (plus le neurone est excité, plus il décharge). Nous pouvons alors affirmer que la latence des décharges d'un

neurone, comme son taux de décharge, vont refléter l'intensité de l'entrée. Il est alors possible de dépendre de l'ordre spécifique dans lequel les premières impulsions ont été générées à travers la population. En effet, lorsqu'on présente un patron d'entrées à une population de neurones, la première impulsion de la vague de potentiels d'action générés est produite par le neurone le plus excité (figure IV-12) [30].

La quantité d'information qui peut être transmise par ce type de code augmente en fonction du nombre de neurones qui constituent la population. Pour un nombre relativement grand de neurones, la puissance de transmission d'un tel code peut satisfaire les besoins de n'importe quelle tâche visuelle [30]. Le fait d'utiliser l'ordre relatif et non pas les latences exactes procure des avantages, la stratégie est plus simple à implémenter, le code procure au système une certaine invariance au niveau des changements d'intensité des stimuli ou des contrastes, et l'information est disponible dès que le premier neurone décharge. Cependant, le système est incapable de faire des jugements précis à propos de la valeur de l'intensité des entrées.

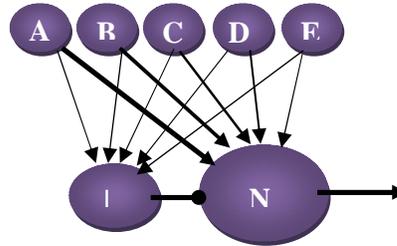


**Figure IV-12 - Vague de potentiels d'action générée par une entrée donnée, appliquée à une population de cinq neurones**

Par contre, on sait que le système visuel humain fonctionne mieux en comparant directement les caractéristiques des stimuli. Pour ce code, les neurones doivent donc être sensibles à la structure temporelle des impulsions qu'ils reçoivent en entrées. Ils doivent répondre de façon sélective à une séquence particulière d'activation de leurs entrées. Les patrons des impulsions d'entrées sont alors représentés dans le domaine temporel et un neurone donne une importance maximale à la première impulsion qu'il reçoit. L'influence des impulsions suivantes sur le neurone diminue progressivement.

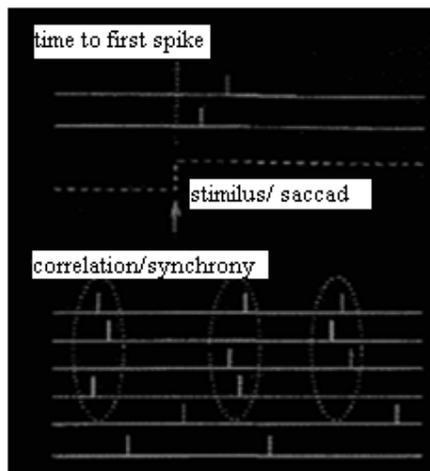
---

Le niveau d'activité résultant pour le neurone est le produit de ses poids synaptiques distribués à ses entrées et de la fonction de désensibilisation croissante. À l'aide d'un seuil approprié, le neurone (figure IV -13) peut alors être sensible à une séquence particulière de ses entrées.



**Figure IV-13 - Circuit d'un neurone, avec inhibition (I) et entrées pondérées (A, B, C, D et E), sensible à la séquence temporelle : A, B, C, D et E.**

**La corrélation et la synchronie** peuvent aussi jouer un rôle dans le codage. Il est aussi possible d'utiliser d'autres neurones comme signal de référence pour un code impulsif. Par exemple, la synchronisation entre deux ou plusieurs neurones (figure IV -14) aurait une signification particulière. D'un autre côté, on pourrait utiliser non seulement la synchronisation, mais aussi des patrons spatio-temporels précis des impulsions. Il est à noter que la corrélation des neurones auditifs est dépendante du stimulus et pourrait fournir de l'information au-delà de la cadence des décharges.



**Figure IV-14 - Code par le temps de la première impulsion (en haut), code par corrélation et synchronie (en bas). [50]**

---

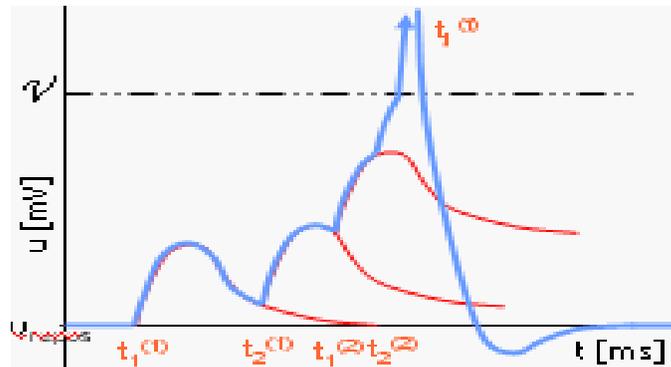
---

#### IV.2.5.4. Modèles de neurones Spike artificiels

**Le neurone avec seuil à décharges** est un modèle simple. Avec un niveau d'abstraction élevé, on peut considérer le neurone comme une unité homogène qui génère des impulsions lorsque son excitation totale est suffisamment importante [30].

**Le modèle simple de neurone à décharge** est décrit par une variable (le potentiel interne) et il peut recevoir des signaux de ses neurones pré-synaptiques. Si cette variable augmente par l'effet de ses entrées et atteint un seuil défini, le neurone décharge. Immédiatement après cette décharge, la variable qui représente le potentiel interne est souvent réinitialisée en additionnant une valeur négative. De cette façon, on obtient une période réfractaire à la suite de chaque décharge [30].

Un seuil dynamique (figure IV-15) peut aussi être utilisé. Dans ce cas, le seuil augmente après chaque décharge et reprend lentement sa valeur originale pendant les périodes où il n'y a pas de décharges.



**Figure IV-15 - Évolution d'un seuil dynamique**

**Le modèle d'impulsion.** L'état interne du modèle d'un neurone est décrit par une seule variable, le potentiel de la membrane. Lorsque ce potentiel augmente et dépasse la valeur d'un seuil, une impulsion est générée. Le résultat des activités au niveau des ions crée un pic escarpé du potentiel suivi d'un long potentiel négatif [30]. Il faut remarquer que lorsque le potentiel d'action augmente ou diminue rapidement, aucune émission d'impulsion n'est possible.

---

---

**Le modèle à intégration et décharge** constitue un exemple important de la classe de modèle avec seuil à décharge. Le comportement du neurone est défini par un intégrateur à fuite avec seuil au niveau du potentiel de la membrane [30]. Lorsque la valeur du potentiel de la membrane dépasse la valeur du seuil, le neurone décharge. Suivant la production d'une impulsion, le potentiel est réinitialisé à une nouvelle valeur. Une période réfractaire est souvent utilisée pour empêcher la production d'une seconde impulsion trop près de la précédente. Dans ce cas, on donne une valeur au potentiel de la membrane qui sera conservée pendant un certain temps. Ensuite, on redémarre l'intégration avec une valeur initiale.

Des **oscillateurs** hautement non linéaires peuvent définir des neurones à décharges. Ils présentent alors un comportement collectif qui peut montrer d'importantes caractéristiques au niveau des calculs [85].

La synchronisation est un comportement que l'on peut obtenir avec un modèle de neurone à intégration et décharges en utilisant des liens synaptiques excitateurs sans délais, ou des liens inhibiteurs avec délais. Ce phénomène peut être utilisé pour définir des fragments de données [85]. Ces fragments sont alors composés de neurones qui déchargent en même temps et les différents fragments possèdent un temps de décharge qui leur est propre. Il est aussi possible de décrire ce comportement comme une segmentation temporelle avec laquelle on sépare les données grâce au retard de phase entre l'excitation des différents groupes [30]. Cette séparation est caractérisée par un petit nombre comme limite de segments, une limitation inhérente au comportement d'oscillateur non linéaire.

La majorité des systèmes biologiques qui présentent une synchronisation peuvent être décrits comme des oscillateurs liés, où l'état parfaitement synchronisé n'est qu'un attracteur dynamique parmi d'autres. De plus, il est possible de diviser les modèles d'oscillateurs liés en deux types, oscillateurs liés par la phase et oscillateurs liés par les impulsions.

#### **IV.2.5.5. Applications des Réseaux de neurones impulsionnels**

Si on cherche à simplifier ces modèles, c'est aussi pour pouvoir les simuler plus simplement et également en plus grand nombre. Nous parlons de la problématique de savoir calculer avec de tels neurones et de les mettre en réseau avant de lister quelques applications utilisant ces modèles [85].

---

---

La première question qui se pose est celle de la maîtrise du pouvoir calculatoire de telles unités binaires, où toute l'information est codée dans le pattern temporel de leur activation et dans leurs interconnexions. Il s'agit d'un mode de calcul original d'un point de vue informatique et ce sujet intéresse également les biologistes qui veulent comprendre le « code neuronal » [30]. Deux stratégies de codage sont généralement décrites. Le codage fréquentiel propose de compter les spikes pour déterminer un niveau d'activité. Si on les compte dans le temps, sur un seul neurone, il faut attendre une durée assez longue avant d'avoir une estimation fiable, ce qui est souvent peu plausible biologiquement.

Il est aussi possible de les compter dans l'espace, sur une population de neurones, ce qui donne un temps de calcul plus acceptable, au détriment de la taille du réseau. La seconde stratégie de codage repose sur le temps. Elle postule que l'information peut aussi être contenue dans la synchronisation entre des neurones ou entre un neurone et un autre signal physiologique [85].

Ainsi, [30] souligne que la rapidité de traitement dans des tâches de discrimination visuelle implique que la décision se fait dès les tous premiers spikes et que seul un codage par phase, c-à-d par retard relatif d'un neurone par rapport à d'autres peuvent expliquer ce type de performances.

Concernant la mise en œuvre de réseaux de tels neurones, [85] donne une bonne description des méthodes et des supports de calcul utilisables.

Mentionnons simplement ici la dualité entre une approche événementielle où l'on cherche, le plus souvent analytiquement, à déterminer la date du prochain événement dans le réseau et à s'y projeter directement, et une approche numérique où les équations différentielles sont approximées par des méthodes de calcul de type Runge-Kuta ou Euler.

La réalisation de réseaux de neurones à spikes répond à deux besoins. D'une part, la réalisation de réseaux simples permet d'étudier certaines propriétés collectives de ces neurones. D'autre part, des réseaux de taille importante sont également développés pour réaliser des applications. On se rapportera également à [85] pour une liste des simulateurs disponibles pour implanter de tels réseaux.

Concernant le premier point, des études sur les liens entre connectivité, topologie, temps de transmission et synchronisation, ont été élaborées [30]. Ou bien sur les liens entre la

détermination des états stables de systèmes dynamiques et les capacités d'apprentissage et de mémorisation des réseaux correspondants [28] [30]. Concernant le deuxième point, pour l'utilisation de retards pour la reconnaissance de formes, ou de synchronisation pour la segmentation de scènes on peut citer [28] [85]. On peut également noter l'existence de modèles génériques de traitement de données comme la « Liquid State Machine », où l'information est codée sous forme impulsionnelle puis projetée dans des espaces de dimension élevée pour faciliter la discrimination. Enfin, soulignons également l'existence de travaux actuels, où des neurones impulsionnels sont utilisés pour parfaire la connaissance du vivant, et pour développer des réalisations technologiques (robot).

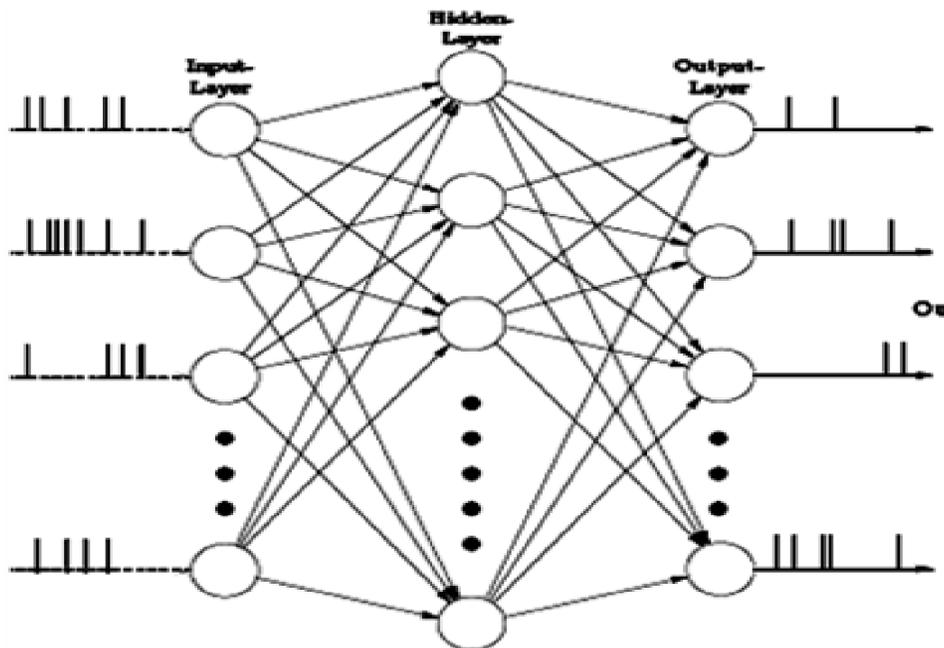


Figure N°IV-16 Feedforward spiking neural network, and connection consisting of multiple delayed synaptic terminals.

#### IV.2.6. Résumé

Il convient tout d'abord de mentionner que la famille de modèles que nous avons présentée ici est en fait une extension des modèles neuronaux classiques dans la mesure où, si l'on cherche l'état stable des équations différentielles mentionnées ici, on retrouve les équations habituelles des neurones artificiels classiques. Par leur aspect dynamique, ces modèles sont plus coûteux, mais aussi plus délicats à simuler. Cependant, comme nous avons voulu le montrer ici, ils offrent des caractéristiques inédites et très puissantes en intégrant des propriétés nouvelles du fonctionnement neuronal et en particulier des

---

---

caractéristiques de calcul temporel asynchrone qui permettent de nouvelles perspectives pour le traitement et l'analyse des données.

Il existe un grand nombre de méthodes permettant de réaliser l'apprentissage neuronal. De manière générale, ces méthodes sont supervisées, elles nécessitent des exemples pour réaliser l'apprentissage et elles sont difficilement applicables dans les cas d'auto-apprentissage et d'adaptation où aucun exemple de comportement n'est connu.

### **IV.3 Conclusion :**

Dans ce chapitre, nous avons présenté deux méthodes différentes de classification fondées sur les machines d'apprentissage : le réseau de neurone impulsionnel (SNN), et les séparateurs à vaste marge (SVM : Support Vector Machines). Aussi, nous abordons dans le prochain chapitre les résultats obtenus par ces méthodes de classification appliquées à la reconnaissance des caractères arabes manuscrits.

En général, la plupart des méthodes d'apprentissage machine comme réseau de neurones possèdent un grand nombre de paramètres d'apprentissage à fixer par l'utilisateur. En contrepartie, la formulation élégante de *SVM* laisse très peu de place aux paramètres utilisateurs.

---

---

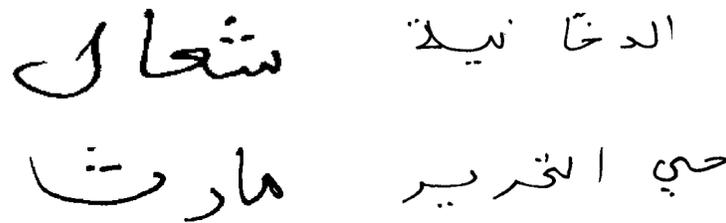
## **Chapitre V : Résultats expérimentaux et Discussions**

### **V.1 Introduction :**

Dans les chapitres précédents, nous avons présenté les différentes étapes d'un système complet de reconnaissance hors ligne (offline) de l'écriture arabe manuscrite. Dans ce chapitre nous allons évaluer les performances du système AHOCR qui a été mis en œuvre, en fonction de la méthode de segmentation implémentée et de chacun des différents algorithmes de reconnaissance proposés, pour permettre de choisir l'algorithme offrant les meilleures performances avec cette méthode de segmentation. Les versions des algorithmes utilisés, SVM et SNN, sont disponibles dans notre laboratoire SIMPA.

### **V.2 Description de la base de données *IFN/ENIT DATABASE***

Lors de l'évaluation des performances d'un système de reconnaissance il est nécessaire de disposer d'une base de données. Il est possible d'obtenir auprès de l'IFN/ENIT <http://www.ifnenit.com/> la base de données IFN/ENIT DATABASE. C'est une nouvelle base de données, qui contient des noms manuscrits de villes/villages tunisiens en arabes. Pour chaque nom les informations de base, par exemple l'ordre des formes de caractère, les informations sur le style de l'écriture, et la ligne de base sont codées. Plus de 400 auteurs ont rempli des formulaires avec plus de 26400 noms contenant plus de 210 000 caractères.



*Figure N° V-01 : Des échantillons de la base IFN/ENIT*

La base de données *IFN/ENIT DATABASE* est divisée en quatre ensembles disjoints : *set\_a*, *set\_b*, *set\_c*, et *set\_d* d'images sous deux types de formats : BMP et TFF, avec une résolution de 300 dpi (noir et blanc). Elle est conçue pour la formation et pour tester les systèmes de reconnaissance hors ligne de l'écriture arabes manuscrite. La base de données IFN/ENIT est disponible pour la recherche. Pour plus d'informations et de détails sur la base de données IFN/ENIT, consulter le site <http://www.ifnenit.com/>.

---

---

Le tableau suivant décrit la répartition des mots et le nombre de scripteurs à travers ces ensembles :

*Le tableau N° V-01 : décrit la répartition des mots et le nombre de scripteurs :*

<i>Ensemble</i>	<i>Nombre de mots</i>	<i>Nombre de scripteurs (ok+bad)</i>
<i>Set_a</i>	<i>6537</i>	<i>88+14=102</i>
<i>Set_b</i>	<i>6710</i>	<i>89+13=102</i>
<i>Set_c</i>	<i>6477</i>	<i>88+15=103</i>
<i>Set_d</i>	<i>6735</i>	<i>90+14=104</i>
<i>SUM</i>	<i>26459</i>	<i>355+56=411</i>

### **V.3 Les ressources matériels et logiciels :**

Les ressources matériels et logiciels utilisées dans ce travail se résument dans la liste suivante : la base de données est l'ensemble *set\_a* de la base IFN/ENIT DATABASE, un PC portable Toshiba (Dual CPU 1.6 Ghz, et 2 Go de RAM), et l'environnement de développement Matlab 7, avec les ToolBox s de SNN et de SVM (versions disponibles dans notre laboratoire SIMPA).

### **V.4 Description et évaluation du système AHOCR :**

#### **V.4.1 L'acquisition de l'image :**

Cette phase de travail ne pose pas de problème parce que les formes des images dans la base IFN/ENIT sont bien préparées, et sous deux formats : bitmap ('.bmp') et Tiff en noir et blanc; car il est très facile de manipuler les objets bitmap par les langages de programmation.

#### **V.4.2. Prétraitement :**

Dans cette étape on procède à suite d'opération, on utilise :

- ✓ Binarisation (pour le cas général), et des techniques pour éliminer le bruit par seuillage
- ✓ Techniques de filtrage, et opérations morphologiques : tel que le filtre médian, érosion, le filtre de Sobel (pour la squelettisation).
- ✓ Des opérations de redressement de l'image par des rotations pour remédier aux problèmes d'inclinaisons verticales et horizontales des mots et des caractères.

La figure suivante montre les différentes étapes de notre AHOOCR, étape par étape.

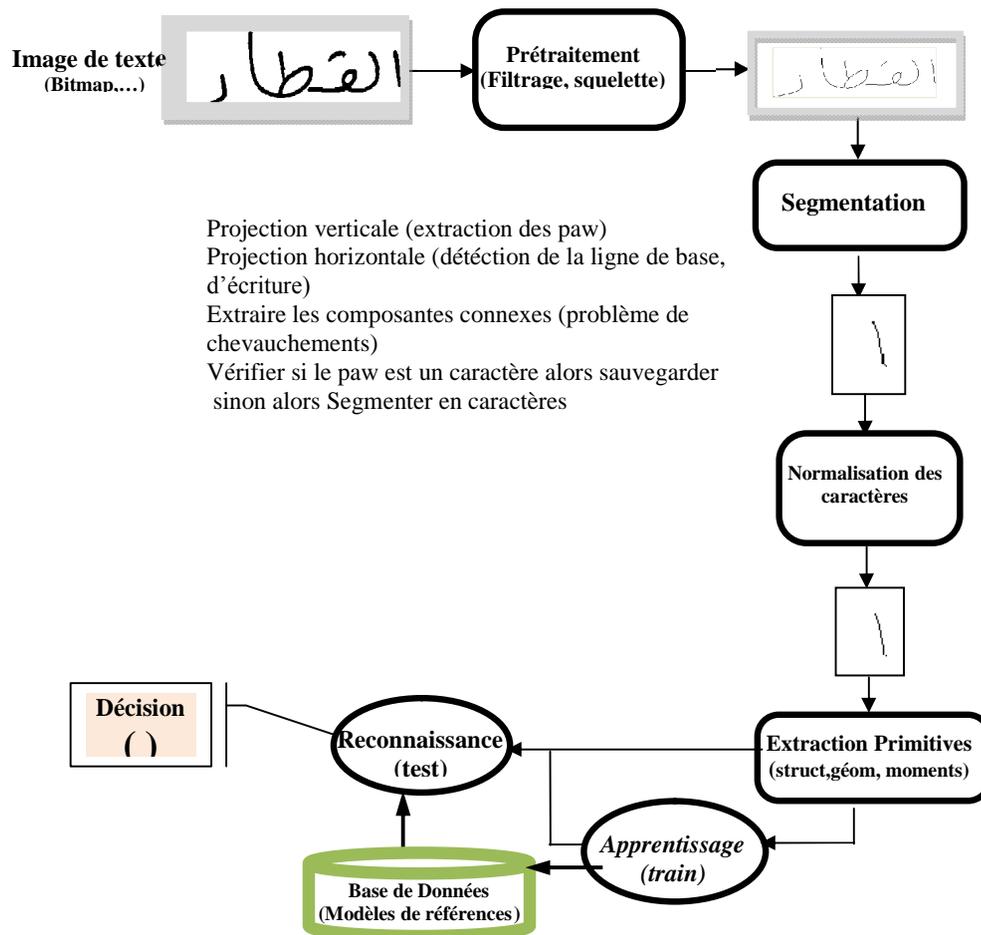


Figure N°V-02 : Schéma général du système AHOOCR développé

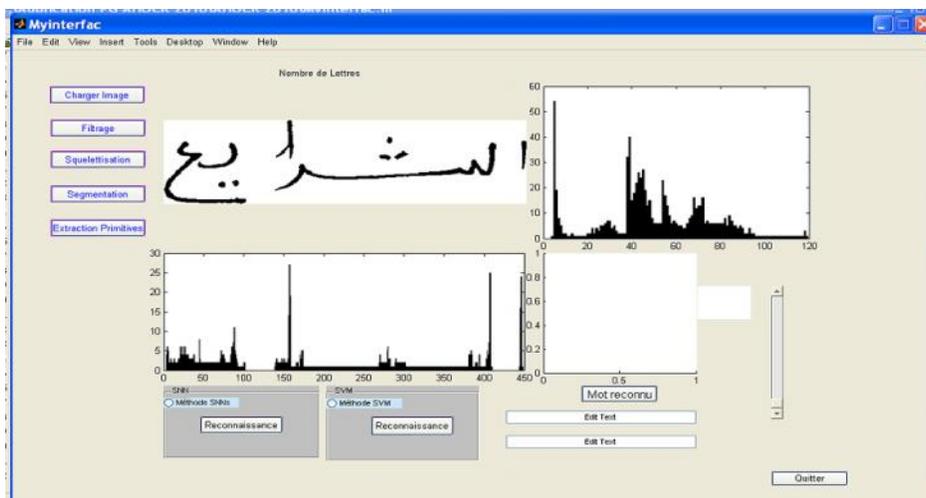


Figure N° V.3 : la fenêtre principale de notre système AHOOCR

### V.4.3. La segmentation :

Puisque le premier but de notre projet est l'amélioration de la méthode de segmentation de l'écriture arabe manuscrite, il s'agit de trouver un rectangle qui inclut un seul caractère dans sa surface de pixels. Pour cela, nous avons choisi comme méthode de segmentation pour un premier prototype la méthode des projections qui consiste en la projection du texte sur une droite verticale pour pouvoir définir les régions occupées par les lignes du texte. Ensuite on applique une autre projection horizontale pour chaque ligne de texte afin d'extraire les largeurs des cadres incluant les mots, les paw et les caractères isolés, ( figure N° V-4).

En prenant en compte que dans l'écriture arabe deux caractères adjacents sont soit connectés (au niveau de la ligne de base) ou bien ne sont pas connectés, on cherche alors les caractères de référence qui vont être considérés comme des repères dans les mots, puisque grâce à la présence de ces caractères dans les mots on obtient les paw. Ces caractères sont au nombre de 6 caractères parmi les 28 lettres du vocabulaire arabe, et qui n'admettent pas un rattachement à leur gauche (propriété pertinente).

Ces lettres sont : { و، ز، أ، د، ذ، ر }.

Après avoir extrait les mots et les paw, et pour résoudre le problème de chevauchement des paw (des lettres), on a créé une fonction qui cherche les composantes connexes. C'est une fonction récursive qui part d'un point en examinant ses 8-voisins afin de retrouver les coordonnées du cadre des composantes connexes. Elle est décrite dans la figure N° V -5.

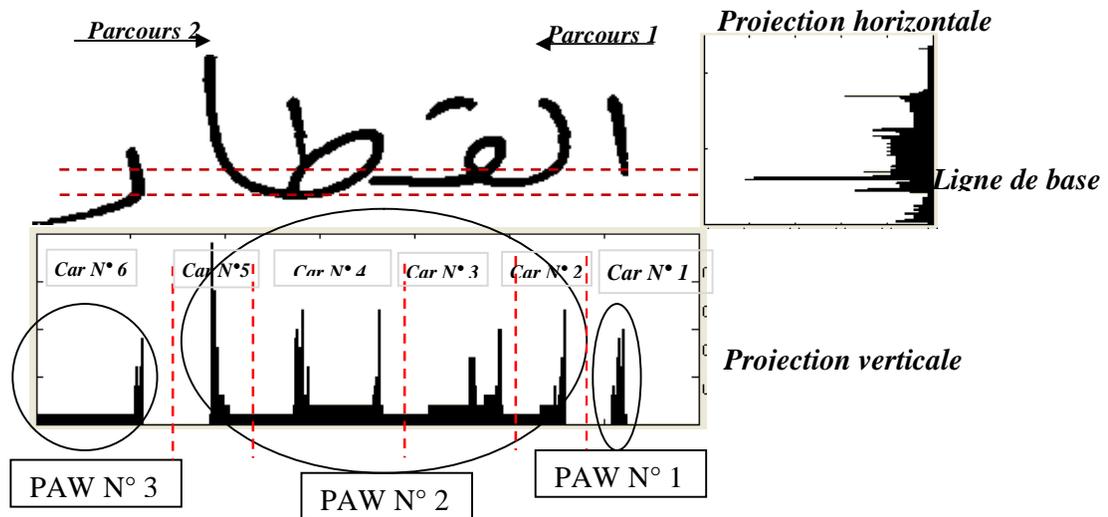


Figure N°V-04: Description de la méthode de segmentation proposée

La deuxième technique ajoutée à la phase de segmentation, est l'utilisation des connaissances à priori sur les caractéristiques pertinentes de l'ensemble du vocabulaire arabe pour segmenter les paw en caractères.

Ces caractéristiques sont par exemple la position de la lettre (début, milieu, fin, isolé) dans le mot (paw), les caractéristiques de hampe et de jambage (à l'aide de la ligne de base de l'écriture), l'existence de points diacritiques (au dessus ou en dessous des lettres), et la taille des caractères obtenus (un minimum de hauteur et de largeur).

Cette technique a pour but de réduire le vocabulaire en certaines catégories de caractères ayant des caractéristiques communes, sans exclure aucune forme de caractère.

Elle permet ainsi de faciliter la recherche, le calcul, et un gain de temps pour la phase de reconnaissance (au lieu de parcourir l'ensemble du vocabulaire, on procède au parcours d'un sous ensemble de lettres).

En outre, cette méthode permet de vérifier si des paw si sont des caractères.

Les catégories de caractères établies dans notre cas, sont au nombre de 4 classes selon la présence de hampes et/ou de jambages. Ces deux caractéristiques sont expliquées dans la figure N°VI.

La hampe est un morceau dans le corps du caractère qui occupe la partie supérieure de la ligne de base (souvent comme bâton).

Le jambage est un morceau dans le corps du caractère qui occupe la partie inférieure de la ligne de base.

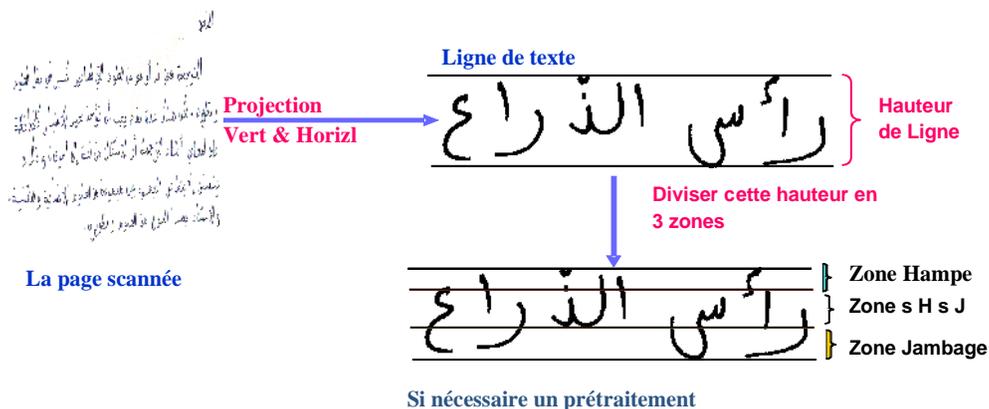


Figure N°V-05: Description de la présence d'hampe et de jambage

Chaque classe est subdivisée selon la présence des points diacritiques.

Le plus important, c'est que l'application de cette technique est faite sur les caractères dans les quatre positions possibles : isolé, début, milieu, fin, puisque il y a des changements remarquables dans certains caractères. Les résultats de cette technique sont utilisés dans la phase de l'extraction des primitives.

**Tableau N°V-02 : Avec hampe: 4 caractères**

N°	Lettre isolé	Au début	Au milieu	A la fin	Prononciation
01	/ /	/ /	/ /	/ /	Alif
02					TTa
03					Dha
04					Kaf

**Tableau N°V-03 : Avec hampe et jambage : un seul caractère**

N°	Lettre isolé	Au début	Au milieu	A la fin	Prononciation
01					Lam

**Tableau N°V-04 : Sans hampe sans jambage : 20 lettres**

Les lettres suivants { ذ, د, ه, ب, ث, ت, ف } sont stables dans cette catégorie (7 caractères)

N°	Lettre isolé	Au début	Au milieu	A la fin	Prononciation
01					Ba
02					Ta
03					Tha
04	/			/	Jim
05	/			/	Ha
06	/		خ	/	Kha
07	د	د			Dal
08	ذ	ذ	ذ	ذ	Dhal
09	/	س		/	Sin
10	/	ش	ش	/	Chin
11	/	ص	ص	/	Sad
12	/	ض	ض	/	Dhad
13	/	ع	ع	/	Ayn
14	/	غ	غ	/	Ghayn
15			ف	ف	Fa
16	/	ق	ق	/	Qaf
17	/		م	/	Mim
18	/	ن	ن	/	Noun
19		ه	ه	ه	He
20	/	ي	ي	/	Ya

**Tableau N°V-05 : Avec Jambage: 16 caractères**

N°	Lettre isolé	Au début	Au milieu	A la fin	Prononciation
01	ج	/	/	ج	Jim
02	ح	/	/	ح	Ha
03	خ	/	/	خ	Kha
04			ر	ر	Ra
05	ز	ز	ز	ز	Zay
06		/	/	س	Sin
07	ش	/	/	ش	Chin
08	ص	/	/	ص	Sad
09	ض	/	/	ض	Dhad
10	ع	/	/	ع	Ayn
11	غ	/	/	غ	Ghayn
12	م	/	/	م	Mim
13	ق	/	/		Qaf
14	ن	/	/		Noun
15					Waw
16		/	/		Ya

Pour profiter de la technique des connaissances à priori, le parcours de l'image (squelette d'image) pour la segmentation se fait dans les deux sens : à partir du début (caractère en position début) et à partir de la fin (caractère en position fin), les autres lettres ont par conséquent la propriété de la position milieu.

On cherche ensuite à extraire les caractères par l'algorithme suivant, qui est itéré deux fois. Une première itération est opérée pour extraire les paw (zones séparées en utilisant les espaces intra-mot, et la fonction d'extraction des composantes connexes) et une deuxième itération sert à extraire et vérifier les formes segmentées (caractères).

Les algorithmes suivants décrivent ces différentes approches, sachant que les dimensions de l'image (squelette du mot) sont données par A : [haut larg]=size(A).

***L'algorithme de projection horizontale***

```

for( i=1 : haut)
  for( j=1 :larg)
    if (A(i,j)==0) hh(i)=hh(i)+1; { hh Le vecteur des frequences horizontales}
  end
end

```

***L'algorithme de projection verticale***

```

for( j=1 : larg)
  for( i=1 :haut)
    if (A(i,j)==0) hv(j)=hv(j)+1; { hv Le vecteur des frequences verticale}
  end
end

```

À l'aide de ces deux algorithmes on obtient le nombre, les hauteurs des lignes (hh,) et les mots (paw) dans chaque ligne (hv).

---

---

### *L'algorithme de segmentation des mots (paw) en caractère*

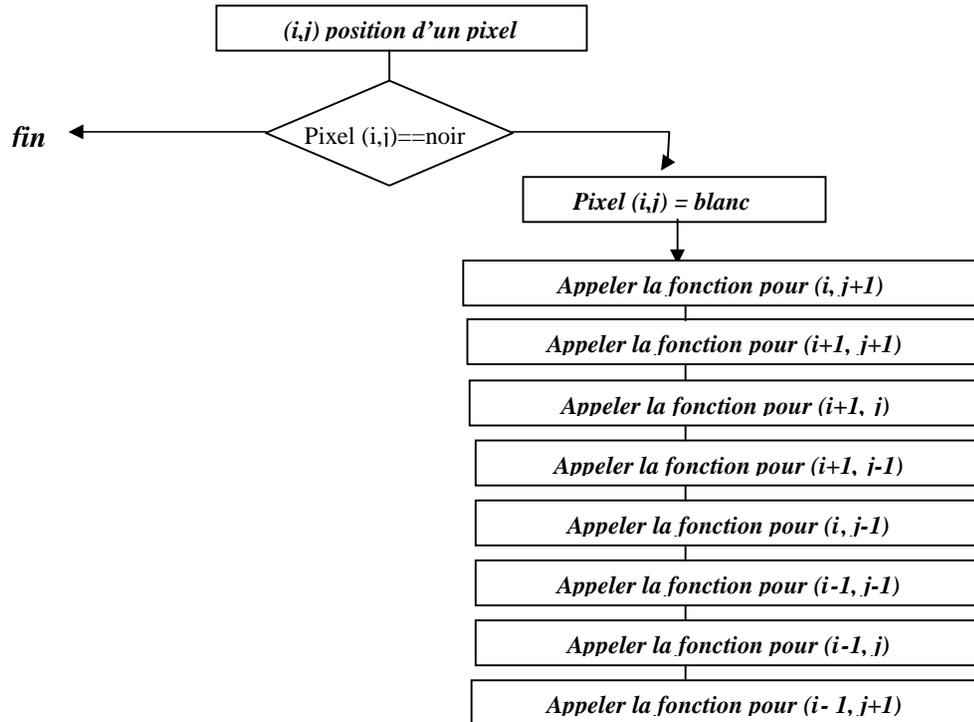
*Pour trouver les paw dans un mot, initialiser nbf= nombre de zéro dans le vecteur hv*

*Pour trouver les caractères dans un paw, initialiser nbf= nombre de un dans le vecteur hv*

*Puis on applique la procédure suivante :*

```
while( i<nbf )
  if(hv(i+1)-hv(i)>5)
    cptf=cptf+2;
    marqf(cptf-1)= hv(i); {marque les coordonnées des caractères dans l'image}
    marqf(cptf)= hv(i+1);
  i=i+1;
else i=i+1;
end
```

### *L'organigramme de la fonction de recherche des composantes connexes :*



*Figure N° V-06: Organigramme de la fonction de recherche des composantes connexe*

Dans la partie de l'extraction des primitives, ces caractéristiques (structurelles, géographiques, statistiques, position, largeur, hauteur) sont utilisées avec d'autres pour constituer l'ensemble des primitives de l'image de caractère segmenté (vecteur de caractéristiques pour chaque lettre), qui va être présenté à la méthode de reconnaissance.

---

Et puisque notre système est un OCR arabe, on a créé une fonction réciproque de la phase de segmentation, qui est une sorte de concaténation des lettres reconnus, pour reconstruire le mot d'origine, et lui affiché et stocker dans un fichier texte (notepad). Cette fonction se base sur les points initiaux suivants :

- ✓ Le nombre de lettre dans le mot à reconnaître.
- ✓ Le nombre et l'ordre des paw segmentés d'un mot.
- ✓ La position, l'ordre et le nombre de le ttre segmenté dans le paw.
- ✓ Les espaces intra-mots (entre deux paw).

#### V.4.4. Normalisation

L'étude de discernement nécessite l'élimination des conditions qui peuvent fausser les résultats, comme la différence de taille. Il faut donc aboutir à la normalisation de la taille des caractères. Une application doit être définie pour effectuer cette tâche. Cette application doit être réversible pour pouvoir aller d'une taille à une autre et revenir par la suite.

$$G : \mathbb{R}^4 \longrightarrow \text{ensemble des couleurs}$$

$$(x1,y1) \longrightarrow \text{couleur (noir/blanc)}$$

G est définie de la façon suivante :

$$\left\{ \begin{array}{ll} G(x,y)=\text{noir} & \text{si pixel } (\mu_1 x, \mu_2 y)=\text{noir} \\ G(x,y)=\text{blanc} & \text{sinon} \end{array} \right.$$

Tel que :  $\mu_1 = (\text{largeur du cadre du caractère}) / (\text{largeur du cadre de la normalisation})$   
 $\mu_2 = (\text{hauteur du cadre du caractère}) / (\text{hauteur du cadre de la normalisation})$

Le cadre proposé dans notre système pour la normalisation des caractères segmentés, est d'une taille : Hauteur 42 / Largeur 24, avec laquelle nous gardons l'ensemble de l'information à l'aide de la fonction définie précédemment.

Les formes (caractères) extraites dans l'étape précédente, vont être sauvegardées dans 28 classes (classes de caractères, selon le vocabulaire de la langue arabe), mais dans

chaque classe il y a 4 sous classes qui déterminent la position des lettres dans le mot ( début, milieu, fin, isolé).

Ces 28 classes sont ensuite distribuées selon le classement déjà décrit plus haut sur les 4 catégories (selon les propriétés de jambage et hampe), afin de faciliter la recherche et réduire le calcul et le temps pour la reconnaissance.

#### V.4.5 L'extraction des primitives :

Cette étape est faite selon deux catégories de caractéristiques : les caractéristiques structurelles/statistiques/géométriques, et les caractéristiques extraites par la technique des moments invariants de chaque image normalisée du caractère extrait.

On rassemble dans cette phase les caractéristiques utilisées dans la phase de segmentation (position, structure, dimensions, propriétés...) en un seul vecteur avec les moments invariants calculés pour chaque caractère à reconnaître. Les moments invariants sont détaillés dans l'annexe B, et la démarche de codage est définie dans l'annexe A.

Les deux tableaux suivant présentent les différentes caractéristiques structurelles et géométriques, et les valeurs des moments invariants de quelques caractères :

*Le tableau N° V- 06 : Exemple des caractéristiques de la lettre ‘ ’*

Primitives	H	J	H /J	SHJ	PtsH	PtsB	Crle	Cxg
Position caractère								
Isolé	0	1	0	0	1	0	0	0
Debut	0	0	0	1	1	0	0	1
Milieu	0	0	0	1	1	0	0	1
Fin	0	1	0	0	1	0	0	0

*H :hampe, J : jambage, SHJ : sans hampe ans jambage, PtsH/B : points dessus/ dessous, Crle : cercle, Cxg : admettre une connexion a gauche*

**Tableau N° V-07 : les valeurs des moments invariants ( $\Phi_i$ ) de certains caractères**

	ة	ى	ف	ظ	ا
$\Phi_1$	-1.180	-1.422	-1.595	-1.536	0.375
$\Phi_2$	-4.027	-5.823	-4.790	-6.244	0.636
$\Phi_3$	-7.293	-9.088	-7.642	-8.935	-2.668
$\Phi_4$	-6.459	-8.621	-10.190	-13.024	-2.804
$\Phi_5$	-13.338	-17.739	-19.181	-24.368	-5.541
$\Phi_6$	-8.576	-11.578	-12.584	-16.325	-2.582
$\Phi_7$	-13.175	-17.404	-20.016	-23.981	-9.058

Nous n'avons pas utilisé les images des caractères segmentés telles qu'elles sont pour la phase de reconnaissance, et ce dans le but :

- d'éviter l'absence de certaines informations (telles que : la position dans le mot)
- d'éviter les ambiguïtés entre les lettres manuscrites, qui sont dues à la variation de l'écriture manuscrite (ressemblance de certains caractères)
- enfin de profiter de la réduction du vocabulaire étudiée auparavant.

Donc les données à traiter par les méthodes de reconnaissance, SNN et SVM, sont de la forme suivante : un vecteur constitué de 12 premiers éléments (représentant les caractéristiques structurelles et géométriques/statistiques), et de 7 éléments (les valeurs des moments invariants (matrice 4\*4) de chaque caractère).

*Exemple de vecteur des caractéristiques* de la lettre imprimée « ف » (les positions ne sont pas représentés dans ce vecteur pour ce caractère, elles valent tous 0) :

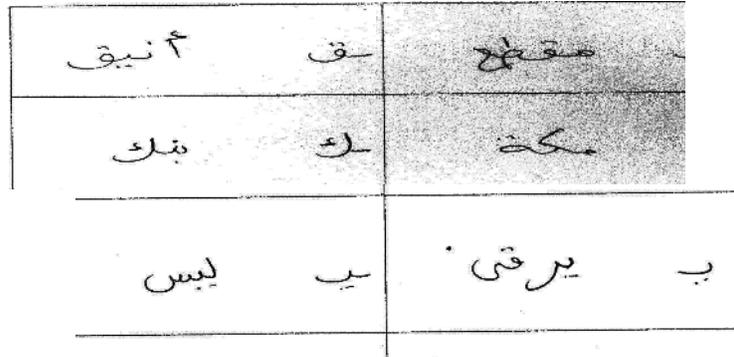
1	2	3	4	5	6	7	Jamb	Hamp	PtsH	PtsB	Cxg	Crle
-1.59	-4.79	-7.64	-10.1	-19.1	-12.5	-20.0	0	0	1	0	1	1

#### **V.4.6 Test et Résultat :**

Il faut premièrement prendre des pages comme pages de test afin d'avoir des résultats qui permettent de donner une idée sur le discernement.

Alors, au début de ce projet on a travaillé sur une petite base de données de l'écriture arabe manuscrite créée dans notre laboratoire SIMPA. Pour cela, nous lui avons attribué le

nom de NotreBase, qui est simple et qui est constituée d'un petit nombre de mots (4 mots selon les 4 positions de lettres \* 28 lettres\*10 scripteurs).



**La figure N° V- 07 : Quelques échantillons de de la base Notre base.**

Pour la phase segmentation, les premiers essais de segmentation des mots en caractères, appliqués à la petite base de données NotreBase (elle est créée dans le LABO SIMPA), montrent l'efficacité de cette technique où on a abouti à un taux de 91%.

Ce qui nous a poussé à optimiser et évaluer cette approche de segmentation pour la nouvelle base IFN /ENIT database ; où on a abouti à un taux de segmentation des mots en caractères de 88 %, sur les exemples choisis (450 mots) de l'ensemble de Set\_a de la base IFN /ENIT database.

Ce qui indique que la méthode de segmentation développée (hybridation de plusieurs techniques) a donné des bons résultats. L'ensemble des résultats de segmentation sont présentés dans le tableau suivant :

**Le tableau N°V-08 : Les résultats de segmentation**

Total de mots dans Set_a de IFN/ENIT DATABASE	Nombre de mots utilisés	Nombre de caractères Dans les 450 mots	Taux de segmentation des mots en caractères
6537	300 (train)+ 150 (test)	Environ 2.100 lettres	88%

Pour la partie reconnaissance, on procède d'abord à la reconnaissance des images normalisées de lettres segmentées, via leurs vecteurs des caractéristiques. La technique des moments invariants a montré une très grande capacité d'invariance face aux translations, aux variations et aux orientations des images des caractères manuscrits.

Le tableau suivant montre les résultats obtenus par les deux méthodes de reconnaissance utilisées : SPIKE & SVM appliquées sur les exemples choisis de la base IFN /ENIT database (corpus Set\_a) :

**Le tableau N°V-09 : Les résultats obtenus par les méthodes de reconnaissance**

	Durée d'apprentissage (en ms)	Taux d'apprentissage en (%)	Durée de reconnaissance en (ms)	Taux de reconnaissance en (%)
<b>SVM</b>	3.287	98,7%	1.076	76,4%
<b>SNN</b>	2.089	97,3%	798	69,2%

$Tau\ de\ reconnaissance = (Nbr\ de\ lettre\ reconnus * 100) / (Nbr\ lettre\ appris) ;$

A l'aide de la fonction réciproque de la segmentation (qui a été décrite plus haut, dans la phase de segmentation), nous opérons la phase de concaténation des lettres reconnues, en procédant à la reconstitution du mot d'origine à travers les caractères reconnus. Elle a permis à notre système AHOCR de donner les résultats suivants :

**Le tableau N°V-10 : Les résultats de la reconstitution des mots**

	Nombre de mots	Nombre de mots segmentés	Nombre de mots reconnus a base SVM	Nombre de mots reconnus a base SNN
<b>Apprentissage</b>	300	396	167	151
<b>Test</b>	150		73	66

## V.5 Interprétation et critiques:

Les résultats obtenus pour la méthode de segmentation et les taux de reconnaissance pour les deux algorithmes SNN et SVM sont encourageants par rapport à la littérature et à la difficulté liée à la manipulation de l'écriture arabe manuscrite, la segmentation et l'extraction des primitives.

Ainsi, les remarques que nous avons tirées de cette étude sont les suivantes :

- ☞ La méthode de segmentation proposée reste efficace malgré sa complexité.
- ☞ Les moments invariants donnent réellement des caractéristiques pertinentes.
- ☞ La méthode SVM donne des résultats meilleurs que la méthode SNN.
- ☞ Il s'avère cependant que la méthode SNN est plus rapide que la SVM, (temps de calcul plus court)

- 
- 
- ☞ Par rapport à la résistance à la variation de l'écriture arabe (lettres intra classe), la méthode SVM est plus stable et plus puissante que l'algorithme de SNN.
  - ☞ Le problème majeur résidant dans la phase de segmentation, est la sur-segmentation (plus d'un caractère est segmenté, fausse détection), et la sous-segmentation (partie d'un caractère, non détection). Exemple : la ligature verticale.
  - ☞ Les phases de segmentation et d'extraction des primitives ont une grande influence sur la phase de reconnaissance.

Il reste à dire que notre système considère que chaque caractère a quatre formes différentes selon sa position dans le mot, parce qu'il y a des vrais changements dans la plupart des lettres. Il est vrai que nous avons augmenté le nombre de formes mais d'autre part nous avons établi une catégorisation des lettres avec leurs propres caractéristiques, en des sous ensembles de quelques nombres de lettres dans chaque catégorie ; c'est une sorte de réduction du vocabulaire localement sans exclure aucune forme de lettres ou de diacritiques.

## **V.6 Conclusion :**

D'après les résultats obtenus pendant tout le travail fait, on peut dire que l'objectif est atteint en grande partie surtout ce qui concerne la phase de segmentation et de l'extraction des primitives, mais pour la partie reconnaissance les résultats sont acceptables par rapport à la littérature.

Les taux obtenus de nos jours en reconnaissance de l'écriture arabe manuscrite (offline or online) ne sont pas encore parfaitement suffisants, et la programmation qu'ils requièrent est nettement plus exigeante que celle que nous pouvions développer dans le cadre d'un projet scientifique, puisque c'est un domaine interdisciplinaire (la nécessité aux matériels, traitement d'image, la théorie d'information, méthode fiable d'apprentissage automatique,...). Le programme que nous avons développé est déjà très complexe, et a nécessité plusieurs mois de travail.

Pour l'amélioration de ce travail, et pour des perspectives futures dans notre laboratoire SIMPA, nous proposons d'optimiser les techniques de segmentation, d'extraction et de reconnaissance en utilisant d'autres moyens, comme par exemple au niveau de la représentation des données, des transformations aptes à garder le maximum d'information des données.

---

---

## **Conclusion générale et Perspectives**

Nous avons présenté des concepts généraux liés à la reconnaissance optique des caractères en générale, et de l'écriture arabe manuscrite hors-lignes comme cas particulier, en précisant les principales méthodes de prétraitement, de segmentation, d'extraction des caractéristiques et de reconnaissance. Nous avons aussi énuméré les principaux problèmes rencontrés par l'OCR Arabe.

Ce travail consistait à la réalisation d'un système complet de reconnaissance de l'écriture arabe manuscrite hors-lignes en utilisant la base de IFN/ENIT DATABASE, par application de deux algorithmes d'apprentissage automatique : le *réseau de neurones à impulsions* (SNN : *Spike Neural Net*), et les *Séparateurs à Vaste Marge* (SVM : *Support Vector Machines*) multi-classes. Les caractères segmentés et normalisés sont ensuite codés à base des primitives extraites par les moments invariants, combinés avec les caractéristiques obtenues à travers la phase de segmentation en des vecteurs de caractéristiques pertinentes, qui sont présentés aux algorithmes de reconnaissance.

Les résultats obtenus sont encourageants par rapport à la difficulté liée à la manipulation de l'écriture arabe manuscrite, à la segmentation et à l'extraction de primitives, tel que le meilleur score obtenu par l'approche de segmentation proposée est 88% pour la base IFN/ENIT DATABASE, en se basant sur les caractéristiques structurelles, statistiques et géométriques. En ce qui concerne les taux obtenus par les deux algorithmes de reconnaissance, le meilleur score de la méthode SVM est 76%, et le meilleur taux pour la méthode SNN est 69%. Ces résultats sont satisfaisants par rapport à la littérature.

La reconnaissance de l'écriture arabe manuscrite (hors ligne ou en ligne) est un axe de recherche récent et qui est prometteur pour des travaux futurs en vue d'augmenter le taux de reconnaissance, la fiabilité, et d'optimiser les performances de nos machines et l'intégrité de système de reconnaissance. Pour les améliorer, il faut chercher des nouvelles méthodes plus efficaces que les anciennes avec moins d'erreurs. À cet égard, et comme perspectives d'un travail futur, nous proposerons d'appliquer les modèles cachés de Markov (HMM) avec les réseaux de neurone Spike (SNN), et ce la en deux raisons : la puissance des HMMs et leur dynamique, et la rapidité des réseaux Spike, dans le but d'avoir des systèmes de reconnaissance de l'écriture arabe puissants et rapides.

---

---

## Annexe A :

### 1. Algorithme de codage :

Le traitement se fait mot par mot, et la reconnaissance est établie au niveau des caractères. A la fin de la phase de reconnaissance nous rassemblons les caractères reconnus pour reconstruire le mot d'origine (procédure inverse de la segmentation).

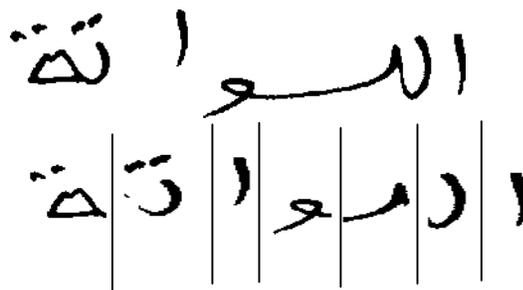
A partir d'une image binaire normalisée de la forme extraite (le plus souvent c'est un caractère) on procède de la façon suivante :

- 1) Mémoriser la position de la forme dans l'étape précédente, et le numéro dans le mot (paw) d'origine.
- 2) Calculer les histogrammes vertical et horizontal de la forme extraite
- 3) Remplir le vecteur des caractéristiques structurales et géométriques : points, dimensions, hampe, jambage
- 4) Calculer le vecteur des moments invariants à partir de l'image de la forme.
- 5) Rassembler les deux vecteurs, calculés précédemment, en un seul. C'est le vecteur des caractéristiques qui va représenter le caractère.
- 6) Créer la base des vecteurs des primitives de chaque forme extraite selon les catégories décrites auparavant.

### 2. Quelques résultats de segmentation et de reconnaissance :

L'opération de la reconnaissance est effectuée au niveau caractères par les deux algorithmes SNN et SVM. Les résultats obtenus de cette dernière sont ensuite concaténés pour reconstruire le mot d'origine, selon les caractéristiques déjà extraites dans la phase de segmentation.

#### *Exemple de segmentation*



Le tableau suivant montre quelques résultats (bon, mauvais, erreur) :

N°	Les Mots	Nbr Paw/mot	Nbr lettres/mot	Nbr lettres segmentés	Mot reconnu
01	نقّة	01	03	03	نقّة
02	حزوة	03	04	04	حزوة
03	رؤاد	04	04	04	رقاد
04	السواتة	04	07	07	

Des échantillons de la base IFN/ENIT DATABASE

مثليز هارث نقّة  
حيا الصلاح الرظراء بوعثمان  
شعاع السعيدة الهدارة

Des échantillons de la base NotreBase

ز زجارة	ز مزيد	ز نشز
س سماء	س نسل	س يسا
ش شبل	ش كشف	ش فتش
ص صوم	ص مصر	ص نص
ض ضوء	ض مضى	ض فض
ط طفل	ط مطر	ط فقط
ظ ظفر	ظ يظهر	ظ يقط
ع عمل	ع معلم	ع لوح
غ غمر	غ نغم	غ نبح
ف فضل	ف قفل	ف يلف

---

---

## Annexe B :

### Les moments invariants :

L'avantage principal de l'extraction des caractéristiques c'est qu'elle permet d'éliminer les redondances des données, et elle représente l'image des caractères par un ensemble de primitives numériques. Ces caractéristiques sont utilisées ensuite par les classifieurs.

Dans notre implémentation, les moments invariants, présentés dans [2], sont utilisés pour la phase de l'extraction des primitives. L'utilisation des combinaisons non linéaires des moments géométriques, nous a donné un ensemble des moments invariants par rapport à la translation, à l'échelle et à la rotation.

Les moments centraux qui sont invariants par rapport à n'importe quelle translation sont définies par la formule suivante :

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

Ou

$$\left. \begin{aligned} x &= \frac{M_{10}}{M_{00}}, y = \frac{M_{01}}{M_{00}} \text{ and} \\ M_{pq} &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \end{aligned} \right\}$$

Dans le cas des images à, la fonction d'intensité d'image continue est remplacée par une matrice, où  $x$  et  $y$  sont les positions des pixels de l'image. Les intégrales dans les deux premières équations sont approximer par des sommes et la formule devient comme suit :

$$M_{pq} = \sum_{x=0}^m \sum_{y=0}^n (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy$$

$$M_{pq} = \sum_{x=0}^m \sum_{y=0}^n x^p y^q f(x, y) dx dy \quad , \text{ ou } m \text{ et } n \text{ sont les dimensions de l'image.}$$

**Tableau les moments invariants représentés par une matrice 4x4**

<i>M</i>	<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>
0	X	X	√	√
1	X	√	√	X
2	√	√	x	x
3	√	X	x	X

L'ensemble des moments invariants  $\Phi_i$  présentés dans [2] sont données par les formules :

$$\Phi_1 = M_{20} + M_{02}$$

$$\Phi_2 = (M_{20} - M_{02})^2 + 4M_{11}^2$$

$$\Phi_3 = (M_{30} - 3M_{12})^2 + (3M_{21} - M_{03})^2$$

$$\Phi_4 = (M_{30} + M_{12})^2 + (M_{21} + M_{03})^2$$

$$\Phi_5 = (M_{30} - 3M_{12})(M_{30} + M_{12})[(M_{30} + M_{12})^2 - 3(M_{21} - M_{03})^2] + (3M_{12} - M_{03})(M_{21} + M_{03})[3(M_{30} + M_{12})^2 - 3(M_{21} - M_{03})^2]$$

$$\Phi_6 = (M_{20} - M_{02})[(M_{30} + M_{12})^2 - (M_{21} - M_{03})^2] + 4M_{11}(M_{30} + M_{12})(M_{21} - M_{03})$$

$$\Phi_7 = (3M_{21} - M_{03})(M_{30} + M_{12})[(M_{30} + M_{12})^2 - 3(M_{21} + M_{03})^2] + 3(M_{21} - M_{03})(M_{21} + M_{03})[3(M_{30} + M_{12})^2 - 3(M_{21} + M_{03})^2]$$

Cette fonction peut être normalisée pour les mettre invariants par rapport à l'échange d'échelle, par l'utilisation des moments centraux normalisés a la place des moments centraux. Les moments centraux normalisés sont donnés par la formule :

$$m_{p,q} = \frac{M_{p,q}}{M_{00}^a}, \text{ ou } a = \frac{(p+q)}{2} + 1$$

Cette formule nous donne sept moments invariants par rapport à la translation, au changement d'échelle et à la rotation. Aussi on peut les remplacer par leurs valeurs logarithmiques.

Nous avons considéré les quatre formes des caractères selon leurs positions, dans l'espace des primitives. Pour chaque caractère, les descripteurs des moments invariants sont calculés, et combinés avec d'autres (structurels, statistiques et géométriques) pour être présentés aux algorithmes de reconnaissance. Chaque caractère est représenté dans une image binaire de 42x 24 (matrice).

---

---

## **Références et Bibliographie :**

- [1] Antoine Cornuéjol et Laurent Miclet, Livre : Apprentissage artificiel : concepts et algorithmes, Préface de Tom Mitchell. Édition EYROLLES deuxième tirage 2003.
- [2] Muhammad Sarfraz, Computer-Aided intelligent recognition techniques and applications (section : Offline Arabic Character Recognition) King Fahd University of Petroleum and Minerals, Kingdom of Saudi Arabia, 2005.
- [3] Emilie poisson, Architecture et Apprentissage d'un Système Hybride Neuro -Markovien pour la Reconnaissance de l'Écriture Manuscrite En-Ligne. Thèse de doctorat université de Nantes, Decembre 2005.
- [4] M. Mansour, M. Benkhadda & A. Benyettou, Optimized Segmentation Techniques for Handwritten Arabic Word and Numbers Character Recognition, 2005.
- [5] A. Belaïd, Analyse et reconnaissance des documents, <http://www.loria.fr/~abelaid/Teaching> (support de cours), 2006.
- [6] A. Belaïd, Reconnaissance de l'Écriture et Analyse de Documents : Numérisation, Pré-traitements (support de cours), 2006.
- [7] A. Belaïd, Analyse de document : Extraction d'éléments (support de cours), 2006.
- [8] A. Belaïd, et al, le projet TDM : Reconnaissance de tables de matières, 1999
- [9] Haitaamar Schahrazed, Segmentation de textes en caractères pour la reconnaissance optique de l'écriture arabe. Thèse de magister université Elhadj-Lakhdar Batna, 2007.
- [10] Haim Cohen et Gadi Bergman, Car License Plate Recognition.
- [11] Abderrazak Zahour, et AL , Contribution à la segmentation de textes manuscrits anciens ,2004
- [12] Dr. Paz Kahana, Advanced Character Recognition ACR Technology and Performance Analysis White Paper, CharacTell September 2003.
- [13] P. Gallinari, Apprentissage Numérique et Réseaux de Neurones (cours), 2001.
- [14] Denis ARRIVAULT, Apport des Graphes dans la Reconnaissance Non-Contrainte de Caractères Manuscrits Anciens. Thèse pour l'obtention du Grade de docteur de l'Université de Poitiers, Mars 2006.
- [15] Wafa Boussemaa et al, Segmentation texte /graphique : Application au manuscrits Arabes Anciens, 2005.
- [16] A. Benouareth, A. Ennaji et M. Sellami Utilisation des HMMs de Durée d'Etat Explicite pour la Reconnaissance des Mots Arabes Manuscrits , 2005.
- [17] Vlad Atanasiu, Le phénomène calligraphique à l'époque du sultanat mamlok . Thèse de Doctorat École pratique des Hautes Études, 2003.
- [18] J. Henry, Analyse de document : Analyse de l'image (support de cours), 2005.

- 
- 
- [19] B. Romaniuk, Modélisation linéaire et non linéaire en rdf statistique (Méthodes a noyaux) support de cours, 2004.
- [20] Dr Suici-Meslati, Reconnaissance de formes et écriture arabe manuscrite , support de cours 2008.
- [21] S. Bourgeois et al, Descripteur de contours pour la reconnaissance d'objets , 2005.
- Tapas kanungo, Non lineaire global and local document dégradation models, 2004.
- [22] Marc Parizeau, Livre : Réseaux de neurones, université LAVAL Automne 2004.
- [23] Andrew D. Back, Neural Network Architectures with Time Delay Connections for Nonlinear Signal Processing.
- [24] Ghassan Mourad, La virgule viendrait-elle de l'écriture arabe ? Constatations sur ses origines graphique et fonctionnelle, TUGboat, Volume 24 (2003), No. 3-Proceedings of EuroTEX 2003.
- [25] Chih-Wei Hsu and Chih-Jin Lin, A comparison of methods for multi-class support vector machines.
- [26] Stephan Ruping, mySVM manual, oct 2000.
- [27] Sylvain Chevallier et Philippe Tarroux, Extraction de saillances par un réseau de neurones impulsionnels, 2007.
- [28] A. Belaïd et AL, La numérisation de documents : Principe et évaluation des performances, 2006.
- [29] Sander Bohte, Spiking neural networks, Thesis Universiteit Leiden.2003,
- [30] H.Zouari et AL, Un panorama des méthodes de combinaison de classifieurs en reconnaissances de formes. In Proc. RFIA 2002, Angers, France, Val.2, pp 499-508.
- [31] Trevor Hastie and Patrice Y . Simard, Metrics and Models for Handwritten Character Recognition. Statistical science Vol. 13 N° 1, 54-65, 1998.
- [32] Karim HADJAR, Une étude de l'évolutivité des modèles pour la reconnaissance de documents arabes dans un contexte interactif. Thèse de doctorat de l'Université de Fribourg (Suisse), 2006.
- [33] Warda Hachour, Reconnaissance hybride des caractères arabes imprimés, JEP-TALN 2004. Traitement automatiques de l'arabe, Fès, 20 Avril 2004.
- [34] Prétraitement et transformation d'images support de cours.
- [35] Maged Mohamed Mahmoud Fahmy, Automatic Recognition Of Handwritten Arabic Characters Using Their Geometrical Feature , 2000.
- [36] *Nawwaf Kharma & Rabab Ward*, A Novel Invariant Mapping Applied to Hand-written Arabic Character Recognition, 1999 .

- 
- 
- [37] A.Bennasri et AL, Extraction des lignes d'un texte manuscrit arabe , Vision Interface '99, Trois-Rivières, Canada, 19-21 May.
- [38] Najoua Ben Amara et autres, Modélisation Pseudo Bidimensionnelle pour la Reconnaissance de Chaînes de Caractères Arabes Imprimés , 1999.
- [39] S. SNOUSSI MADDOURI, Modèle perceptif neuronal à vision globale-locale pour la reconnaissance de mots manuscrits arabes , 2002.
- [40] Samia Snoussi Maddouri and al, Local Normalization Towards Global Recognition of Arabic Handwritten Script, 2002.
- [41] Najoua Ben Amara et al, Une méthode stochastique pour la reconnaissance de l'écriture arabe imprimée, 1996.
- [42] Najoua Ben Amara et al, Utilisation des modèles markoviens en reconnaissance de l'écriture arabe : Etat de l' art, 2000.
- [43] Karim HADJAR et al, Une méthode de reconnaissance interactive pour les documents à structure complexe, 2002.
- [44] Motasem ALRAHABI, Filtrage sémantique de textes en arabe en vue d'un prototype de résumé automatique. Le traitement automatique de l'arabe, JEP-TALN 2004, Fès, 19-22 avril 2004.
- [45] Tim Klassen, Towards Neural Network Recognition Of Handwritten Arabic Letters. In Partial Fulfillment of the Requirements for the Degree of master of computer science, 2001.
- [46] Deya motawa, Segmentation of Arabic cursive script, IEEE 1997.
- [47] Yousef Al-Ohali, MohamedCheriet et Ching Suen, Databases for recognition of handwritten Arabic cheques. Pattern Recognition Society 36 (2003) 111 – 121.
- [48] A. WRÓBLEWSKA ET al, Segmentation and feature extraction for reliable classification of microcalcifications in digital mammograms. OPTO-ELECTRONICS REVIEW 11(3), 227–235 (2003)
- [49] SUNG LIN TSAI, On-line recognition of handwritten Arabic characters , 1999.
- [50] Richard Alan Peters II, Image processing (DIGITAL IMAGE and MATLAB, FREQUENCY FILTERING), 2007.
- [51] Gilles BUREL et al, Caractérisation et classification de textures sur images naturelles, 1991.
- [52] Franck LECLERC, Des gaussiennes pour la modélisation des signatures et la segmentation de tracés manuscrits, 1992.
- [53] Abdel BELAÏD et George SAON, Utilisation des processus markoviens en reconnaissance de l'écriture. Traitement du Signal 1997-Volume 14-n°2.
- [54] R. SABOURIN, G. GENEST, Définition et évaluation d'une famille de représentations pour la vérification hors-ligne des signatures. Traitement du Signal 1995 - Volume 12 - n° 6.

- 
- 
- [55] Mario Pechwitz and Volker Maergner, HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT Database. Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
- [56] Freddy perraud et al, Modèle de langages adaptés à la reconnaissance de l'écriture en ligne, 2004.
- [57] Sofien Touj et al, Generalized Hough Transform for Arabic Optical Character Recognition. Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
- [58] Sorin Moga et Philippe Gaussier, Artificial Neural Network for Sequence Learning , 2001.
- [59] Abir EL ABED, Caractérisation de tracés manuscrits et recherches de similarités en vue de la compression par approche fractale. Rapport de master recherche INSA de Lyon, Juin 2004.
- [60] Volume des Actes de la 7ème Conférence d'Apprentissage (CAp) 2005.
- [61] Ale Koerich et al, Système Hybride de Reconnaissance de Mots Manuscrits sur un Grand Vocabulaire utilisant des Réseaux Neuronaux et des Modèles de Markov Cachés. CIFED. Volume X – n° X/2002
- [62] Bouhlel Bassem ET AL, Algorithmes de détection d'orientation , 2006.
- [63] Philippe BOLON ET AL, Analyse d'image, Traitements de bas niveau (support de cours), 1998.
- [64] Support de cours : ANALYSE D'IMAGE (techniques instrumentales) École des Mines de Saint-Etienne, 1997.
- [65] Lionel Lacassagne, Light Speed Labelling : un nouvel algorithme d'étiquetage en composantes connexes, 1997.
- [66] Geoffrey Hinton and Vinod Nair, Inferring Motor Programs from Images of Handwritten Digits, 2003.
- [67] Gérard DREYFUS, Reconnaissance de formes par réseaux de neurones (support de cours), 2002.
- [68] G.R. Céheux, Traitement automatique de l'écriture et du document (support de cours), 1998.
- [69] Prof. Jean-Philippe THIRAN, Reconnaissance des formes : Segmentation d'images (support de cours).
- [70] Jean-Pierre CRETTEZ ET AL, Reconnaissance de l'écriture manuscrite, (support de cours) 1998.
- [71] Vincent Vigneron, Brève introduction a la Reconnaissance des Formes support de cours ,1994.
- [72] Sonia FLISS et Philippe MOIREAU, Méthode de segmentation pour le recalage d'images CT et PET, 2004.

- 
- 
- [73] L. S. Oliveira, Segmentation de caractères manuscrits basée sur une approche structurale, 1999.
- [74] Andrew D. Back, Neural Network Architectures with Time Delay Connections for Nonlinear Signal Processing.
- [75] Régis Clouard, Une méthode de développement d'applications de traitement d'images, 2006.
- [76] Nawwaf N. Kharma & Rabab K. Ward, Systèmes de reconnaissance de caractères pour les non- experts, 1999.
- [77] Nedjem Eddine Ayat, Selection de modèle automatique des machines a vecteurs de support : application a la reconnaissance d'images des chiffres manuscrits. Thèse de doctorat, Université du QUÉBEC, LE 20 JANVIER 2004.
- [78] Bernard GOSSELIN, Application de réseaux de neurones artificiels a la reconnaissance automatique de caractères manuscrits. Thèse de doctorat Faculté Polytechnique de Mons, 1996.
- [79] Guy Almouzni, Livre : Traitement numérique des images, 2007.
- [80] Sylvain Chevalier, Reconnaissance d'écriture manuscrite par des techniques markoviennes: une approche bidimensionnelle et générique. Thèse de doctorat de l'Université René Descartes - Paris 5, 2004.
- [81] Nadia Benahmed, Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés : sélection et pondération des primitives par algorithmes génétiques. Thèse de doctorat d'Université du QUÉBEC, MARS 2002
- [82] Abdul Rahim AHMAD, Reconnaissance de l'écriture manuscrite en -ligne par approche combinant systèmes à vastes marges et modèles de Markov cachés. Thèse de doctorat de l'Université de Nantes, le 29 Décembre 2008.
- [83] Ramin Mehran, Hamed Pirsiavash et Farbod Razzazi , A Front-end OCR for Omni-font Persian/Arabic Cursive Printed Documents. Proceedings of the Digital Imaging Computing: Techniques and Applications (DICTA 2005).
- [84] Rochel O, Une approche événementielle pour la modélisation et la simulation de réseaux de neurones impulsionsnels. Thèse de l'Université Henri Poincaré, Nancy, 2004.
- [85] Keinosuke Fukunaga, Livre : Introduction to Statistical Pattern Recognition. Academic Press Second Edition, 1990.

Philippe Besse, Apprentissage Statistique & Datamining -Version Octobre 2006, pp.101-108.